

Spring 2018

PHASOR MEASUREMENT UNIT TESTING AND CHARACTERIZATION

Joe Perigo
Montana Tech

Follow this and additional works at: https://digitalcommons.mtech.edu/grad_rsch



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

Recommended Citation

Perigo, Joe, "PHASOR MEASUREMENT UNIT TESTING AND CHARACTERIZATION" (2018). *Graduate Theses & Non-Theses*. 175.

https://digitalcommons.mtech.edu/grad_rsch/175

This Thesis is brought to you for free and open access by the Student Scholarship at Digital Commons @ Montana Tech. It has been accepted for inclusion in Graduate Theses & Non-Theses by an authorized administrator of Digital Commons @ Montana Tech. For more information, please contact sjuskiewicz@mtech.edu.

PHASOR MEASUREMENT UNIT TESTING AND CHARACTERIZATION

by
Joe Perigo

A thesis submitted in partial fulfillment of the
requirements for the degree of

Master of Science in Electrical Engineering

Montana Tech

2018



Abstract

The power grid is a complex system composed of many different elements. Knowledge of electrical waveform content across the grid is key to managing power and maintaining stability. Electrical waveforms can be represented mathematically as time-varying phasors. Electrical phasors give information pertaining to the magnitude of the waveform as well as the phase relationship of the waveform to a reference. A Phasor Measurement Unit (PMU) is a device that is installed at a node on the power grid and measures electrical phasors. The potential of near real time phasor information is powerful and has driven PMU application technology forward at a rapid pace over the last decade. The IEEE C37.118 standard establishes requirement guidelines for PMU performance. Before PMUs are implemented in real time applications researchers may wish to test PMUs to obtain specific device behavior. This thesis will describe the development of a robust PMU test bench that is capable producing dynamic input signals. Dynamic testing and analysis will reveal time domain and frequency domain characteristics of the PMU under test. A discussion on the reported response and the real time response of PMUs will be covered to give more insight into the real time behavior and application of PMUs. Conclusions will take one last look at the project and discuss the potential of this work as a starting point for future PMU testing.

Keywords: Phasor Measurement Unit, PMU, IEEE C37.118 Standard, Signal Generation, Step Testing, Phase Modulation Testing, PMU Frequency Response

Dedication

This work is dedicated to my two children; Donny and Milla Perigo. Without you two I would not have been inspired to become something more in life. You guys make me proud every day, and this degree is our accomplishment! This chapter of our lives is almost over, and soon we will be on to bigger and better things!

Acknowledgements

I have been at Montana Tech a long time. It feels like the Electrical Engineering department has become a second home to me and the professors a second family. I've spent many hours with all of you over the years, and much of that time has been in your office or in the lab after hours. If it wasn't for your help and understanding I wouldn't have made it.

Dr. Dan Trudnowski first gave me the opportunity to work on testing PMUs when I was in my junior summer. This opportunity would eventually lead to this thesis. I'll never forget sitting in Dan's office when I had to confess my FE test results. Dan knew I wanted to go to grad school and I was sure he would be upset with me. Instead, Dan gave me some of the best advice of my life and encouraged me to take the test again, so I could start work on my masters degree. To me and the rest of the students Dan is more than just a department head, or a dean. Dan pushes us and helps us realize our potential. Thanks for everything Dan!

Finally, I want to say thanks and 'rock on' to Dr. Josh Wold! It never felt right to call Josh my 'committee chair'. I consider Josh my mentor and a good friend. Two important moments stand out to me the most in college. The first was our IEEE field trip to visit SEL's campus in Pullman. Hanging out with Josh and seeing how awesome life could be was impacting and motivated me to do step up my game! The second was sitting with Josh and the SEL crew at our graduation dinner. It feels really good to belong to something that took so much work to achieve. I couldn't have done it without you! Thanks Josh!

Table of Contents

ABSTRACT	II
DEDICATION	III
ACKNOWLEDGEMENTS	IV
LIST OF TABLES	VIII
LIST OF FIGURES.....	IX
LIST OF EQUATIONS	XII
1. INTRODUCTION	1
1.1. <i>Problem Statement</i>	1
1.2. <i>Thesis Organization and Scope</i>	2
2. THE PHASOR MEASUREMENT UNIT	4
2.1. <i>PMU Applications</i>	5
2.1.1. Power System Analysis and Model Validation.....	6
2.1.2. Power System Monitoring.....	6
2.1.3. Power System Control.....	7
2.2. <i>IEEE C37.118 Standard</i>	7
2.2.1. TVE, FE, RFE	8
2.2.2. Synchrophasors	9
2.2.3. Frequency and ROCOF Estimation.....	10
2.2.4. Step Testing Specifications	11
2.2.5. Phase Modulation Testing Specifications	13
3. TEST BENCH DESIGN REQUIREMENTS	15
3.1. <i>Block Diagram</i>	16
3.2. <i>National Instruments Hardware</i>	17
3.2.1. NI PXIe-1062Q	17

3.2.2.	Slot 1: NI PXIe-8840 Embedded Controller.....	19
3.2.3.	Slot 2: NI PXI-6683H Timing and Synchronization Module.....	19
3.2.4.	Slot 3: NI PXIe-4300 with NI TB-4300B	20
3.2.5.	Slot 4: NI 6124 Multifunction DAQ with NI TB-2706 Terminal Block.....	20
3.2.6.	Slot 5: NI PXIe-6738 Analog Output Module and SCB-68A I/O Block	20
3.3.	<i>SEL-2407 GPS IRIG-B Clock</i>	<i>21</i>
3.4.	<i>Tabor Electronics 9400-25 4 Channel Voltage Amplifier.....</i>	<i>21</i>
3.5.	<i>SEL 3373 Phasor Data Concentrator</i>	<i>22</i>
3.6.	<i>Test Bench: Software.....</i>	<i>23</i>
3.6.1.	Matlab	23
3.6.2.	National Instruments LabVIEW	23
3.7.	<i>Test Bench Signals.....</i>	<i>24</i>
3.7.1.	Test Bench Signals – ‘GeneralSigGen.m’ Matlab Software.....	25
3.8.	<i>Test Bench Timing</i>	<i>26</i>
3.9.	<i>Test Bench Accuracy.....</i>	<i>28</i>
3.9.1.	Oscilloscope Clock Test	28
3.9.2.	NI PXIe-6738 Testing and Analysis Using PMU Phase Angle Data	30
3.9.3.	NI PXIe-6738 Testing and Analysis Using PXIe-4300.....	34
4.	PMU TESTING	37
4.1.	<i>Pre-testing Procedures and Precautionary Steps.....</i>	<i>37</i>
4.2.	<i>Testing Procedure</i>	<i>39</i>
4.2.1.	Signal Generation Procedures	39
4.2.2.	PMU Data Acquisition	42
4.2.3.	Parsing PMU Output Data and Analog Input Data	47
4.2.4.	Summary - Running a Test.....	48
5.	PMU STEP TESTING.....	49
5.1.	<i>Magnitude Step Tests – Input Signal Description.....</i>	<i>49</i>
5.1.1.	Magnitude Step Test – Results	52

5.2.	<i>Frequency Step Tests – Input Signal Description</i>	54
5.2.1.	Frequency Step Test - Results.....	55
5.3.	<i>Construction of High Resolution PMU Step Response</i>	56
5.3.1.	High Resolution PMU Step Response - Results.....	58
6.	PHASE MODULATION TESTING	66
6.1.	<i>Phase Modulation Signal Description</i>	66
6.2.	<i>Analysis – Producing the Bode Plot - Overview</i>	70
6.3.	<i>Analysis – Producing the Bode Plot – Deriving Frequency</i>	75
6.4.	<i>Analysis – Producing the Bode Plot – Replacing the FFT with NLLS</i>	76
6.4.1.	Accounting for Initial Phase.....	77
6.5.	<i>Analysis – Producing the Bode Plot – Accounting for Imperfect Sample Rate</i>	79
6.6.	<i>Summary - The Transfer Function Calculation with Correction Factors and the Bode Plot</i>	80
7.	DISCUSSION.....	84
7.1.	<i>The Bode Plot</i>	84
7.2.	<i>PMU Dynamics for Real Time Applications</i>	85
7.2.1.	Dr. Trudnowski’s Phase Requirements Compared to the C37.118 Standard Frequency Error Requirements	86
7.2.2.	Dr. Trudnowski’s Requirements Compared to the PMU Under Test.....	87
8.	CONCLUSIONS.....	88
9.	REFERENCES CITED (OR BIBLIOGRAPHY)	89
10.	APPENDIX A: DR. TRUDNOWSKI’S PHASE REQUIREMENTS COMPARED TO C37.118 STANDARD FE REQUIREMENTS	91

List of Tables

Table I: Results - Testing for True Sampling Frequency of the PXIe-6738 Analog Output Module	36
Table II: Magnitude Step Tests – Step On Start Times	57
Table III: Phase Modulation Testing – Modulation Frequency Battery Test	67
Table IV: Appendix A Summary	86

List of Figures

Figure 1: PMU Signal Processing from C37.118 Standard	5
Figure 2: Pulse Per Second	9
Figure 3: UTC Cosine Reference vs. PMU Signal	10
Figure 4: Step Tests – Delay Time and Overshoot	12
Figure 5: Test Bench Block Diagram	16
Figure 6: Generic and Detailed PXIe-1062Q Layout by Slot Number.....	18
Figure 7: A/B/C Phase Base Signal Matlab Script	25
Figure 8: Matlab Script for Equation 19.....	25
Figure 9: Matlab Script for Equation 20.....	26
Figure 10: Test Bench with Timing Versus Test Bench With Out Timing.....	27
Figure 11: Oscilloscope Clock Test – Set Up.....	29
Figure 12: Imperfect Sample Rate – Frequency Test	32
Figure 13: Imperfect Sample Rate – Phase Angle	33
Figure 14: Analog Reader Front Panel	34
Figure 15: 300V_AI_Isolated/ai0/1/2/3.....	35
Figure 16: Zeroing Test Signals.....	38
Figure 17: PXIe 6738 – Two Methods for Signal Generation.....	39
Figure 18: PXIe 6738 – GenThreePhase.vi Software Front Panel	40
Figure 19: PDC Assistant – Connecting to the SEL 3373	42
Figure 20: PDC Assistant – Configure Input.....	43
Figure 21: Assistant – Configure PMU Archive.....	44
Figure 22: PDC Assistant – Configure Archive Collection.....	45

Figure 23: PDC Assistant – Retrieve Archive	46
Figure 24: Basic csv Reader Software	47
Figure 25: Setting Time Vector for Magnitude Step Test	50
Figure 26: Structured Parameter Settings for Magnitude Step Test	51
Figure 27: Magnitude Step Test – Input Signals with Trigger	52
Figure 28: PMU Magnitude Step Response – Full Response.....	53
Figure 29: Zoom On Non-Causal PMU Response	54
Figure 30: Frequency Step Test Input Signal	55
Figure 31: Frequency Step Test	56
Figure 32: Magnitude Step Test 1 – 10% Step at 15 Seconds	58
Figure 33: PMU Response to Magnitude Step Tests 1-11 All Plotted Together	59
Figure 34: Zoomed PMU Response to Magnitude Step Tests 1-11 All Plotted Together	59
Figure 35: PMU Response to Amplitude Steps – No Time Adjustment	60
Figure 36: PMU Response to Amplitude Steps – Time Adjusted	61
Figure 37: PMU Response to Amplitude Steps – High Resolution Constructed Curve....	62
Figure 38: High Resolution Step Response – Advanced Timestamping	63
Figure 39: PMU Response to Magnitude Step Test - Overshoot.....	64
Figure 40: PMU Step Response – Delay Time.....	65
Figure 41: Phase Modulation Signal Generation Software.....	69
Figure 42: PMU Phase Angle Output	70
Figure 43: Phase and Frequency	72
Figure 44: FFT Analysis of Frequency	73
Figure 45: PMU Phase Modulation – Phase and Frequency Tracking.....	74

Figure 46: Backward Euler as FIR Filter	75
Figure 47: Backward Euler Delay.....	76
Figure 48: Zoom of PMU 0.1 Hz Phase Modulation Output.....	78
Figure 49: Phase Modulation Testing – PMU Input and Test Bench Error.....	80
Figure 50: Phase Modulation Testing – PMU Output and Post Processing Errors	81
Figure 51: PMU Reported Bode Plot.....	82
Figure 52: PMU Frequency Response – Reported vs Real Time	85
Figure 53: PMU Under Test versus Dr. Trudnowski’s Real Time Requirements.....	87

List of Equations

Equation

(1)	4
(2)	4
(3)	8
(4)	8
(5)	8
(6)	10
(7)	11
(8)	11
(9)	11
(10)	11
(11)	11
(12)	12
(13)	13
(14)	16
(15)	24
(16)	24
(17)	24
(18)	25
(19)	25
(20)	26
(21)	30

(22)	31
(23)	33
(24)	64
(25)	71
(26)	71
(27)	71
(28)	75
(29)	75
(30)	75
(31)	76
(32)	77
(33)	77
(34)	77
(35)	78
(36)	79
(37)	79
(38)	80
(39)	80
(40)	81
(41)	81
(42)	86
(43)	91
(44)	91

(45)	91
(46)	91
(47)	92
(48)	92
(49)	93
(50)	93
(51)	93

1. Introduction

Although Phasor Measurement Units (PMUs) have been around since the 1980s, their implementation into the power grid is relatively new. The basic concept behind PMUs is simple. It's the idea that an electrical engineer could place the leads of an oscilloscope many miles apart and measure waveforms at nodes of interest on the power grid. This concept and its potential have motivated engineers into developing PMU application technology. In 1995 the IEEE began work on standardizing PMU performance. Today the IEEE C37.118 standard for synchrophasor measurements gives power-system researchers a valuable reference for PMUs. Knowledge of device performance is very important before use in the power grid. The safety and reliability of the power grid is mission critical to utilities that operate them. When safe operation of the power grid is paramount, utilities and researchers will find great value in the C37.118 standard but may also wish to test devices such as PMUs themselves. This thesis aims to establish an understanding of PMU testing and PMU performance characteristics with an emphasis on real time dynamic frequency response characteristics.

1.1. Problem Statement

This thesis covers many areas pertaining to PMU testing, but one goal drives the development of each element of the project; testing a PMU's real time dynamic response. The project can then be split into three main components. The first component will be the design and construction of a PMU test bench. The second component will be dynamic testing in the form of step testing and phase modulation testing. The third component will gather results from dynamic testing to derive PMU real time frequency estimation and will also summarize the results in the form of a Bode plot for the PMU under test. Producing a real time Bode plot for a PMU is not without challenges and many hurdles needed to be crossed before the end product was attained.

The sections of this thesis divide up the challenges into a step by step process that will serve as a road map for future engineers who want to know about PMU testing, and PMU functionality through empirical analysis.

1.2. Thesis Organization and Scope

The temptation to jump straight into testing a PMU may be great, but a theoretical look into these devices is necessary to understand what a PMU is at a fundamental level. Section 2 of this thesis is dedicated to understanding PMUs. Phasor mathematics will be summarized to provide the reader with an understanding of what PMUs do. Applications of PMUs will be discussed to provide the reader with the motivation that drives PMU implementation forward. The IEEE C37.118 standard will then be covered in part. This section will focus on key aspects of PMUs that pertain to their dynamics as this relates to the main objective of this thesis. The C37.118 standard will help define certain characteristics and mathematics that go into PMU testing.

Section 3 of this thesis will focus on the design and construction of a PMU test bench. Without a robust test bench, the goals of this project will be impossible to properly achieve. The test bench design was focused on being able to test PMUs with a wide range of dynamic input signals, and to be able to accurately time input signals and PMU output data. The test bench is complex. The goal of this section is to provide test bench users with the knowledge they need to operate the test bench. This requires a breakdown of test bench hardware and software. Each component of the test bench will be covered and summarized in block diagram format. Test bench software will be covered in part in this section. The details of the software developed for this project will be covered in depth in the testing sections. A discussion on the accuracy of the

test bench will cover critical aspects of the hardware that need to be accounted for by test bench users to produce good test results.

Section 4, section 5, and section 6 will describe PMU testing. Starting with section 4, test bench users will become familiar with the more generic aspects of PMU testing. Signal generation, PMU data archiving and data parsing will be covered here. Section 5 will cover PMU step testing. This will allow test bench users to bench mark PMUs under test and see PMU performance as compared to the IEEE C37.118 standard. This section will lead into the topic of FIR filters and filtering windows. Filter windowing and the results from this section will play an important role in the discussion section. Section 6 covers phase modulation testing. Section 6 will also investigate the analysis techniques used to produce a Bode plot. The results from step testing and phase modulation testing will come full circle in the discussion section where real time frequency estimation will be covered. These results will then be summarized in the form of a real time Bode plot for the PMU under test.

2. The Phasor Measurement Unit

A PMU is a device that measures electrical phasors on the power grid. Phase relationships between waveforms will be made with respect to a reference phasor called the Universal Coordinated Time reference, or the UTC reference. The UTC reference will be described in detail in section 2.2. Simple AC waveforms can be represented using trigonometric functions shown in (1).

$$v(t) = V_p \cos(2\pi f_o t + \phi) \quad (1)$$

where V_p is the peak signal voltage, f_o is the fundamental frequency in Hz, and ϕ is the phase of the signal in radians. The trigonometric form in (1) can be simplified to phasor form in (2) using Euler's identity.

$$\bar{V} = \frac{V_p}{\sqrt{2}} (\cos\phi + j\sin\phi) = \frac{V_p}{\sqrt{2}} e^{j\phi} = \frac{V_p}{\sqrt{2}} \angle\phi \quad (2)$$

PMUs measure the rms amplitude and phase angle of the waveform with respect to a UTC reference phasor. PMUs can also measure frequency and rate of change of frequency. When these measurements are taken they are timestamped via GPS time. Phasors measurements that are synchronized to an absolute time base then become synchrophasors. PMUs may be located miles apart throughout the power grid. By taking measurements with respect to a common time base it is then possible to determine the relative phase of the signals at each location in the grid with respect to a UTC reference phasor. The UTC reference phasor is based on nominal system frequency, 60 Hz for North America. A PMU must be connected to a GPS device to establish an absolute time base for each signal that it measures.

There are many ways that PMUs can accomplish the task of providing the phasor content of the electrical signals that it measures. The C37.118 standard provides an example of the signal processing that takes place inside a PMU as shown in Figure 1.

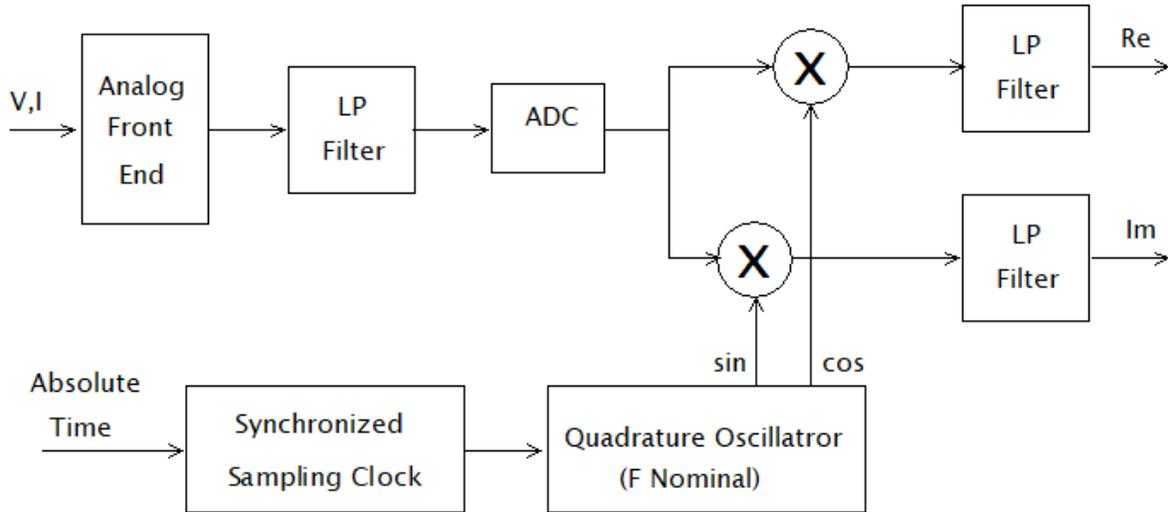


Figure 1: PMU Signal Processing from C37.118 Standard

2.1. PMU Applications

Aging infrastructure, new sources of energy that behave differently than traditional sources, and a growing demand for power have put new pressure on today's power engineers [2]. More than ever it is critical to monitor, protect, and control the power grid with a high level of resolution, certainty and accuracy. Supervisory Control and Data Acquisition (SCADA) systems report measurements on average every 4-10 seconds [8]. The power grid is a dynamic system that can change rapidly. The time resolution of SCADA measurements will likely miss critical changes in the power system and can only report magnitude information. PMUs can provide measurements at 60 sps of both magnitude and phase at their bus location [5]. This near real time data is often made a selling point for PMU manufactures and is known as situational awareness. High resolution phasor data available at near real time makes PMUs an attractive option for electrical engineers who wish to model, monitor, and control the modern power grid.

2.1.1. Power System Analysis and Model Validation

Power engineers use system models built on first principles to simulate grid performance and understand its behavior [2]. Synchrophasors are a cost-effective option for using real world data to build accurate power system models. Side by side comparisons of real world event data and simulated data aids in the development and validation of system models. These models are then incorporated into power system studies that are more representative of real power grid behavior. With a better understanding of how individual elements in the power system interact with the grid as a whole, engineers are able to operate the grid with greater certainty and less stability risks.

2.1.2. Power System Monitoring

In the 1960s mathematical methods were developed to predict the state of the power grid based on a scan of measurements. A traditional system for obtaining these measurements is SCADA [2]. Assumptions are made regarding these measurements, primarily that when the scan takes place that the power system is operating in steady state. This assumption can lead to errors in the state estimation mathematics used to monitor the power grid [2]. State estimation models are recursive in that they predict the current state of the system based on its previous state. New measurement scans provided by SCADA refresh the model and the cycle begins again [2]. Situational awareness can enhance a power system engineers ability to observe the dynamic nature of the power grid. This observability helps operators and engineers maintain safe operation of the grid as opposed to operating based on a best guess of the true state of the power grid. PMUs have also been employed in hybrid state estimation algorithms to provide better estimates of the current state of the grid [1]. As PMU installation throughout the power grid increases state estimation can then be replaced with PMU measurements.

2.1.3. Power System Control

Synchrophasor data acquired from locations many miles apart give control algorithms new sources of feedback information. Some traditional feedback controllers in power systems, such as those used to control electric machines, used localized signals to determine the controller's actions [5]. Using only localized signal information isolates these control schemes and requires system models to estimate the world outside of the controller's sensors. Measurements taken at sub-second intervals (60 sps) can be collected in Wide Area Measurement Systems (WAMS) and then used by control algorithms to represent the power grid instead of system models. Synchrophasor data allow new control schemes to manage machines, system frequency, system voltage and power flow over wide areas of the electric grid [1].

Inter-area oscillation damping is a control application that makes use of PMUs as sensors in the feedback loop. Oscillations in power can cause system instability and require that utilities operate lines at conservative power levels. A controller designed to dampen these oscillations becomes an attractive option for maintaining stability and allows utilities to operate lines at a higher capacity. Damping control requires real-time synchrophasor data to detect oscillations and to determine control action. Of particular importance to this control scheme is the performance of the PMU in real time and its ability to estimate frequency. [7]

2.2. IEEE C37.118 Standard

The IEEE C37.118 Standard for Synchrophasor Measurements provides industries and researchers with guidelines that all manufactured and commercially available PMUs are encouraged to meet. Compliance to this standard ensures that measurements taken by different PMUs will yield similar results within a margin of error. There are four main guidelines that the C37.118 standard establishes; synchrophasor estimation, frequency, rate of change of frequency,

and time tagging. Guidelines are based on the concept of Total Vector Error (TVE), Frequency Error (FE), and Rate of Change of Frequency Error (RFE).

2.2.1. TVE, FE, RFE

The C37.118 standard quantifies allowable margins of error in synchrophasor measurements using the concept of Total Vector Error (TVE). Phasors are complex numbers with a real and imaginary part. The TVE limit can be viewed as a circle in the complex plane that is centered around the tip of a true phasor. The tip of the estimated phasor must lie within the TVE circle. Mathematically the TVE is the difference between samples of the inputs and the estimated sampled values that the PMU calculates. The C37.118 standard normalizes the magnitude in per unit as shown in equation (3).

$$TVE(n) = \sqrt{\frac{(\hat{X}_r(n) - X_r(n))^2 + (\hat{X}_i(n) - X_i(n))^2}{(X_r(n))^2 + (X_i(n))^2}} \quad (3)$$

The estimated quantities $\hat{X}_r(n)$ and $\hat{X}_i(n)$ are given in rectangular form for each n sample contained in the vectors. The actual quantities $X_r(n)$ and $X_i(n)$ are also given in rectangular form. The TVE encapsulates all measurement error in a single number instead of separating vector error by magnitude or phase.

Frequency Error (FE) and Rate of Change of Frequency Error (RFE) are described similar to TVE in that they take the difference between an actual quantity input to the PMU and the estimated value measured by the PMU. FE and RFE are defined by the C37.118 standard as

$$FE = |f - \hat{f}| \quad (4)$$

$$RFE = \left| \left(\frac{df}{dt} \right) - \left(\frac{d\hat{f}}{dt} \right) \right| \quad (5)$$

where f is the actual frequency at the inputs of the PMU and \hat{f} is the PMUs estimated frequency. The C37.118 standard specifies two classes of PMUs related to performance and precision. Protection class PMUs (P class) have the fastest response time. P class PMUs are an attractive option for real-time applications or applications requiring speed of measurements. Metering class PMUs (M class) have a slower response characteristic than P class PMUs. M class PMUs focus on accuracy of measurements and are used in applications where aliasing could become detrimental. When considering the performance of PMUs, the C37.118 standard specifies allowable TVE, FE, and RFE for each class of PMU. Further, the C37.118 standard specifies allowable TVE, FE, and RFE in steady state conditions and dynamic conditions. PMU response to dynamic input signals is the primary focus for this project and the following C37.118 standard subsections.

2.2.2. Synchrophasors

Coordinated Universal Time (UTC) is the baseline for GPS time. It is defined as the time of day at the Earth's prime meridian. UTC time is distributed via satellite GPS. A Pulse Per Second (PPS) function is used to describe time in seconds. This concept is illustrated in Figure 2 below.

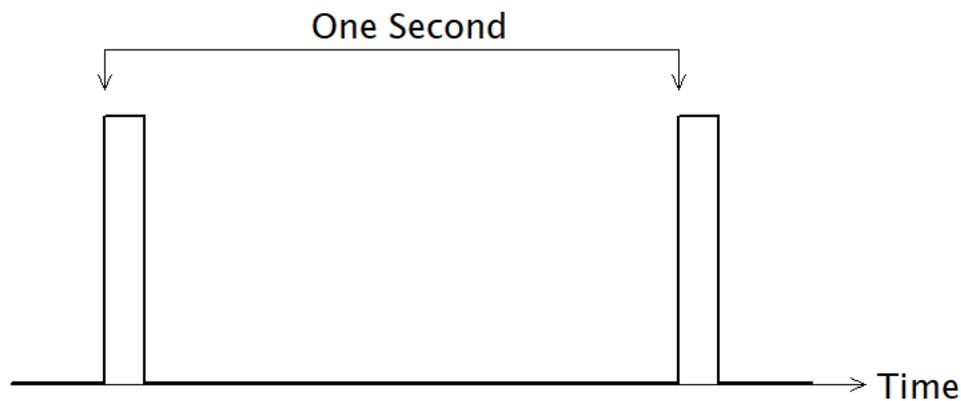


Figure 2: Pulse Per Second

Consider the cosine function in (1). If this function has its maximum at the top of each second based on UTC time the phasor function in (3) takes the value of $\frac{V_{peak}}{\sqrt{2}}$. This cosine is based on nominal system frequency, 60 Hz in North America. The UTC reference waveform used to define phase relationships between PMUs is this cosine function. Figure 3 illustrates how the phase relationship between the UTC cosine waveform may be used to calculate a phasor. The C37.118 standard requires PMUs make timestamps relative to GPS time based on UTC time. Equation 6 shows how the phase relationship may be calculated.

$$\phi = \frac{\Delta t}{T} (360^\circ) = \frac{0.0077 \text{ sec}}{0.01667 \text{ sec}} (360^\circ) = 166^\circ \quad (6)$$

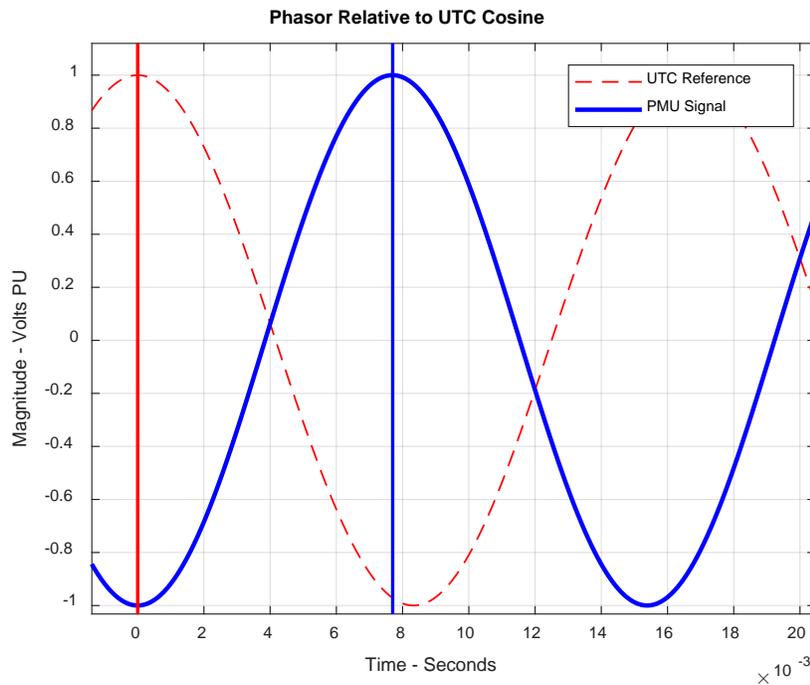


Figure 3: UTC Cosine Reference vs. PMU Signal

2.2.3. Frequency and ROCOF Estimation

The C37.118 standard specifies PMU frequency estimation based on the following mathematics. If the cosine function described in (1) takes the form in (7):

$$v(t) = V_p \cos[\psi(t)] \quad (7)$$

where

$$\psi(t) = 2\pi f_o t + \phi(t) = 2\pi \left[f_o t + \frac{\phi(t)}{2\pi} \right] \quad (8)$$

Frequency is defined as the derivative with respect to time of $\psi(t)$ in (7) as shown in (9) below.

$$f(t) = \frac{1}{2\pi} \frac{d}{dt} [\psi(t)] = \frac{1}{2\pi} \frac{d}{dt} \left[2\pi \left[f_o t + \frac{\phi(t)}{2\pi} \right] \right] = f_o + \frac{d}{dt} \left[\frac{\phi(t)}{2\pi} \right] = f_o + \Delta f(t) \quad (9)$$

where f_o represents nominal system frequency and $\Delta f(t)$ represents a deviation from the nominal system frequency. Rate of change of frequency is defined as

$$ROCOF = \frac{d^2}{dt^2} \left[\frac{\phi(t)}{2\pi} \right] = \frac{d}{dt} [\Delta f(t)] \quad (10)$$

2.2.4. Step Testing Specifications

As part of dynamic compliance, the C37.118 standard lists allowable overshoot and delay time for step inputs. The requirements are listed for step inputs applied to the input magnitude and input phase terms. Three phase signals A/B/C are defined in the standard for step testing. Equation (11) below describes the A phase signal. B and C phases are shifted in phase by $\pm 120^\circ$.

$$X_a = X_m [1 + k_x f_1(t)] \times \cos[\omega_o t + k_a f_1(t)] \quad (11)$$

where X_m is the amplitude term, $f_1(t)$ is a unit step function, ω is the modulation frequency, ω_o is the nominal system frequency, k_a is the phase angle modulation factor, and k_x is the amplitude modulation factor. Step testing is defined by the C37.118 standard as a transition from one steady state to another steady state where the magnitude or phase of the input signal has changed. The C37.118 standard uses step tests to determine delay time and overshoot as illustrated below.

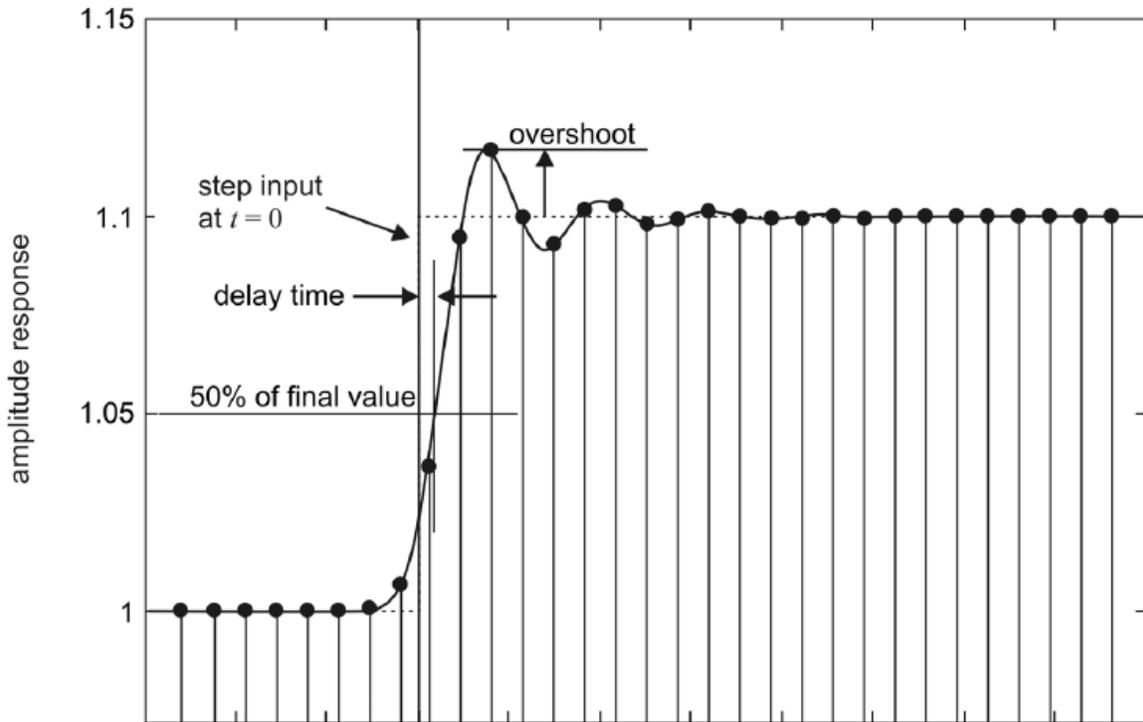


Figure 4: Step Tests – Delay Time and Overshoot

For a 10% step in magnitude, or where $k_x = 0.1$, delay time is allowed to be

$$Delay\ Time_{Fs\ 60sps} = \frac{1}{4 \times F_s} = \frac{1}{4 \times 60} = 0.004167\ seconds \quad (12)$$

Overshoot requirements for M class PMUs are given for a 10% step in magnitude. The requirement states that the maximum allowable overshoot be 5% of step magnitude. Overshoot requirements for P class PMUs are given for a 10% step in magnitude. The requirement states that the maximum allowable overshoot be 10% of step magnitude. An important consideration is the non-causal behavior seen to step inputs as shown in Figure 4. The C37.118 standard describes measurement reporting latency as the time interval between when an event occurs at the input of the PMU to when it is reported and available as PMU output data. Total latency will include filter group delay and processing time. The C37.118 standard allows for group delay compensation in the form of measurement reporting latency which results in the non-causal

behavior seen in Figure 4. If all group delay were compensated perfectly by advanced time stamping, then the delay in Figure 4 would be zero. The delay time in (12) can be seen as an allowable margin that group delay compensation can be off from. The allowable limits for measurement reporting latency are listed for P class PMUs and M class PMUs. For P class PMUs the maximum allowable measurement reporting latency is defined as $2/F_s$, where F_s is the reporting rate of the PMU. For this thesis the reporting rate will be set to the maximum of 60 sps. This means that for the testing performed in this thesis a P class PMU should have no more than $2/60$ seconds of measurement reporting latency. For M class PMUs the maximum allowable reporting latency is defined as $5/F_s$. For the testing to be done in this thesis an M class PMU should have no more than $5/60$ seconds of measurement latency.

2.2.5. Phase Modulation Testing Specifications

Three phase signals A/B/C are defined in the standard for modulation testing. Equation (13) below describes the A phase signal. B and C phases are shifted in phase by $\pm 120^\circ$.

$$X_a = X_m[1 + k_x \cos(\omega t)] \times \cos[\omega_o t + k_a \cos(\omega t - \pi)] \quad (13)$$

The C37.118 standard specifies maximum allowable TVE for synchrophasor measurements in the presence of phase modulation. For phase modulation testing the phase angle modulation factor, k_a , is set to 0.1. The bandwidth of phase modulation frequencies for P class PMUs is set to be within 0.1 Hz to 2 Hz. The bandwidth of phase modulation frequencies for an M class PMU is to be within 0.1 Hz to 5 Hz. For both P class and M class PMUs the maximum allowable TVE for synchrophasor measurements in the presence of phase modulation is to be 3% or less for their respective modulation frequency bandwidths. Maximum allowable FE is listed in the C37.118 standard for frequency measurements in the presence of phase modulation. For P class

PMUs the maximum allowable FE is 0.06 Hz, while the maximum allowable FE for M class PMUs is 0.3 Hz.

3. Test Bench Design Requirements

The main objective of this test bench design will be to provide a means for testing a PMU with dynamic input signals. Dynamic input signals will give insight into PMU behavior during transient conditions on the power grid. A final objective will be to provide a full frequency response for the PMU under test in the form of a Bode plot. With these objectives in mind the following section will describe test bench design, construction, and will give details on the specific hardware implemented in the final design. Testing for accuracy of the test bench will also be covered here. This will provide test bench users with an idea of what errors occur on the bench and how to account for them in test analysis.

The heart of this test bench will be a data acquisition and signal generation system. This system will be able to produce user specified analog signals as inputs to the PMU under test, as well as collect PMU output data. Test bench timing will also be a key aspect of the design since analog input signals will need to be compared with respect to time against PMU output data. In other words, it will be important that the timing of the analog input signals correlate to the time stamps of the PMU output data. To accomplish this the test bench will have a GPS clock that is then connected to both the PMU and the signal generation system. The design of test signals will be most easily accomplished via a scripting language such as Matlab or a VI language such as LabVIEW. LabVIEW will also be employed to communicate with hardware components on the test bench. PMUs installed in the power grid will measure voltage signals at approximately 120 volts, and as such the test bench will need an amplification unit to bring low voltage signals from the DAC analog output up to realistic levels. The amplification unit will be required to replicate the bandwidth of the DAC analog output. Concerns regarding the voltage amplifier's current output threshold will be a factor. This is because PMUs employ instrument transformers at their

inputs to convert high voltage signals to low voltage signals for processing. The inputs of the PMU therefore reflect an inductive load as shown below.

$$Z_l = j\omega L \ (\Omega) \quad (14)$$

With low frequencies the corresponding impedance of the PMU (Z_l) will be reduced to a level that is dangerous to the amplification unit, as it exceeds the maximum current output for the device. With this design in mind the following sections will describe in detail the test bench, hardware components used, and software used.

3.1. Block Diagram

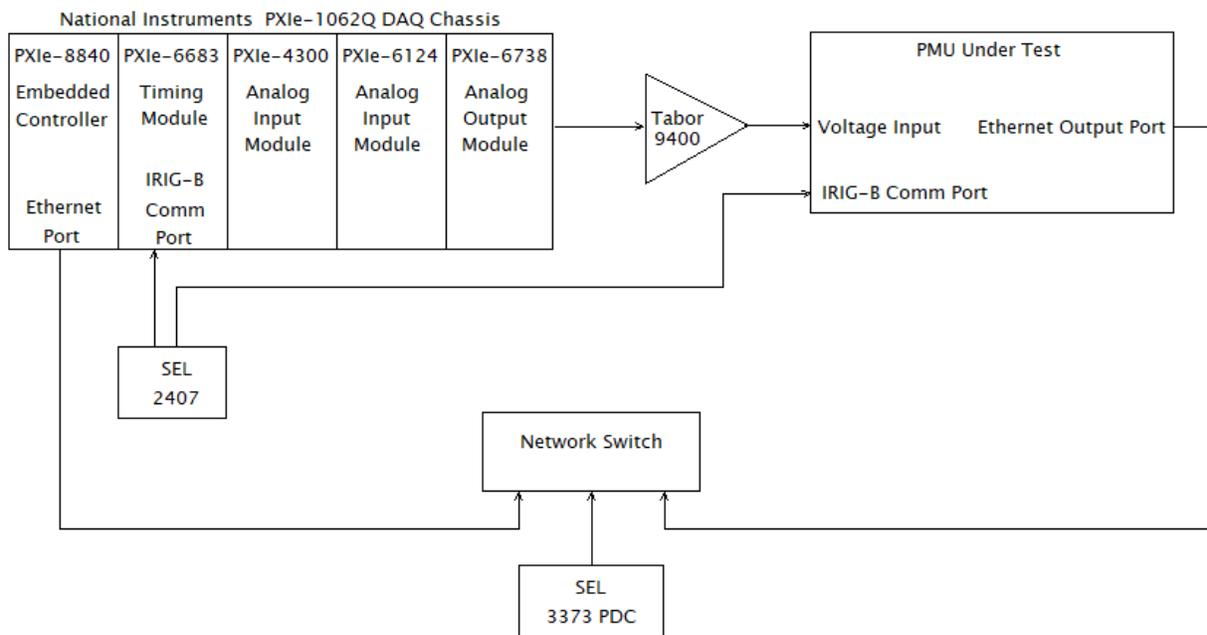


Figure 5: Test Bench Block Diagram

Figure 5 visually summarizes the test bench layout. Each element in this setup will be described in detail in the following sections. Hardware interfacing, wiring, and hardware to software interfacing will also be described.

3.2. National Instruments Hardware

National Instruments (NI) hardware was used for signal generation and data acquisition for this test bench. NI hardware meets all test bench design requirements. Another important feature of NI hardware is that the accompanying NI software, LabVIEW, allows users to write custom software for signal generation and communication between all installed NI hardware.

3.2.1. NI PXIe-1062Q

Data acquisition and signal generation were key elements to this project. The NI PXIe-1062Q is a DAQ chassis that allows for time coordination between different NI hardware. Coordination between installed devices is accomplished via NI LabVIEW software that was developed for this test bench. This software will be covered in the test bench software section below. The PXIe-1062Q DAQ chassis is flexible by including four PXI peripheral slots, one PXI Express slot, and two PXI Express hybrid slots. A high-bandwidth backplane allows for time coordination between the PXI Express slotted devices and the PXI Express hybrid slots. Time coordination is key for outputting signals to the PMU at user specified times.

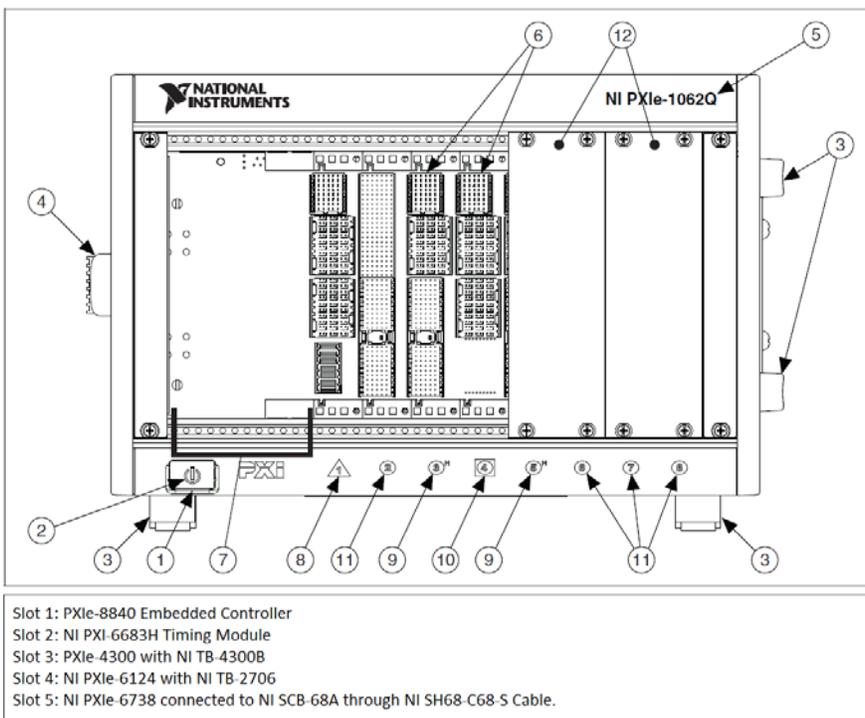
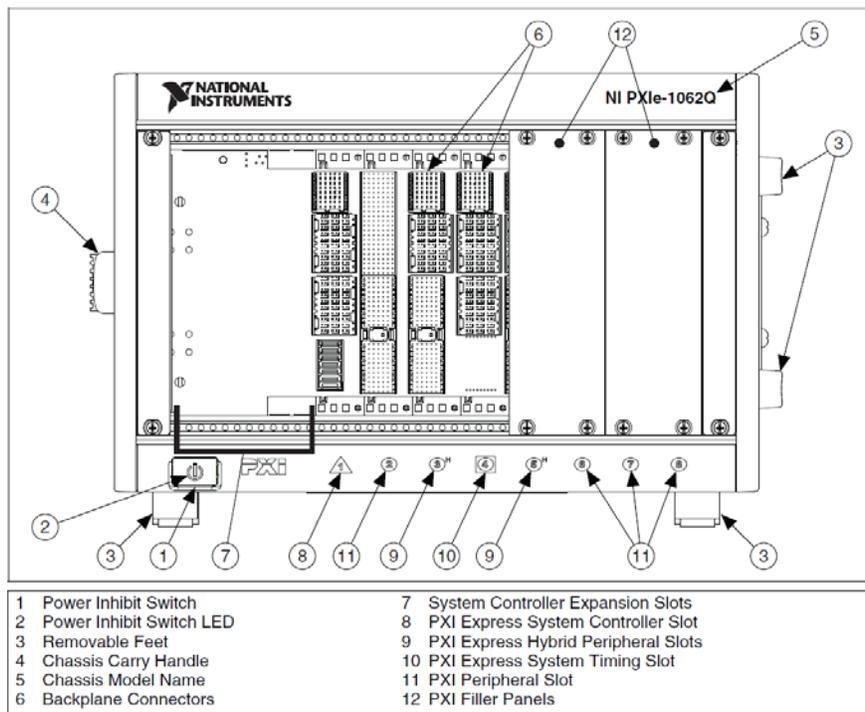


Figure 6: Generic and Detailed PXIe-1062Q Layout by Slot Number

Figure 6 shows a frontal view of the PXIe-1062Q with all DAQ functionality labeled. Figure 6 also shows the PXIe-1062Q with all installed NI hardware labeled by slot number. The following subsections will describe the hardware components installed in the PXIe-1062Q.

3.2.2. Slot 1: NI PXIe-8840 Embedded Controller

The PXIe-8840 is installed in slot 1 as shown in Figure 6 above. The PXIe-8840 is a Windows OS based embedded controller. It has been configured with a 2.7 GHz dual-core Intel i5 processor, 8 GB/s bandwidth and 4 GB/s slot bandwidth and 4 GB single channel 1600 MHz DDR3 RAM. LabVIEW software is installed on the PXIe-8840 which allows for communication with other NI hardware installed in the chassis. Signal generation therefore is based on computations performed by the PXIe-8840 via LabVIEW. The final output from the PMU under test will be collected by the PXIe-8840. The PXIe-8840 interfaces with SEL 3373 Phasor Data Concentrator through the network as shown in Figure 5. PMU settings software is also installed on the PXIe-8840, which makes it possible for test bench users to connect to the PMU under test via ethernet network connection.

3.2.3. Slot 2: NI PXI-6683H Timing and Synchronization Module

The PXI-6683H timing module allows for time synchronization between devices installed on the PXI and PXI Express slots. The PXI-6683H receives a satellite signal that provides absolute time to the test bench. The time information is received from the SEL 2407 GPS clock and transferred via IRIG-B format. The test bench user will be able to use the PXIe-6683 to generate triggers which are used by other NI hardware installed on the chassis. These triggers are programmed via LabVIEW software, and allow for coordination between devices for tasks such as outputting test signals at specific times.

3.2.4. Slot 3: NI PXIe-4300 with NI TB-4300B

The NI PXIe-4300 is a 16 bit, 8-channel, 250 kS/s, 300 V Ch-Ch Isolated PXI analog input module. Since input information may be required in further calculations this module allows test bench users to record high voltage analog inputs to the PMU for further analysis. The NI PXIe-4300 is interfaced via the NI TB-4300B terminal block. The PXIe-4300 is a high voltage analog input module and can be used to record signals from the amplification unit.

3.2.5. Slot 4: NI 6124 Multifunction DAQ with NI TB-2706 Terminal Block

The NI 6124 Multifunction DAQ has four analog input channels and 2 analog output channels. The NI 6124 interfaces to the PXIe-8840 via LabVIEW and can be used for recording analog data that is input to the PMU. The ADC resolution is 16 bit, and it samples at a rate of 4 MS/s per channel. Voltage range for the inputs ranges from ± 10 volts. This voltage range requires that PMU input signals be run to the module before signal amplification. Recording PMU input data may be of interest to the test bench user. The two analog outputs of the NI-6124 are not sufficient for PMU testing since three phase voltages would require three analog output channels.

3.2.6. Slot 5: NI PXIe-6738 Analog Output Module and SCB-68A I/O Block

The PXIe-6738 serves as the means for low voltage signal generation on this test bench. The PXIe-6738 analog output module has 32 analog output channels with 16 bit channel resolution. The sample rate for the PXIe-6738 is determined by how many analog output channels are being used. A maximum sample rate of 1 MS/s is available for up to eight analog outputs. Although the 1 MS/s sample rate is available, this sample rate is very high for the signals generated by this test bench via LabVIEW. Sample rate is configurable by the test bench user, as will be described in the signal generation section below. A key aspect of this module is

that it coordinates with the PXIe-6683H timing module. LabVIEW signal generation software will set signal start times as specified by the test bench user. Knowledge about the starting time of the input signal is essential to PMU characterization calculations.

3.3. SEL-2407 GPS IRIG-B Clock

The SEL-2407 is a satellite synchronized clock often used in PMU applications to provide the UTC time reference for synchrophasor estimation. It is important that the time associated with the NI signal generation and the timestamps made internally by the PMU are relative. Time shifts between input and output data will impact phase relationships in the final analysis. The SEL-2407 is accurate to ± 100 ns and requires a satellite antenna to receive GPS time. The antenna is located on the south east side of the main hall building and has an unobstructed view of the sky. This allows for uninterrupted satellite signals to be received by the clock. If the 'hold over' LED indicator on the front panel of the SEL 2407 is green, then satellite communication is locked. If the 'hold over' LED shows orange, then satellite communications have been lost and need to be re-established for accurate timing.

3.4. Tabor Electronics 9400-25 4 Channel Voltage Amplifier

Low voltage signal amplification for this test bench is performed via the Tabor Electronics 9400. Signal amplification is required to simulate signal levels that PMUs in the field receive. The Tabor Electronics 9400 voltage amplifier has a bandwidth of 0 Hz to 500 kHz. Four independent channels allow for 3 phase voltages to be amplified from the PXIe-6738 as well as a trigger signal for oscilloscope purposes. Channel monitors are located on the back of the amplifier so that oscilloscope measurements may be taken. Spec output voltages from the Tabor 9400 range from ± 200 volts. The Tabor 9400 specification limit for output current is 50 mA. This is an important specification when considering that the amplifier is directly interfaced with

the PMU under test. If the front plate of the PMU is removed the inputs to the device feed directly into instrument transformers. Transformers are inductive loads. Consider the equation for the impedance of an inductor shown in equation (14) above. This relationship shows that as frequency, ω , decreases so does the corresponding impedance. Steady state sinusoid testing has shown that input signals should not go below 35 Hz. Going below this threshold will cause the PMU unit under test to draw over the 50 mA limit for the Tabor 9400. The input impedance for the Tabor 9400 is $1\text{M}\Omega$, so interfacing the PXIe-6738 to the Tabor 9400 is not an issue. Specs for the Tabor 9400 list it as a fixed gain amplifier of 50x, this however is not the gain used for this test bench. A special order was made for a gain of 25x. The Tabor 9400 requires a 2.5-amp fuse to operate. The fuse serves as overcurrent protection for the amplifier and is imbedded in a screw terminal on the back of the unit.

3.5. SEL 3373 Phasor Data Concentrator

The SEL 3373 Phasor Data Concentrator (PDC) allows the test bench user to archive and collect PMU output data. The SEL 3373 is connected to a network switch and is interfaced with via PDC Assistant software on the PXIe-8840. PMU data can be archived in .csv format which is ideal for test analysis in Matlab. The SEL 3373 formats PMU data as expressed in the IEEE C37.118 standard and is configured with 60 GB of memory for archiving. Options in PDC Assistant allow for adjustment to the message per second rate of the PMU output data. This rate will determine file size but will likely not be a concern for test signals that last under ten minutes. The message per second rate for testing is set to the max rate of 60 messages per second to increase the resolution of PMU output for analysis.

3.6. Test Bench: Software

Three main software packages were utilized for PMU testing; Matlab, LabVIEW, and SEL PDC Assistant. Signal generation is fundamental to PMU testing, and there are many methods that can accomplish this task. Besides generating signals, the test bench needed tools that would make it possible to parse PMU data and to analyze test results. The various aspects of test bench software will be covered in the following sections.

3.6.1. Matlab

Matlab is an Integrated Development Environment (IDE) which makes use of its own script-based language. Matlab also has many commonly used functions pre-written for the user which are available through various tool-box add-ons to the Matlab IDE. With Matlab it is possible to write precisely the signals of interest, and to save them to a .csv file. Matlab software was written for data parsing and analysis as well. Section 4, section 5, and section 6 will cover PMU generic testing, PMU step testing, and PMU phase modulation testing respectively. The Matlab software developed for this testing will be covered step by step in those sections to bring its functionality full circle for future test bench users. All Matlab software files are available on the appended drive. All software is commented extensively for future test bench users.

3.6.2. National Instruments LabVIEW

Software developed in LabVIEW serves as the hub for signal generation and hardware communications on this test bench. LabVIEW software is used to interface to test bench hardware, such as the PXIe-6738 analog output module or the PXIe-6124 analog input module. This software makes use of DAQmx VIs that coordinate timing between the PXIe-6683H timing module and the PXIe-6738 analog output module. LabVIEW software was also written that would generate full phase modulation signals without the need for Matlab generated .csv files.

Test bench users have two options for signal generation; Matlab and LabVIEW. These two methods will be covered in more detail in section 4.2.1. All LabVIEW software files are available on the appended drive and are commented extensively for future test bench users.

3.7. Test Bench Signals

Design of the PMU test bench was tied directly to the dynamic signals that would be used for testing. Mathematical signal descriptions serve as the basis for signal code written in either Matlab or LabVIEW. Three phase signals A/B/C were used for PMU testing. The base form for the A phase signal is given in (15) below.

$$v(t) = A_o + A_1(t) \cos(2\pi f_o t + \phi(t)) + \sum_{i=2}^{\infty} A_i \cos(i2\pi f_o t + i\phi(t) + \phi_{oi}) + n_m(t) \quad (15)$$

where

$$A_1(t) = A_{1,0}(1 + k_x \cos(2\pi f_m t)) \quad (16)$$

$$\phi(t) = \phi_0 + 2\pi\Delta f_r(t - t_r)u(t - t_r) + k_a \cos(2\pi f_m t) \quad (17)$$

where A_o is DC offset, $A_1(t)$ is the amplitude of the fundamental as a function of time, $\phi(t)$ is the phase angle as a function of time, A_i is the amplitude of harmonics, f_o is the fundamental frequency, k_x amplitude modulation factor, k_a is the phase angle modulation factor, Δf_r is a constant frequency for phase ramping, f_m is phase modulation frequency, and $u(t - t_r)$ is a unit step function. The summation terms in (15) account for harmonics. Equation (16) accounts for magnitude step tests and amplitude modulation. Equation (17) accounts for phase ramp (frequency step) tests and phase modulation tests. For instance, if phase modulation signals were required for testing the user would adjust the phase modulation magnitude and frequency terms. Signal descriptions and settings are available for magnitude step tests and phase modulation tests described in section 5 and section 6 respectively.

3.7.1. Test Bench Signals – ‘GeneralSigGen.m’ Matlab Software

The signals described in (15), (16) and (17) will be expanded upon here into their testing forms and corresponding Matlab script form. The testing form for equation (15) is as follows

$$V_a(t) = [A_o + A_1(t)\cos(2\pi f_o t + \phi(t))][u(t - tStart) - u(t - tEnd)] \quad (18)$$

This signal subtracts harmonic content from (15) and adds unit step functions to turn the signal on and off at times tStart and tEnd. The Matlab script form of (18) is shown below with the all three phases present for testing.

```
Aphase = AmpOffset+...
        A1.*cos(2*pi*fo*t+phi).*...
        (stepfun(t,tStart)-stepfun(t,tEnd));

Bphase = AmpOffset+...
        A1.*cos(2*pi*fo*t-2*pi/3+phi).*...
        (stepfun(t,tStart)-stepfun(t,tEnd));

Cphase = AmpOffset+...
        A1.*cos(2*pi*fo*t+2*pi/3+phi).*...
        (stepfun(t,tStart)-stepfun(t,tEnd));
```

Figure 7: A/B/C Phase Base Signal Matlab Script

The testing form of equation (16) is shown below.

$$A_1(t) = A_{1,0} \left[1 + k_x \cos(2\pi f_m t) \left(u(t - t_{AMOn}) - u(t - t_{AMOff}) \right) + A_{step} \left(u(t - t_{AstepOn}) - u(t - t_{AstepOff}) \right) \right] \quad (19)$$

The Matlab script for equation (19) is as follows.

```
A1 = SigMag*(1+Amp.Mod.Mag*cos(2*pi*Amp.Mod.Freq*t).*...
        (stepfun(t,Amp.Mod.tOn)-stepfun(t,Amp.Mod.tOff))+...
        Amp.Step.Mag.*...
        (stepfun(t,Amp.Step.tOn)-stepfun(t,Amp.Step.tOff)));
```

Figure 8: Matlab Script for Equation 19

The testing form for (17) is as follows.

$$\phi(t) = \phi_o + 2\pi\Delta f_r(t - t_r)[u(t - t_r) - u(t - t_{PhaseRampOff})] + k_a \cos(2\pi f_m t)[u(t - t_{PMOn}) - u(t - t_{PMOff})] \quad (20)$$

Equation (20) includes step functions that turn on/off phase ramping and phase modulation signals. The script form for (20) is shown below.

```
phi = phiOffset+...
      2*pi*Phase.Ramp.Freq*(t-Phase.Ramp.tOn) .* ...
      (stepfun(t,Phase.Ramp.tOn)-stepfun(t,Phase.Ramp.tOff))+...
      Phase.Mod.Mag*cos(2*pi*Phase.Mod.Freq*t) .* ...
      (stepfun(t,Phase.Mod.tOn)-stepfun(t,Phase.Mod.tOff));
```

Figure 9: Matlab Script for Equation 20

The Matlab code in Figure 9 above makes use of structures to store signal parameters. This is an ideal way to organize large sets of variables that follow a hierarchy or family tree format. By setting signals up in a generic format, test bench users can make use of one Matlab script, ‘GeneralSigGen.m’, which can be used for many types of signal testing. Adjusting the unit step functions will give users the ability to turn certain aspects of the signal on and off when desired.

3.8. Test Bench Timing

To obtain the frequency response characteristic of the PMU under test the device response to sinusoids of varying frequencies (one at a time) will be computed. The calculation used to obtain the frequency response will be done using sampled output data from the PMU and knowledge about the input signal and its start time. Consider Figure 10 below and the relationship of the output frequency over the input frequency.

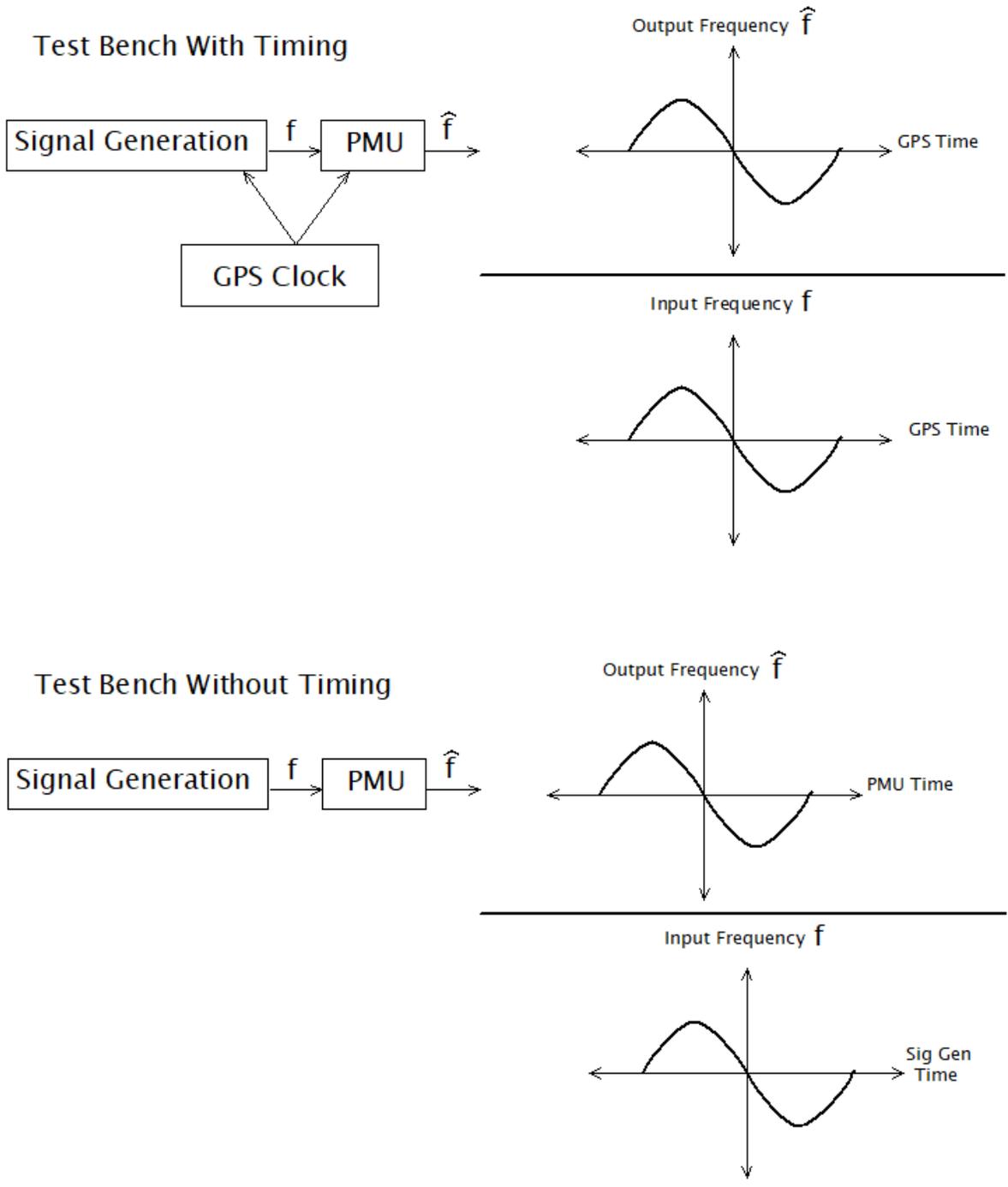


Figure 10: Test Bench with Timing Versus Test Bench With Out Timing

Without test bench timing the input and output data will be on time axes that are not relative. This will then cause error in phase when computing the response of the PMU. With

proper test bench timing the input and output data are time synchronized meaning they are both on the same time axis. Test bench timing, as described in Figure 10, is critical to the analysis performed in section 6.

3.9. Test Bench Accuracy

Considering that the overlaying objective of this project was to determine PMU frequency estimation in real time, PMU testing for this project was time sensitive and frequency sensitive. Accuracy of the test bench with respect to signal start time and analog output sampling frequency were the main factors affecting the integrity of PMU test results. Testing was performed on the test bench to check the accuracy of specified signal start times and the accuracy of the PXIe-6738 analog output module sampling frequency. The following subsections highlight test bench accuracy testing and results.

3.9.1. Oscilloscope Clock Test

The oscilloscope clock test was designed to ensure the functionality of the LabVIEW signal generation software and to test the accuracy of signal start time. This test verifies that the test bench software and hardware is correctly timing the input signal shown in Figure 10. To accomplish this a trigger signal was developed in Matlab and then transferred to LabVIEW for generation. The signal was comprised of a 5 volt step which occurs 10 seconds after the user specified start time. The SEL 2407 provides a PPS function that is available at BNC output 3 located on the back of the clock. This PPS function provides absolute time for the test and is monitored on a Tektronix oscilloscope. Figure 11 below illustrates this test.

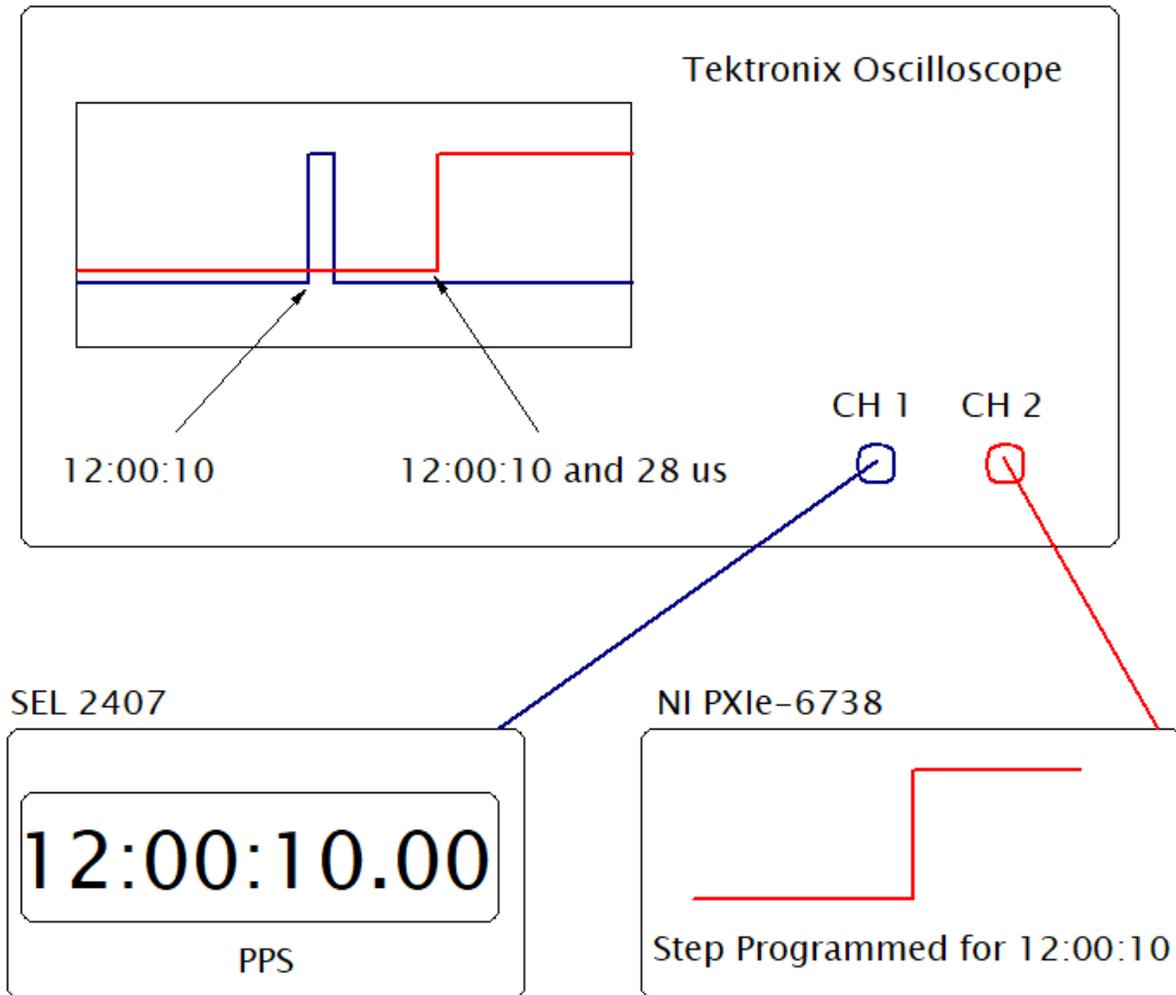


Figure 11: Oscilloscope Clock Test – Set Up

The Results from this test show that analog signals from the NI PXIe-6738 start at approximately the specified time. A delay in the signal does occur and is on the order of microseconds. For a single step signal that is 30 seconds long and sampled at 40 kHz the delay is 28 us. Furthermore, if this step signal is run over and over, the delay time is fixed at 28 us on each run. Repeated testing was performed to ensure that this delay was constant. It can be concluded that analog signal start time is a function of memory, or the amount of space the signal requires on the memory buffer. Input signal start time accuracy must be considered when running certain tests as described in Figure 10. Results of the oscilloscope clock test reveal two

important factors. The first is that the LabVIEW signal generation code was written correctly and is functioning as expected. The second is that the NI hardware has limitations that need to be considered when designing certain tests.

3.9.2. NI PXIe-6738 Testing and Analysis Using PMU Phase Angle Data

Phase modulation testing will be covered in depth in section 6, but as a precursor to that section it is important to consider the sampling rate of the PXIe-6738 analog output module. The phase modulation signals used for these tests are carefully designed around the frequency and modulation frequency terms in (15) and (17). It is imperative that the sampling of the PXIe-6738 be known. Recall the relationship between sample frequency and the time between sample points.

$$f_s = \frac{1}{T_s} \quad (21)$$

where f_s is the sampling frequency in Hz, and T_s is the time between sample points. If a steady state 60 Hz sinusoid signal has a sampling frequency of 1 kHz, then the time between samples will be 1 ms. Further, assume that the time vector for this signal lasts for 1 second. If this signal was produced in Matlab and then saved as a .csv, the file would contain a total of 1000 data points. When LabVIEW is used to load a .csv it only knows that there is 1000 points in the file. The sampling frequency must be specified in the LabVIEW signal generation software as 1 kHz. Now consider what happens when the signal is sampled by an analog output module like the PXIe-6738. If the module does not produce the signal at exactly the specified 1 kHz sampling frequency, then the time between sample points will not be 1 ms. If the sample frequency of the analog output module is 0.999 kHz then the time between samples will be 1.001 ms, and the time duration of the signal will be 1.001 seconds. This in effect takes the signal that is supposed to be

60 Hz for one second and stretches it to 1.001 seconds. The intended 60 Hz sinusoid will then be stretched, and the signal frequency will be as follows.

$$f_{act} = f_{prog} \left[\frac{f_{samp_act}}{f_{samp_prog}} \right] = (60 \text{ Hz}) \left[\frac{999 \text{ Hz}}{1000 \text{ Hz}} \right] = 59.994 \text{ Hz} \quad (22)$$

Where f_{act} is the actual signal frequency, f_{prog} is the programmed signal frequency, f_{samp_act} is the actual sample frequency, and f_{samp_prog} is the programmed sample frequency. The NI PXIe-6738 analog output module can sample at up to 1 MHz. This sample rate is excessive for the testing described here in. The sample frequency for phase modulation test signals was set to 10 kHz. The data sheet for the PXIe-6738 specifies a sampling accuracy of 50 parts per million of the sampling frequency. This means that for a sample rate of 10 kHz, the PXIe-6738 may in fact be sampling at $10 \text{ kHz} \pm 0.5 \text{ Hz}$. With phase modulation signals as in (17), it is important to understand the actual sampling frequency that is being performed by the PXIe-6738.

Mathematical analysis on PMU output data may be performed to obtain output signal frequency and the sampling frequency of the PXIe-6738. A test signal containing 3 phase signals at 60 Hz was written via Matlab and is shown below. The signal steps on at 20 seconds and off at 80 seconds. The signal sampling frequency was programmed to be 40 kHz.

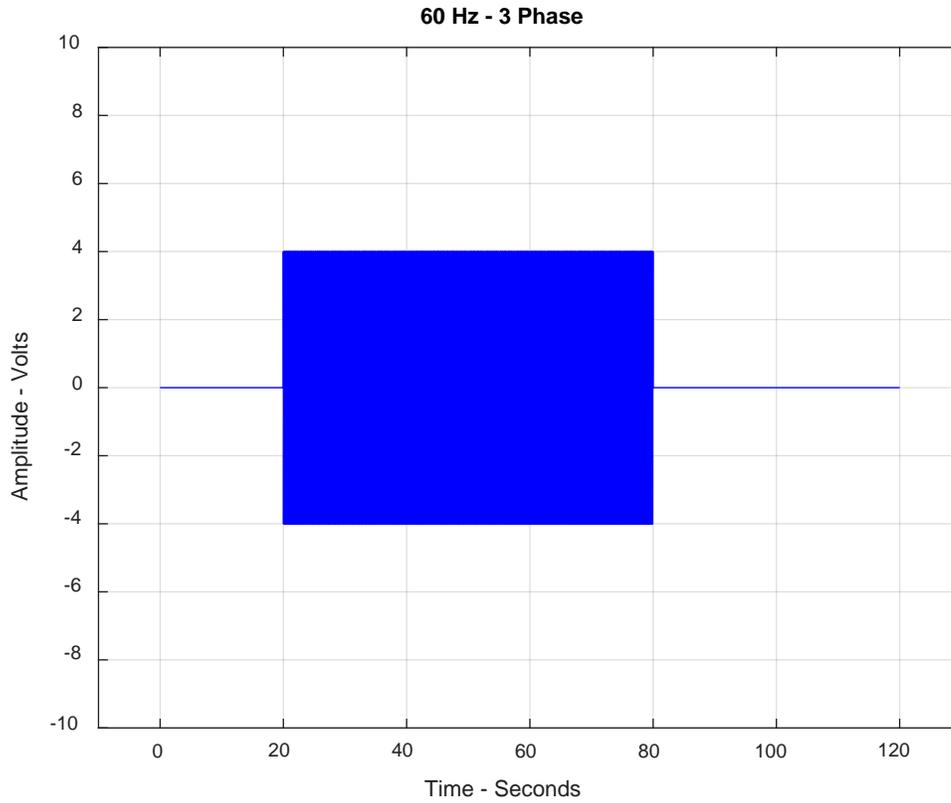


Figure 12: Imperfect Sample Rate – Frequency Test

This signal was then input to the PMU under test and PMU output data collected. The A phase angle data was then extracted and analyzed. Taking the phase angle data and plotting it against time yields the following plot.

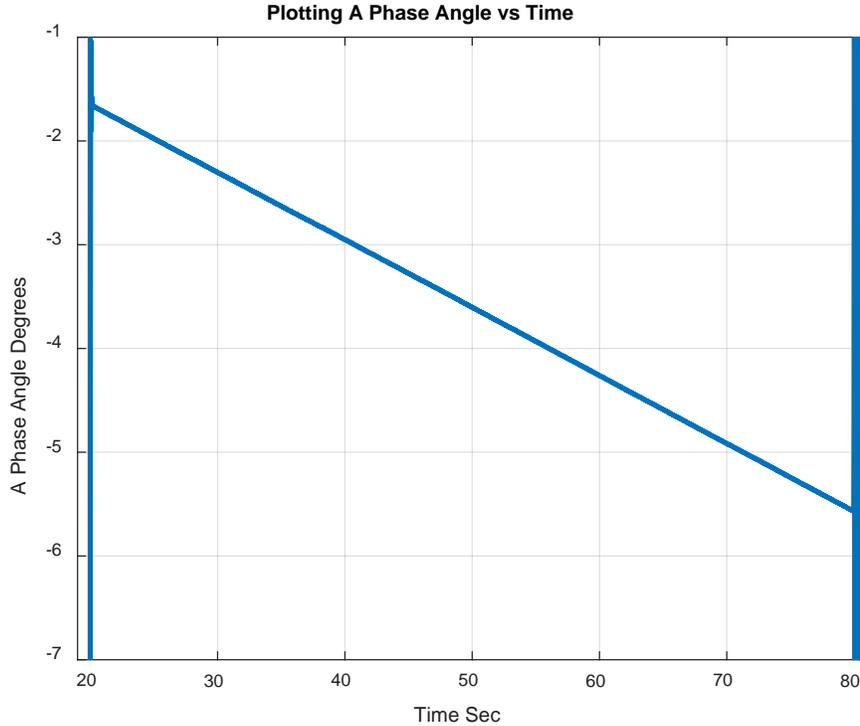


Figure 13: Imperfect Sample Rate – Phase Angle

Now considering the following relationship for frequency.

$$f(t) = f_o + \frac{d}{dt} \left[\frac{\phi(t)}{360^\circ} \right] = 60 \text{ Hz} + \left(\frac{-0.065315 \frac{\text{degrees}}{\text{sec}}}{360^\circ} \right) = 59.999819 \text{ Hz} \quad (23)$$

The corresponding sample rate can be determined by rearranging terms in (23), yielding an actual sample rate of 39999.879 Hz. The time between samples for a 40 kHz sampling rate is 25 us, and the time between samples for a sampling frequency of 39999.879 Hz is 25.0000756 us. This yields an error of 75.6 ps. This method for determining sample frequency error relies upon the PMU for phase angle data. Another test was designed to determine PXIe-6738 sample rate error as a check to make sure this test is accurate.

3.9.3. NI PXIe-6738 Testing and Analysis Using PXIe-4300

For certain hardware checks, such as testing for the true sampling frequency of the PXIe-6738 analog output module, it may be necessary to record analog input data. This can be done using the PXIe-4300 analog input module in conjunction with LabVIEW software written for this purpose. The LabVIEW software ‘Analog Input Reader.vi’ is available on the appended drive to this thesis. Running this software will open a front panel display as shown in Figure 14 below.

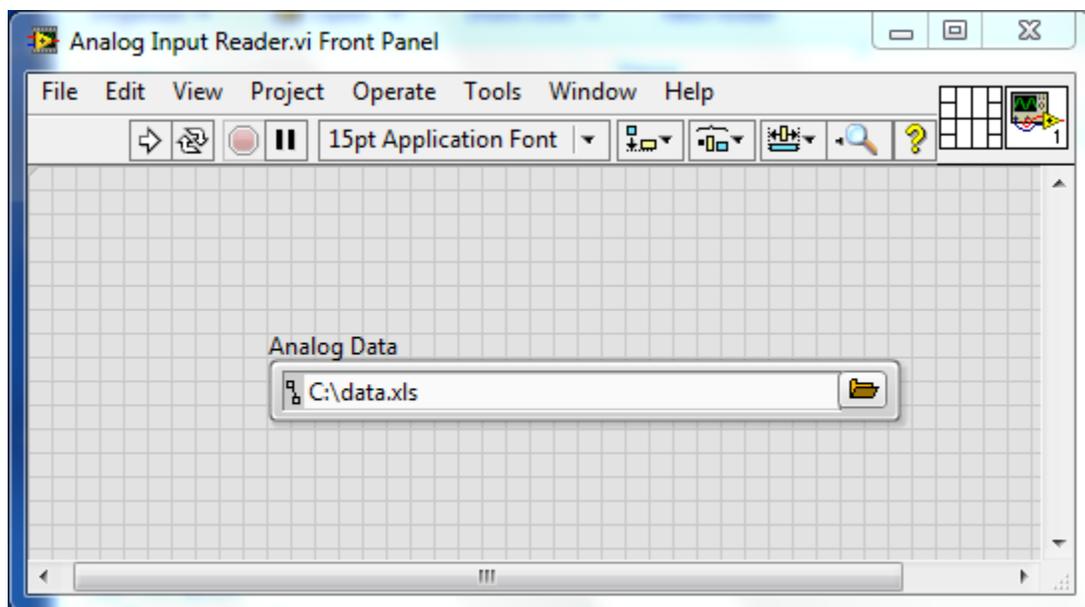


Figure 14: Analog Reader Front Panel

Clicking the folder icon will open a saving option in windows file explorer. Files must be saved with the .csv extension in the file name for this software to work. Output signals from the Tabor 9400 then need to be wired to the NI 4300B terminal block, which then connects to the PXIe-4300 analog input module. The analog inputs in the ‘Analog Input Reader.vi’ software are configured as 300V_Isolated/ai0-3. These inputs must match the physical connections to the terminal block input. Clicking ‘ctrl+E’ on the front panel of the LabVIEW software will reveal

the code in the background. Figure 15 shows the analog input configuration that must match the connections on the NI 4300B terminal block.

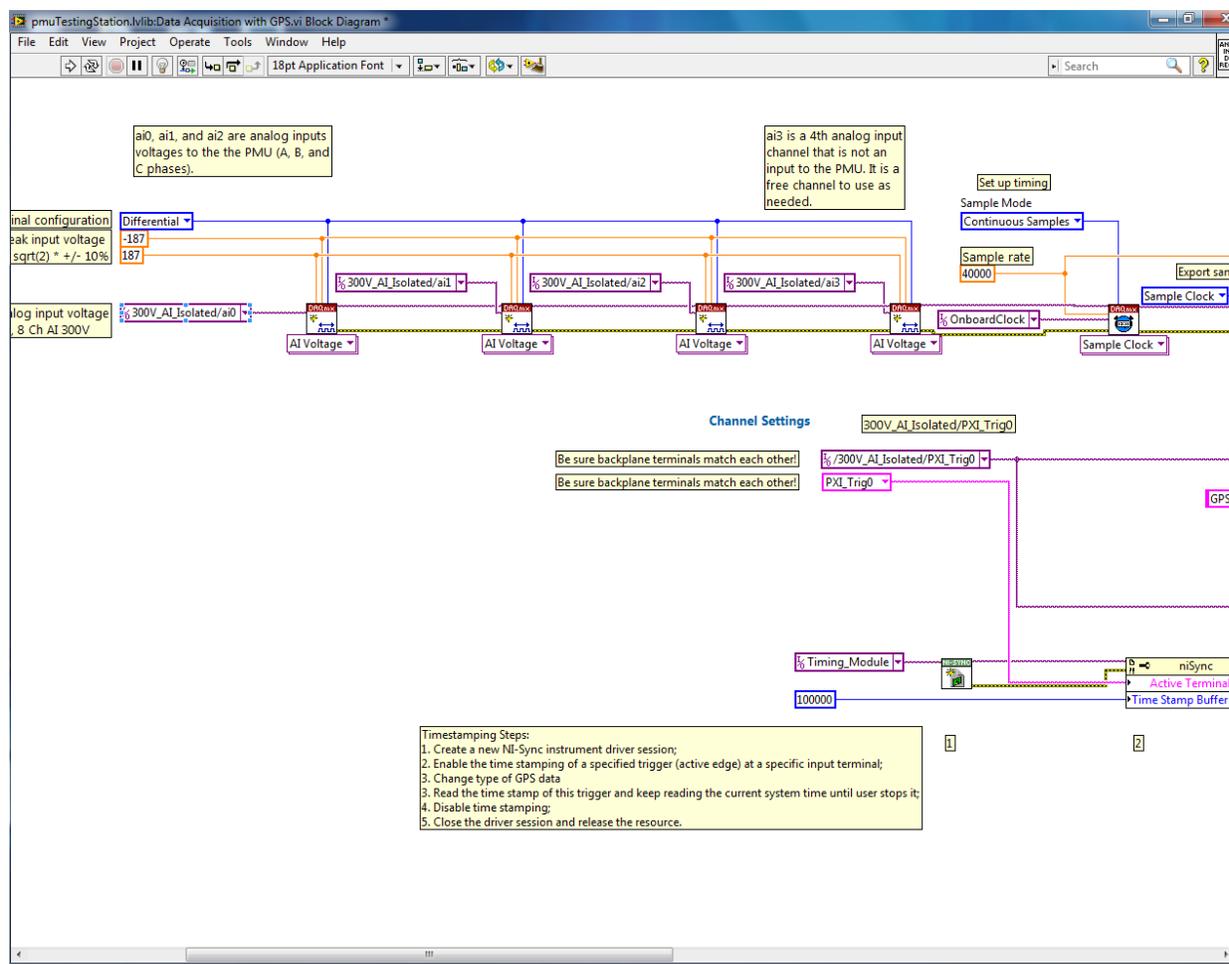


Figure 15: 300V_AI_Isolated/ai0/1/2/3

After wiring has been completed and the terminal block is connected to the PXIe-4300, analog input signals can then be recorded by running the ‘Analog Input Reader.vi’ software. These signals will be saved as .csv files and will need to be parsed using the software described in section 4.2.3.

To test the sampling frequency of the PXIe-6738 the same three phase signals shown in Figure 12 above were input to the PXIe-4300. This data was recorded and analyzed in Matlab to determine the frequency of the three phase sinusoids. This analysis yields frequencies for the

three phase signals of 59.999807 Hz. This result compares closely to the PMU phase angle data which yielded a frequency of 59.999819 Hz. The data for these two tests are summarized in Table I below.

Table I: Results - Testing for True Sampling Frequency of the PXIe-6738 Analog Output Module

	Testing PXIe-6738 Sample Frequency Using PMU Phase Data	Testing PXIe-6738 Sample Frequency Using PXIe-4300 Analog Input Data
Programed Frequency	60 Hz	60 Hz
Actual Frequency	59.999819 Hz	59.999807 Hz
Programed Sample Frequency	40 kHz	40 kHz
Actual Sample Frequency	39.9999879 kHz	39.999871 kHz
Programed Time Between Sample Points	25 us	25 us
Actual Time Between Sample Points	25.0000756 us	25.0000806 us
Time Error Between Sample Points	75.6 ps	80.6 ps

The results from both tests yield results which are very close. The time error between sampling points seems small but is enough to affect the results of phase modulation testing. Consider the phase plot in Figure 13. If sampling frequency of the PXIe-6738 were perfect this plot would show a horizontal line for the phase angle. This assumes a steady state frequency of 60 Hz. The negative slope shows that the frequency is not at its programmed 60 Hz, and that this error in frequency will accumulate phase error with time. This analysis must be performed when running frequency sensitive tests and will be highlighted again in section 6 when phase modulation testing is covered.

4. PMU Testing

A testing procedure relative to the test bench described in section 3 will be given here. This procedure is generic in that it is not dependent on any specific type of test. The PMU under test in this section will be treated as a black box. At the end of this procedure the test bench user will have data ready for analysis in the form of .csv files.

4.1. Pre-testing Procedures and Precautionary Steps

To ensure the integrity of PMU testing and the safety of the test bench a pre-testing procedure is covered here. The Tabor 9400 provides amplification of signals from the PXIe-6738. The amplified signals are then input to the PMU under test. If the face plate to the PMU is removed it is likely that the inputs will be fed into instrument transformers that will take high voltage signals down to a low level for processing. As described above the input impedance is a function of frequency, and low frequencies will make the PMU look like a short to the Tabor 9400. Another important consideration is how the PXIe-6738 deals with the end values of signals. Typically, three phase signals will be used to test PMUs. If a test signal ends at a value other than zero, the PXIe-6738 will continue to produce the final values of the three phase signal end points at DC. Consider Figure 16 below.

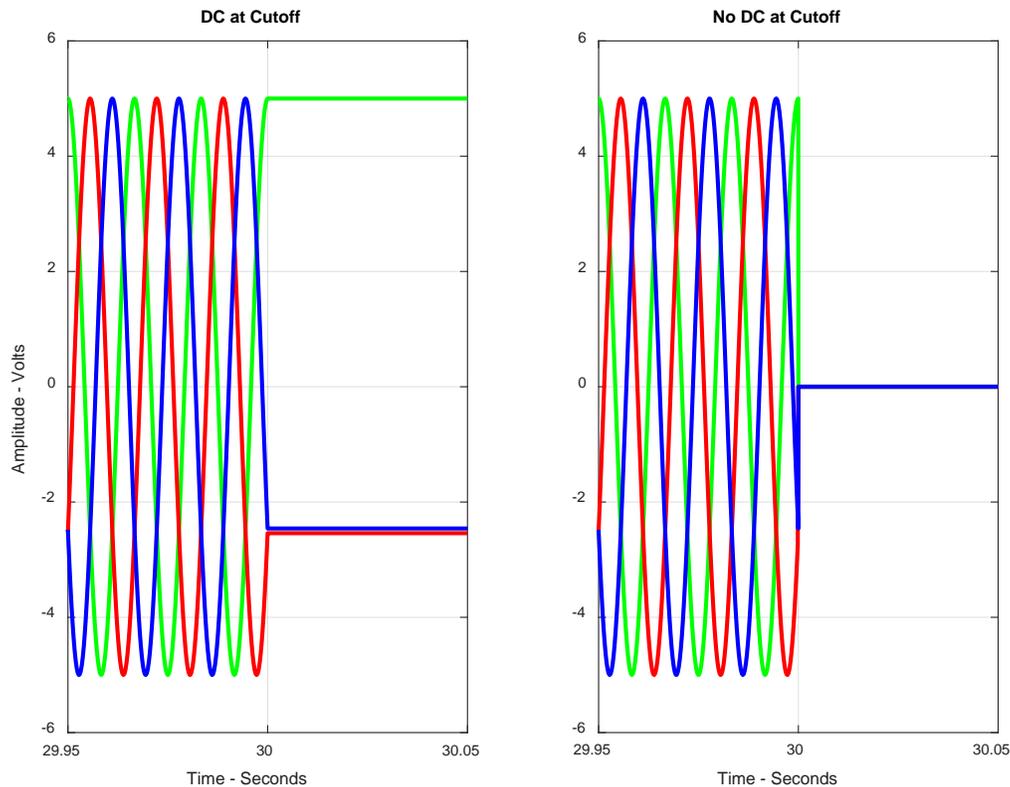


Figure 16: Zeroing Test Signals

PMU input signals need to be written so that they end at zero volts, otherwise the ending value of the three phase voltages will continue on as a DC signal. Considering Z_L in (14), DC signals will make the PMU look like a short to the Tabor 9400 and will then damage the unit as it outputs current much greater than its specified max current output. Signals should be checked thoroughly to ensure that frequency content does not go below 35 Hz. It is a good idea to check that signal performance goes as expected on the oscilloscope before running tests on the PMU. If the sampling rate is not correctly defined between Matlab and LabVIEW software the test signal will not behave as expected. This may likely damage the Tabor 9400 if the PMU input signals drop too low in frequency. Another precautionary procedure is to monitor signals input to the PMU on an oscilloscope while testing. A kill switch is located on the front of the Tabor 9400 if something unexpected occurs.

4.2. Testing Procedure

Test procedures described here will be relative to the test bench described in section 3 above. Procedures for PMU testing will reference the test bench software written for this project. Software files will be available on the appended drive to this thesis. The software is commented extensively to aid new users in understanding the test bench software. Portions of this procedure will require the user to alter certain aspects of the software code, so it is important to keep the original versions of the software separate from altered versions. Unaltered software will be a backup if the test bench software gets a bug.

4.2.1. Signal Generation Procedures

Signal generation is accomplished one of two ways. Figure 17 below illustrates a basic block diagram of the two signal generation methods.

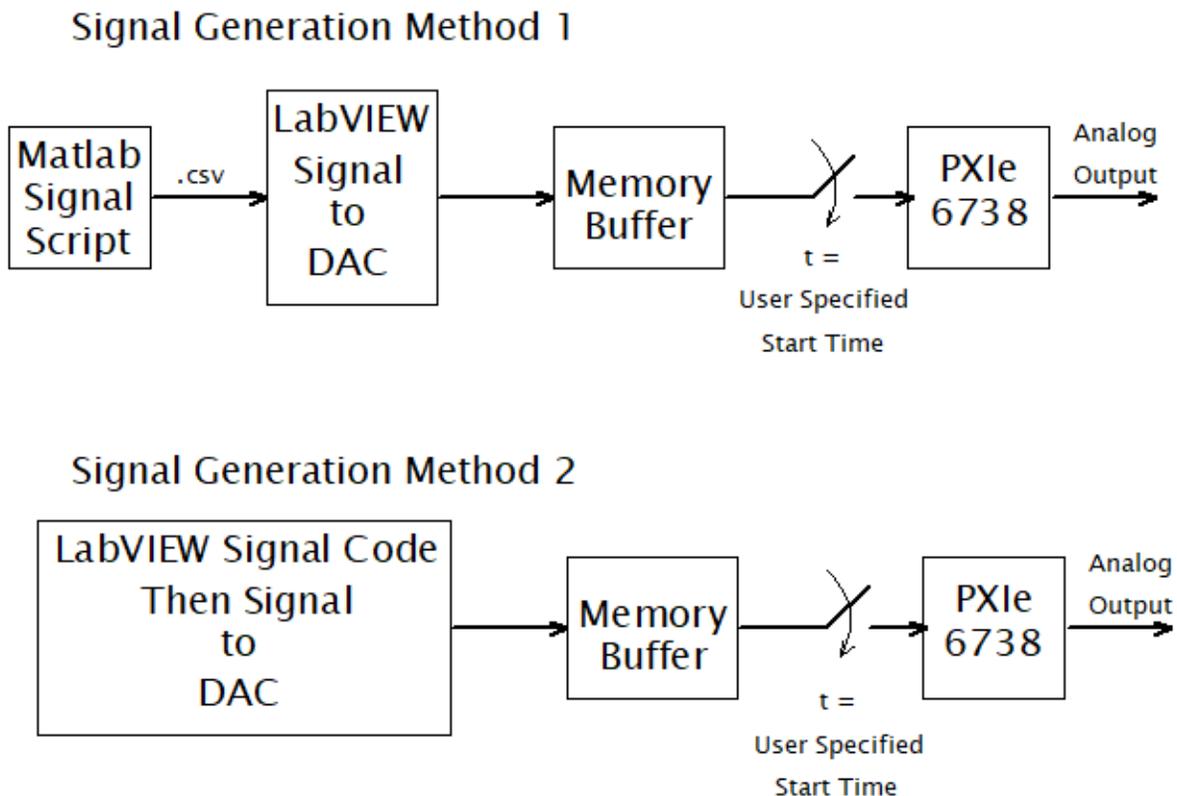


Figure 17: PXIe 6738 – Two Methods for Signal Generation

In both method 1 and method 2 LabVIEW is used to coordinate with the PXIe-6683H timing module and then to communicate with the PXIe-6738 hardware which in turn produces the analog signal. Method 1 for generating signals uses Matlab to produce .csv files containing the test input signal. Method 1 makes use of 'GeneralSigGen.m' Matlab software described in section 3.7.1. Running the signal generation script will produce the .csv file containing a test signal. The .csv file will contain a matrix with the voltage signals stored as rows. Next, the LabVIEW file 'GenThreePhase.vi' should be opened. The front panel for this software is shown below.

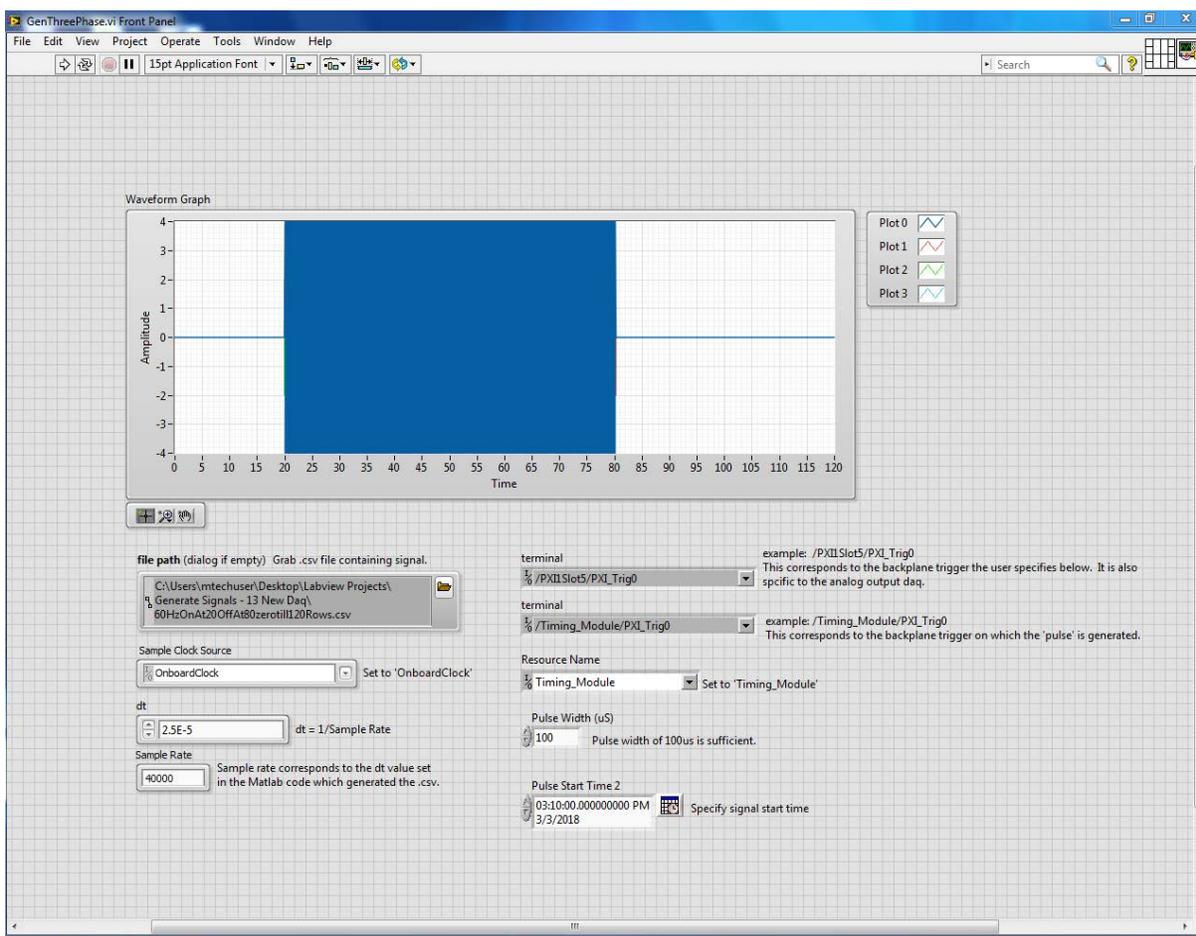


Figure 18: PXIe 6738 – GenThreePhase.vi Software Front Panel

Opening the 'GenThreePhase.vi' signal generation software will bring up the front panel shown above. To run the Matlab generated .csv signal press the folder icon beneath the 'file path' directory. Windows file explorer will allow users to navigate to the signal .csv file of interest. The sample rate that was used to generate the .csv in Matlab must also be set on the front panel at 'Sample Rate', as well as setting 'dt'. With the .csv file selected, it is then necessary to set a signal start time. This is done by setting the appropriate month/day/year and time at 'Pulse Start Time' on the front panel. All other options on the front panel are set up automatically for the user. To start the software simply hit the 'run' icon located on the top tool bar. The 'Waveform Graph' plots the test signal against time. Upon completion of the test signal, or once the signal zeros out the software should then be stopped by hitting the 'stop' icon also located on the top tool bar. To ensure signal zeroing has occurred it is best to visually check signals on the test bench oscilloscope.

The drawback to signal generation method 1 is that large files, or test signals with a high sample rate and a long-time duration, will take considerable time to load into the LabVIEW software. The .csv files for these signals will be on the order of Gigabytes in memory. Another aspect to consider is that these large files will affect the signal start time and delay time described in the oscilloscope clock test above. Larger files will cause a longer time delay with respect to the specified signal start time. Phase modulation testing as described in section 6 requires a signal that is 8 minutes and 45 seconds long sampled at 10 kHz to be run. The start time delay associated with this signal is approximately 500 μ s. Time delay can be corrected for in the analysis software, but another signal generation option might be a more efficient use of test bench users time. Method 2 for signal generation cuts Matlab out of the process. As a result, loading times are cut significantly and time delay relative to specified signal start time is reduced

to a negligible amount. The possible draw back to this method is that unlike scripting languages the test signal to be produced must be ‘written’ in LabVIEW. This method for signal generation was employed for phase modulation testing which is covered in section 6 below. The signals and signal generation software developed for these tests will be covered in section 6.

4.2.2. PMU Data Acquisition

PMU data is acquired via the SEL 3373 Phasor Data Concentrator (PDC) and its accompanying software PDC Assistant. As in Figure 5 the PXIe-8840 and SEL 3373 are connected to a network switch. The SEL 3373 archives PMU data specified by the user through PDC Assistant. To obtain PMU test data first start the PXIe-8840 and open PDC Assistant in the SEL applications folder. Right click on the PDC Assistant icon and select run as administrator. The front panel for PDC Assistant will require you to connect to the SEL 3373. Figure 19 shows the connection screen.

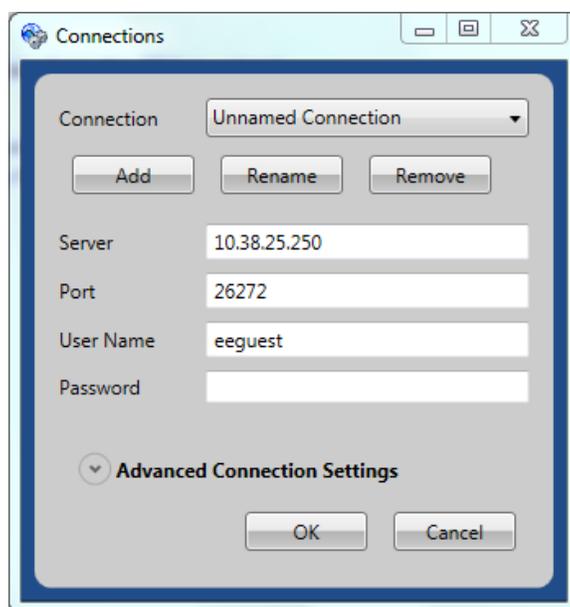


Figure 19: PDC Assistant – Connecting to the SEL 3373

The IP address, port number, user name and password specific to this test bench are located on the front panel of the SEL 3373 and are listed in Figure 19 above. The password for

the SEL 3373 is 'EE4mtech\$'. After connecting to the SEL 3373, an input for the PMU under test will need to be configured.

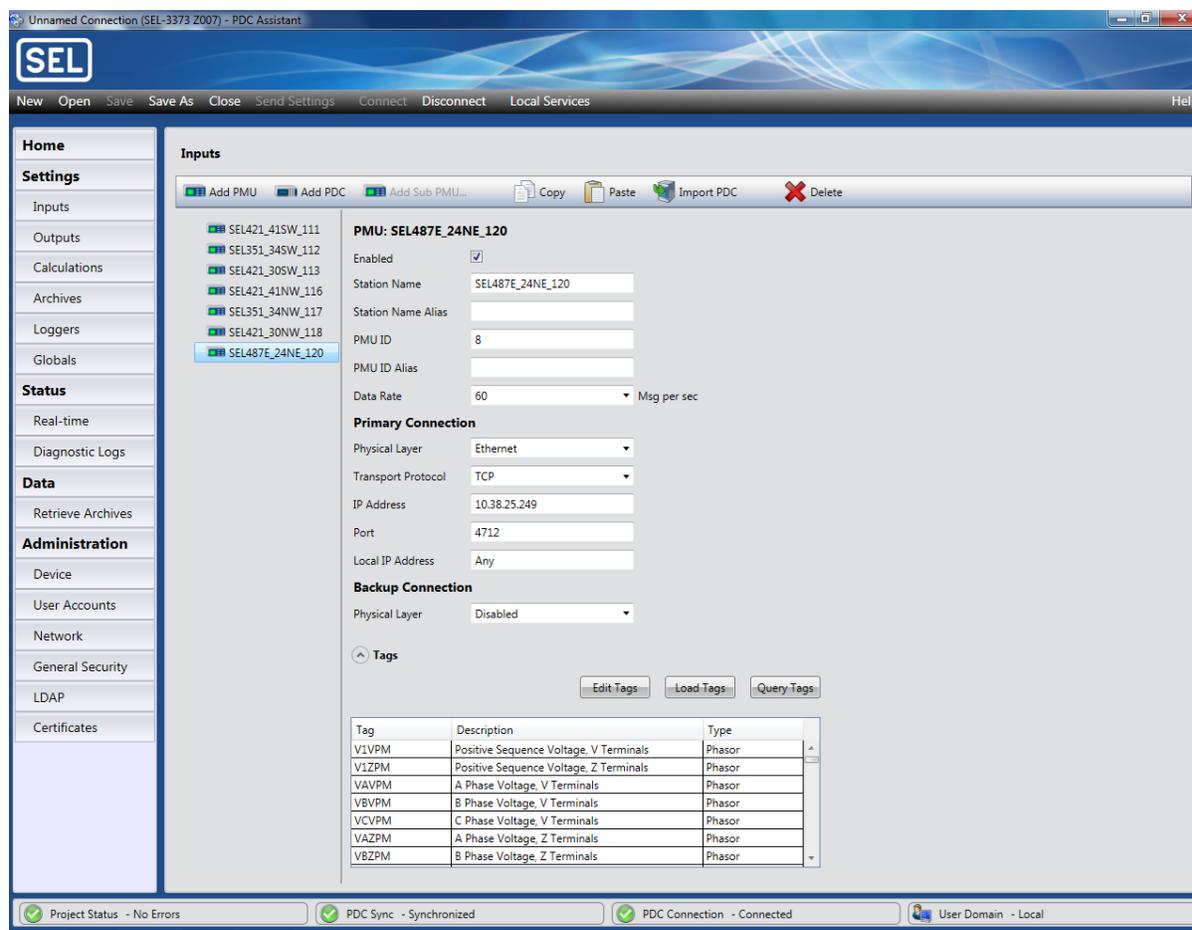


Figure 20: PDC Assistant – Configure Input

To add an input, click 'Add PMU'. Input parameters will need to match PMU settings such as station name, PMU ID, PMU IP address, port number and data rate. For example, if the PMU settings specify the data rate to be 60 samples per second, and the data rate set in PDC assistant is 30 samples per second the software will give an error which can be viewed in the 'Diagnostic Logs' tab. Make sure the input is enabled as shown in Figure 20. Any PMU data to be used for analysis needs to be set in the 'Tags' section. Clicking the 'Load Tags' button will bring up a selection screen with generic files that will list PMUs by model. Selecting the PMU

model for the unit under test will populate the tags with all PMU data available for that model. After setting the input up, the next step is to configure the PMU archive data. Clicking on the ‘Archive’ button will display the following screen.

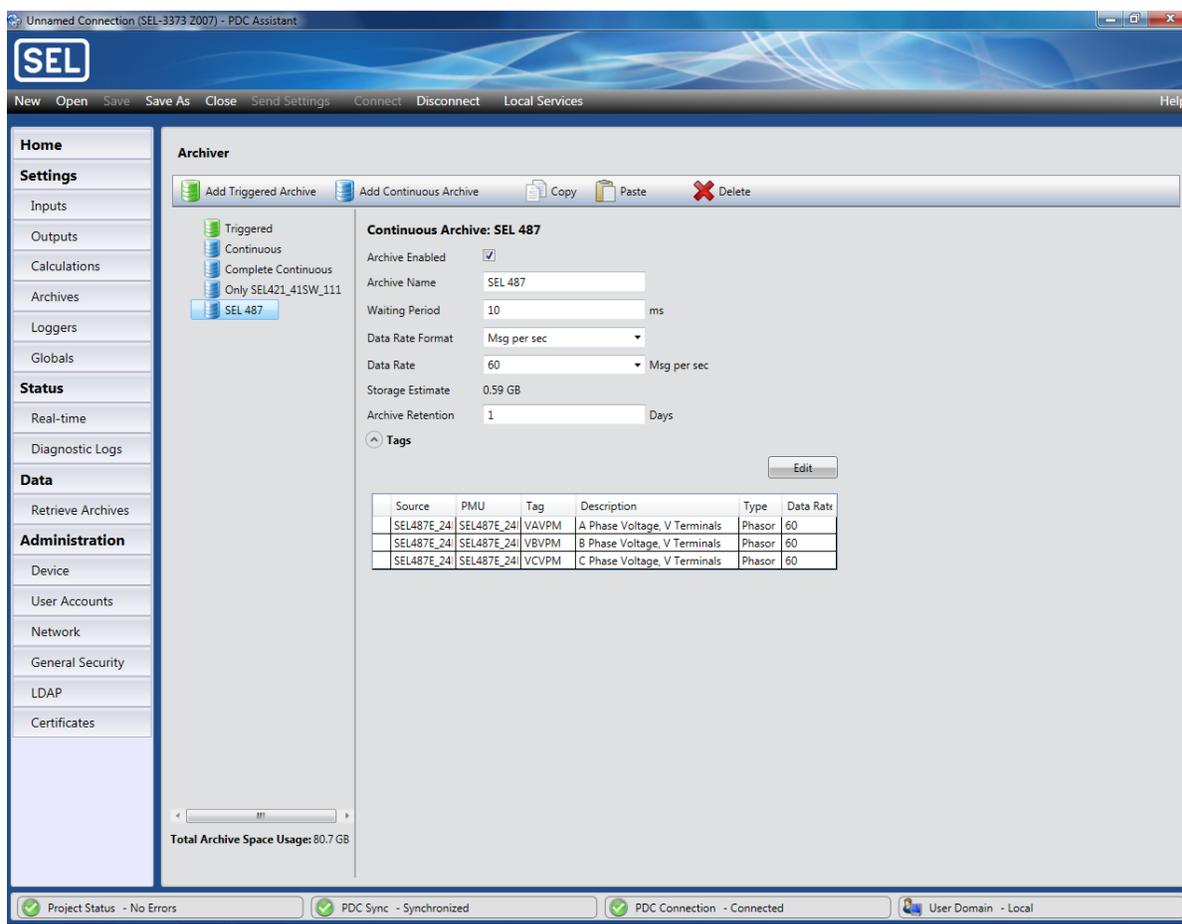


Figure 21: Assistant – Configure PMU Archive

Clicking the ‘Add Continuous Archive’ button will display all available PMUs. Selecting the PMU under test will bring up the options shown in Figure 21 above. Be sure to enable archiving, and to set the message rate accordingly. The ‘Edit’ button under the tags option will bring up all available PMU data for the unit under test. Select all PMU data to be used for test analysis. Now that the PMU archive has been set up click ‘Local Services’ and select ‘Archive Collection Service Configuration’. A menu will be displayed as shown in Figure 22 below.

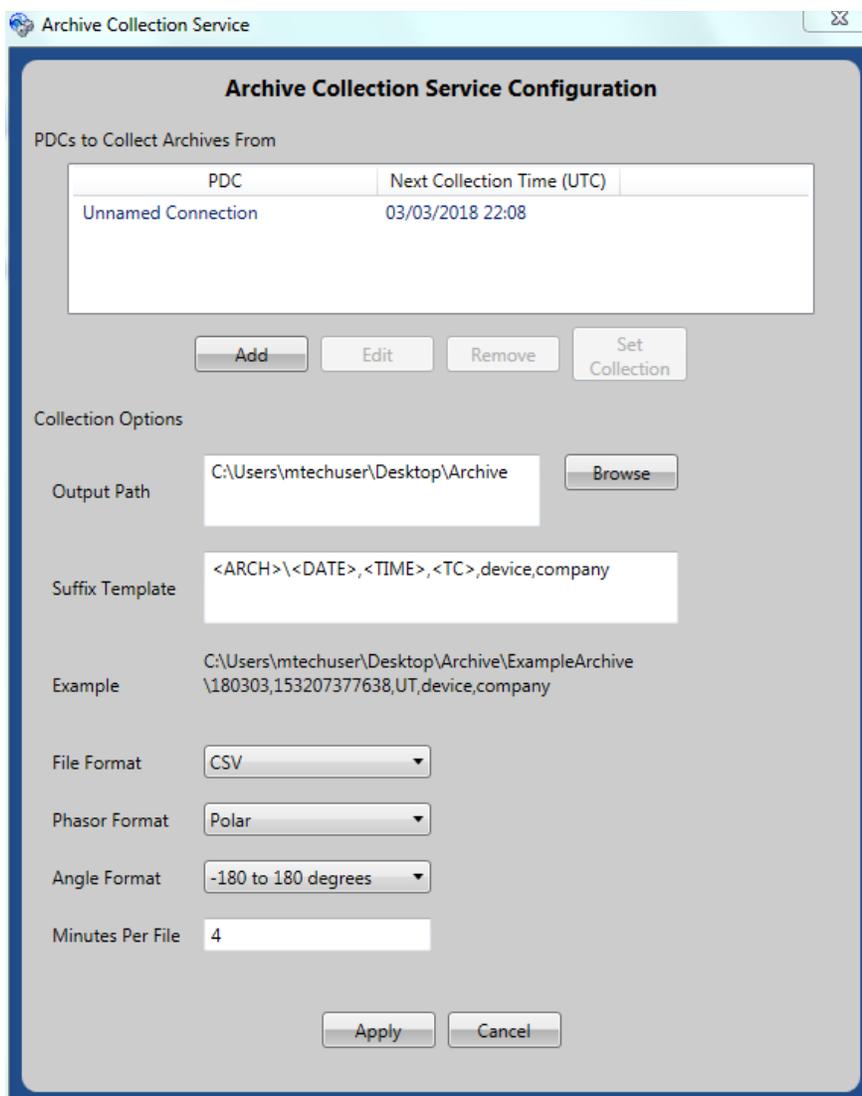


Figure 22: PDC Assistant – Configure Archive Collection

First add the PDC for archiving and adjust the phasor format and angle format to suit testing needs. PMU parsing software provided with this test bench uses a .csv read function, so file format should be adjusted to CSV. Specify an output path for the .csv file to be saved under and adjust the ‘Minutes Per File’ to accommodate the length of the test being performed. To start collecting PMU data click ‘Local Services’ and select ‘Start Collecting Archive’. PMU data is now being archived. To stop archiving, click ‘Local Services’ and select ‘Stop Collecting Archive’. To obtain the PMU data click ‘Retrieve Archive’ as shown below.

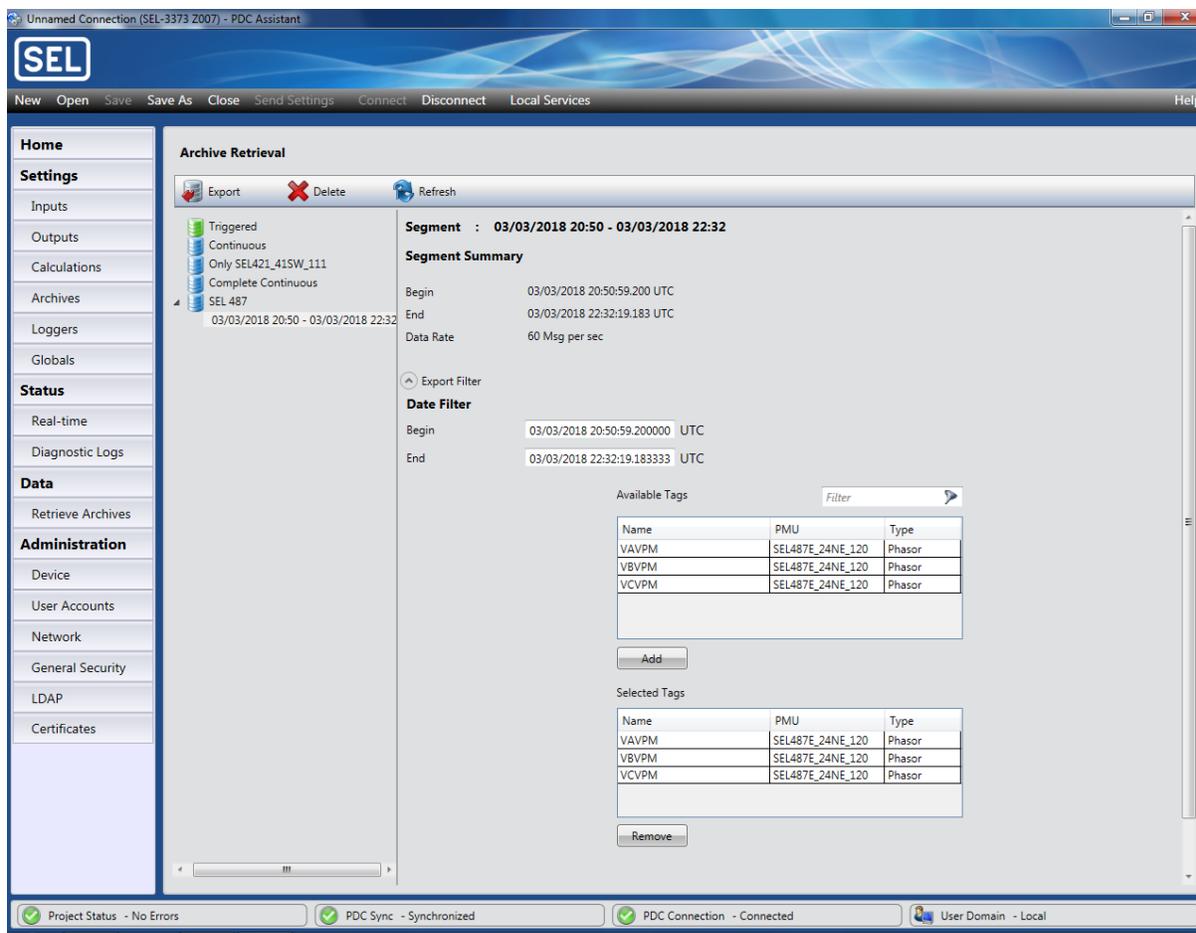


Figure 23: PDC Assistant – Retrieve Archive

A list of available PMUs will be displayed. Each PMU will have an accompanying list of all available archives. In Figure 23 above the continuous archive ‘SEL 487’ is selected and the archived data file ‘03/03/2018 20:50 – 03/03/2018 22:32’ is selected. Selecting the ‘Data Filter’ option will allow users to adjust the begin and end time of the file, as well as selecting from available PMU data to be tagged and stored in the .csv. Once these settings have been adjusted clicking ‘Export’ will populate the specified directory with the .csv file containing the PMU test data.

4.2.3. Parsing PMU Output Data and Analog Input Data

To parse PMU output data or analog input data open the 'BasicCsvReader.m' Matlab software. This software is written so that parsing either PMU output data, or analog input data can be done by running a single script file. Sections in the code are labeled as 'PMU Data Parsing' or 'Analog Input Data Parsing'. Depending on the needs of the user sections can be run simultaneously or separately. Textscan functions are used to separate .csv data into a cell arrays. Data is further organized into variables containing various information such as voltages and time vectors. The data are then saved as .mat files so that it can be easily loaded into separate analysis software. Figure 24 shows how to open and parse specific files.

```
fileID = fopen('PDCstuff.csv');
CP = textscan(fileID,...
              '%f%c%f%c%f %f:%f:%f %f %f %f %f %f %f',...
              'Headerlines',1,'Delimiter',' ');
fclose(fileID);
```

Figure 24: Basic csv Reader Software

The fopen function is used to open specific .csv files which must be in the same folder as the BasicCsvReader.m software. The textscan function then reads through the .csv and parses the data. Textscan uses floating points, indicated by %f, and characters, indicated by %c, to determine how to store data in the cell array. The user may need to adjust the order of the second line in the textscan function to match the contents of the .csv file of interest. It is important to note that the PMU output data will only contain values set in the tagging section of the archives in PDC Assistant as described in section 4.2.2. If adjustments are made to return different PMU output data the parsing software will need to be adjusted to reflect the new contents of the .csv. By opening the PMU output .csv in Microsoft Excel the user can visually see the header line in the first row of the .csv along with the corresponding column data for each header. When

adjusting the parsing code to match the PMU output data it is advised to check the raw data in Microsoft Excel to make sure the data columns have been separated out and named correctly.

4.2.4. Summary - Running a Test

After the pre-testing procedures have been performed and the safety checks have been made a generic PMU test may be performed as follows. First open PDC Assistant and LabVIEW signal generation software on the PXIe-8840. Consider the length of the test signal and plan a start time for the test. In PDC assistant initiate PMU data archiving before the test signal start time to ensure all output data is recorded for the input test signal. Once the input signals have begun monitor signal performance on the test bench oscilloscope. This is a safety step in case something goes wrong in the testing process. After completion of the test click the stop sign in the LabVIEW signal generation software. In PDC assistant click 'Local Services' and stop PMU data archiving. Export PMU output data for parsing and analysis.

5. PMU Step Testing

Testing for the step response of PMUs was performed as part of this project. Step tests serve as a base line for understanding PMU dynamic performance and give insight into PMU filtering. As seen in Figure 1 PMU signal processing is largely made up of filters. Finite Impulse Response (FIR) filters are used in PMUs for phasor estimation. FIR filters take a window of sampled values from the input of the PMU to make a single phasor estimate. The placement of this phasor estimate within the FIR filter window is important to understanding the true behavior of PMUs to dynamic inputs. Delay time, as seen in Figure 4, will be affected by the placement of the time tag within the FIR filter window. PMU manufacturers will wish to place the time tag so that the delay appears to be nearly zero. This accounts for the group delay inherent in filtering signals. Compensating for group delay has the benefit of ensuring that different PMUs with different phasor estimation algorithms will perform similarly as described in the C37.118 standard.

By observing PMU responses to step tests, test bench users can begin to see visually the FIR filtering window and can gain an understanding as to how time tagging is performed within this window. This knowledge will build into section 6 and will also come full circle in the discussions section. Section 4 covered a procedural approach to generic PMU testing. This section will follow those basic procedures for PMU step testing and will build by covering signal descriptions, analysis, and results from step testing.

5.1. Magnitude Step Tests – Input Signal Description

Magnitude step testing was implemented using signal generation method 1. Signal run time and corresponding file size made this type of testing ideal for Matlab signal generation. A major aspect of this project was developing the signals for PMU testing. With the basics of the

'GeneralSigGen.m' covered in section 3.7.1, the details in setting up a magnitude step test will be covered here. The PMU will be tested using three phase voltage signals with nominal frequency set to 60 Hz. Test length will last for 40 seconds. The three phase signals will step on at 10 seconds into the test and then will step off at 30 seconds. This will give 10 seconds of zero volts at the beginning and end of the test. Settings for the time vector are shown below.

```
%% Time Vector
fs    = 10000;           % Sampling frequency in Hz
ts    = 1/fs;           % Time between samples
tTot  = 40;              % Test length in seconds
t     = 0:ts:tTot-ts;   % Time Vector
```

Figure 25: Setting Time Vector for Magnitude Step Test

The structured parameters for the magnitude step test will be set as follows. To step the signal on at 10 seconds the variable tStart is set to 10. To step the signal off at 30 seconds the variable tEnd is set to 30. The variable for SigMag is set to 4.5 while Amp.Mod.Mag is set to 0.222. When the magnitude step occurs at ten seconds these settings will result in a peak voltage magnitude of $(4.5) \cdot (1.222) = 5.5$ volts. Peak voltage magnitude must be taken into consideration when the low voltage signal from the PXIe-6738 will be amplified by the 25x gain of the Tabor 9400, resulting in a peak amplified voltage of 137.5 volts and 97.2 volts rms.

```

fo                = 60;           % Eqn 18  -> fo
AmpOffset         = 0;           % Eqn 18  -> Ao
tStart           = 10;           % Eqn 18  -> tStart
tEnd             = 30;           % Eqn 18  -> tEnd
SigMag          = 4.5;           % Eqn 19  -> A10
Amp.Mod.Mag     = 0;           % Eqn 19  -> kx
Amp.Mod.Freq    = 0;           % Eqn 19  -> fm
Amp.Mod.tOn     = 0;           % Eqn 19  -> tAmodOn
Amp.Mod.tOff    = 0;           % Eqn 19  -> tAmodOff
Amp.Step.Mag    = 0.222;        % Eqn 19  -> Astep
Amp.Step.tOn    = 10;           % Eqn 19  -> tAmpStepOn
Amp.Step.tOff   = 30;           % Eqn 19  -> tAmpStepOff
phiOffset       = 0;           % Eqn 20  -> phiOffset
Phase.Ramp.Freq = 0;           % Eqn 20  -> deltafr
Phase.Ramp.tOn  = 0;           % Eqn 20  -> tPhaseRampOn
Phase.Ramp.tOff = 0;           % Eqn 20  -> tPhaseRampOff
Phase.Mod.Mag   = 0.0;         % Eqn 20  -> ka
Phase.Mod.Freq  = 0;           % Eqn 20  -> fm
Phase.Mod.tOn   = 0;           % Eqn 20  -> tPhaseModOn
Phase.Mod.tOff  = 0;           % Eqn 20  -> tPhaseModOff
TrigMag         = 5.5;         % Eqn 21  -> Trigger Magnitude
Trig(1).tOn     = 10;         % Eqn 21  -> Trigger One On
Trig(1).tOff    = 30;         % Eqn 21  -> Trigger One Off
Trig(2).tOn     = 0;           % Eqn 21  -> Trigger Two On
Trig(2).tOff    = 0;           % Eqn 21  -> Trigger Two Off

```

Figure 26: Structured Parameter Settings for Magnitude Step Test

The signal parameters above will yield the following signal.

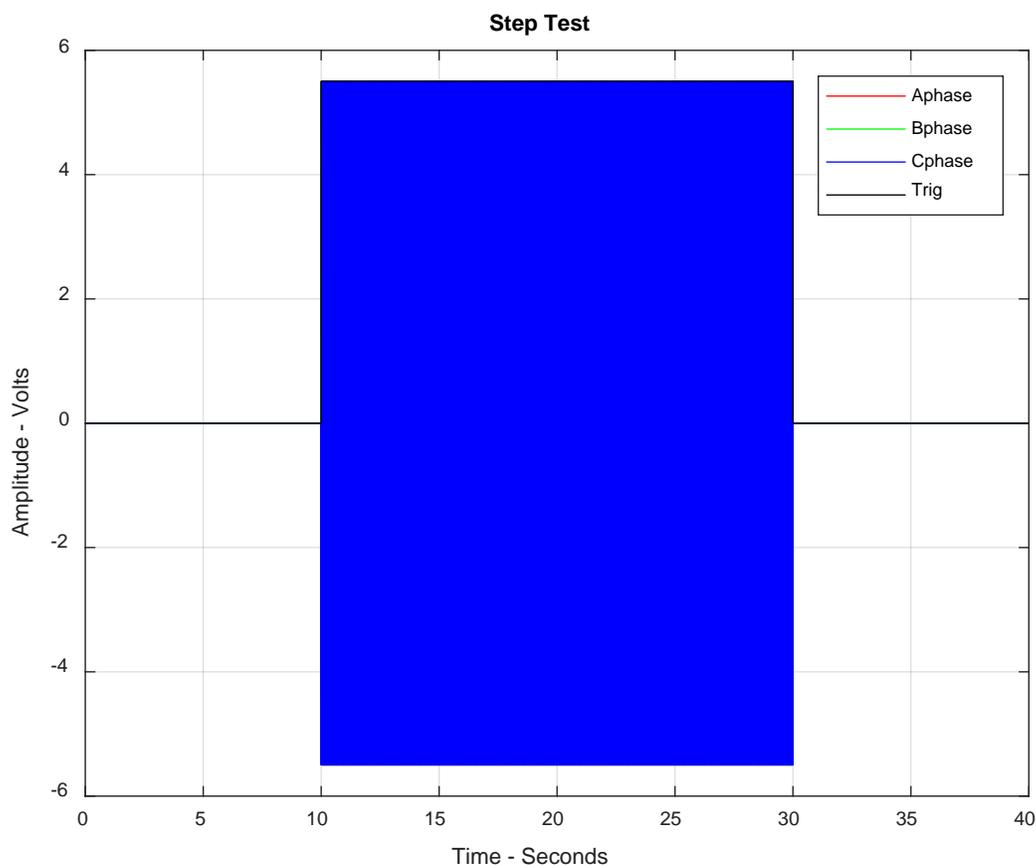


Figure 27: Magnitude Step Test – Input Signals with Trigger

The signals are then stored in a row matrix and saved to a .csv file for transfer to the LabVIEW signal generation software. PMU data archiving was started, and the signal start time was then specified. Upon completion of the test PMU output data was collected and parsed using the software described in section 4.2.3.

5.1.1. Magnitude Step Test – Results

The results from magnitude step testing are shown below. The plot shown in Figure 28 is a fully zoomed out view of the A phase magnitude retrieved from the PMU output data. A phase magnitude will be used to view the response of the PMU to magnitude step tests throughout this section as a summary of the performance of all three phases.

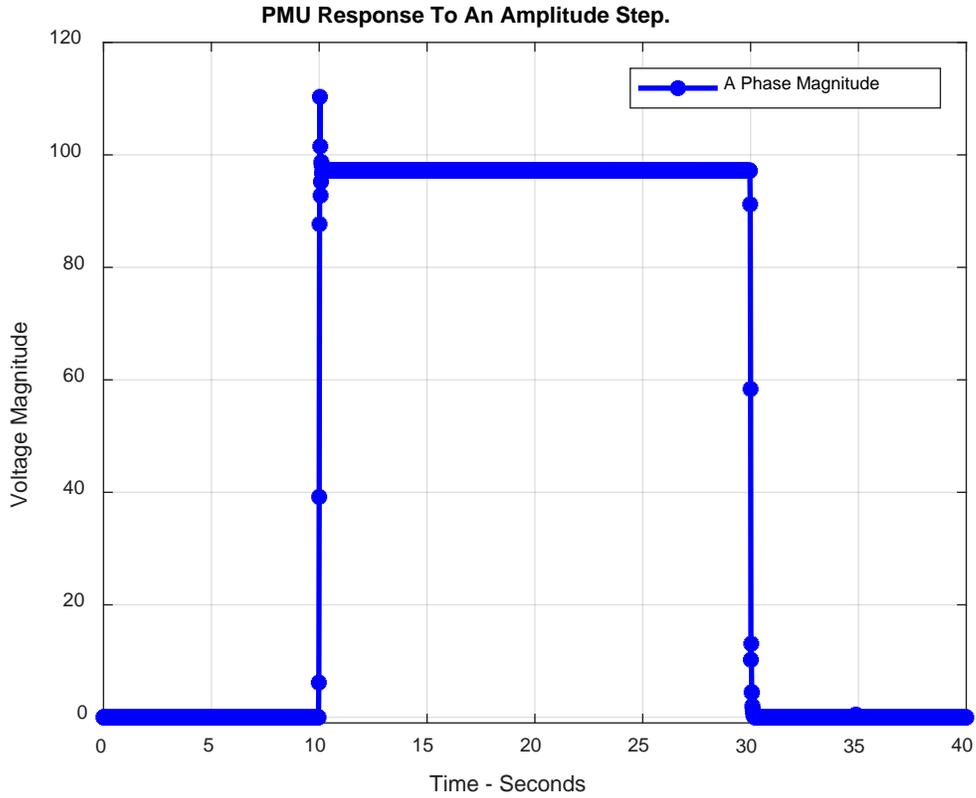


Figure 28: PMU Magnitude Step Response – Full Response

To get a better view of the step response, Figure 28 above is zoomed-in at 10 seconds. This plot shows that the PMU responds to a step change non-causally. A non-causal response refers to the PMU responding to the step input before the step change occurs at the input of the device.

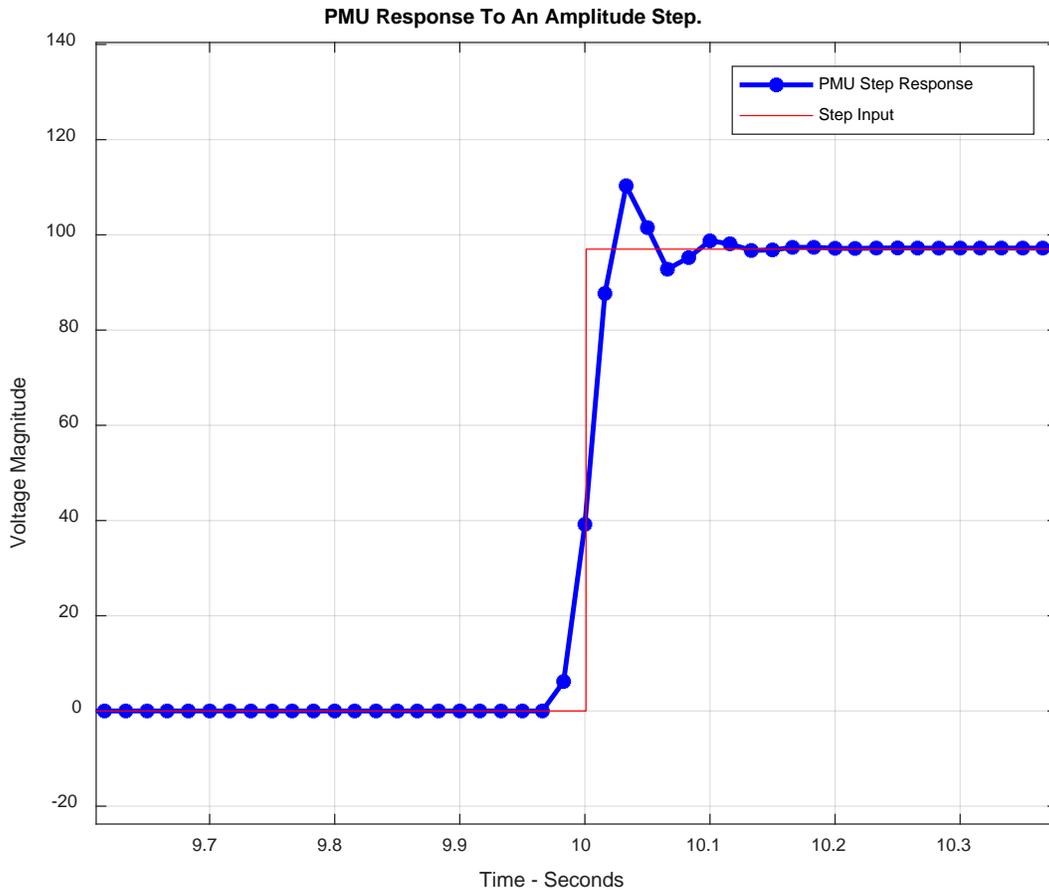


Figure 29: Zoom On Non-Causal PMU Response

5.2. Frequency Step Tests – Input Signal Description

Step testing was also performed by stepping frequency. Signal frequency was stepped from 60 Hz to 60.5 Hz. The signal steps on at 10 seconds into the test at 60 Hz. Signal frequency then steps to 60.5 Hz at 15 seconds into the test. The signal is zeroed out at 30 seconds into the test. Figure 30 illustrates this test signal, with the trigger indicating where the frequency step occurs.

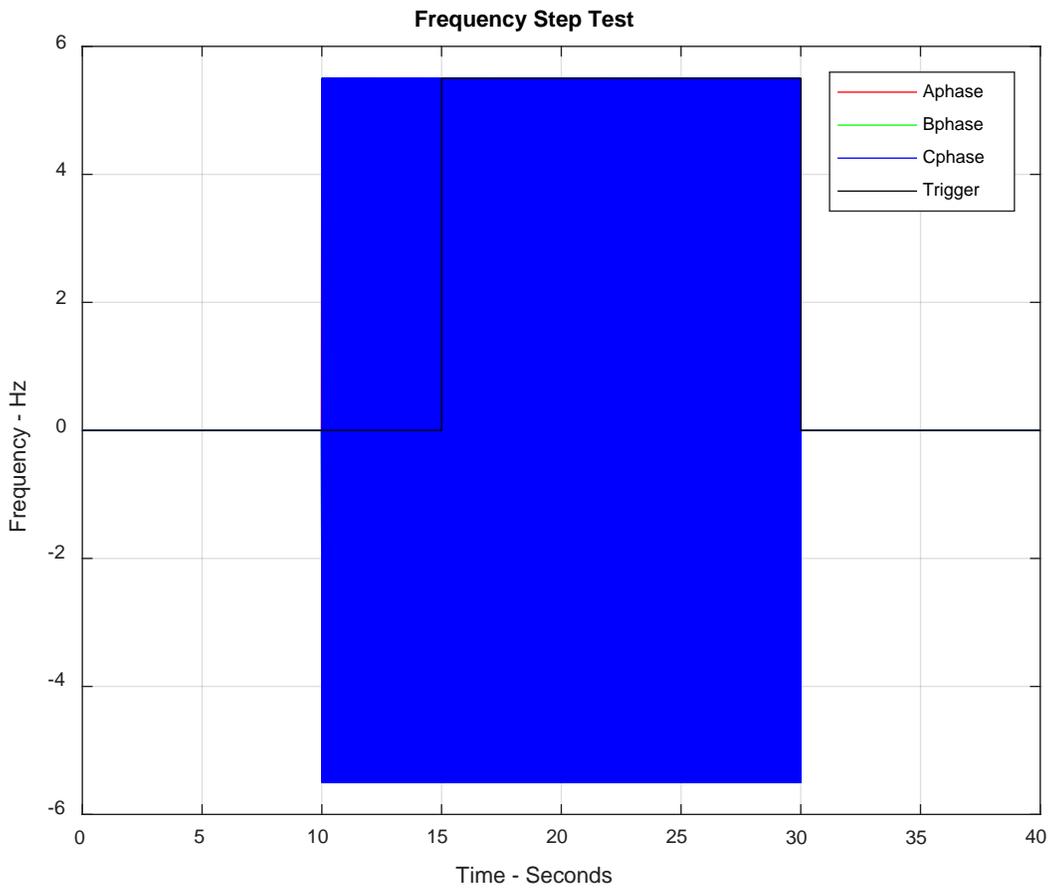


Figure 30: Frequency Step Test Input Signal

5.2.1. Frequency Step Test - Results

The results from stepping frequency are shown in Figure 31 below. These results also show that the PMU responds to a frequency step non-causally.

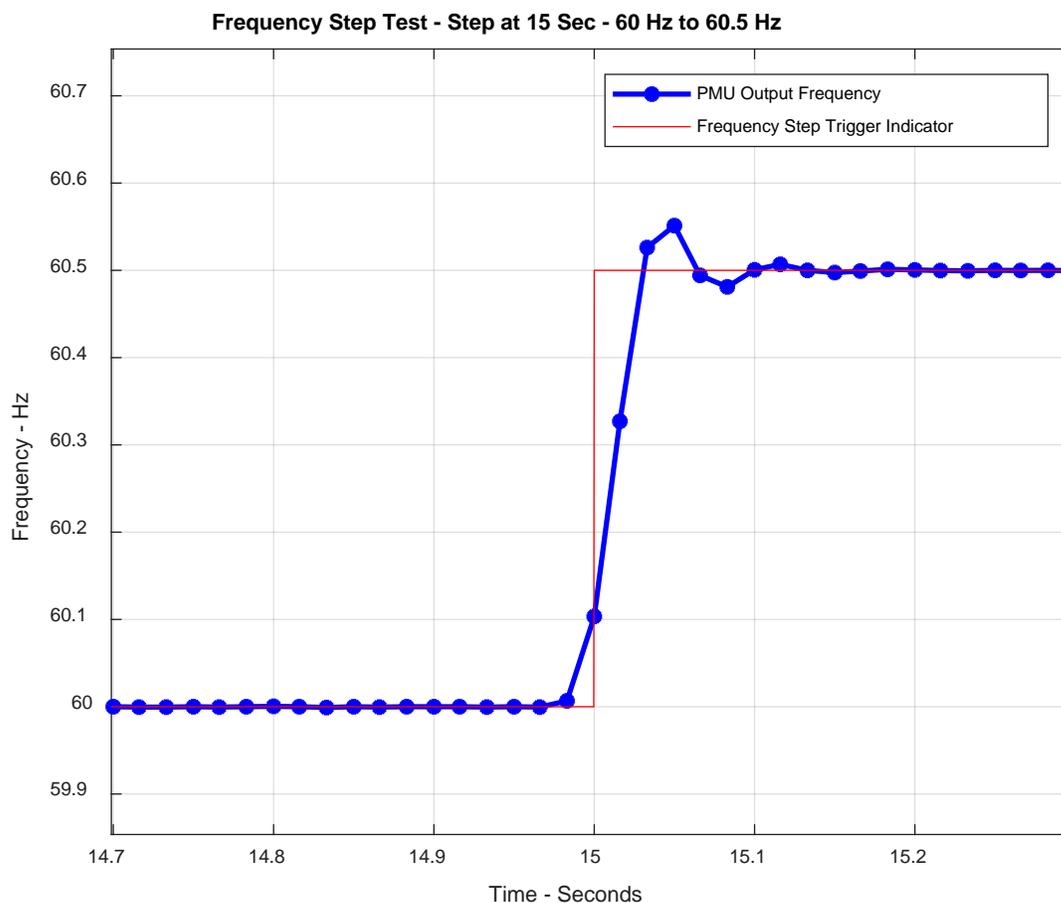


Figure 31: Frequency Step Test

5.3. Construction of High Resolution PMU Step Response

Inspection of the magnitude and frequency step tests show that the FIR filter window spans roughly 4 samples and that timestamping occurs roughly in the middle of this window. A test was developed to give a better picture of the FIR filtering window, and where timestamping occurs within this window. With a reporting rate of 60 sps, the output time resolution of the PMU is 16.67 ms between samples. This sample rate is not high enough to see the exact response of the PMU to a step input. Being able to start input signals to the PMU at exact times allows for a test which will ‘fill in’ the PMU response curve to a step input. This test will consist of 11 different magnitude step tests run individually. All tests were started at the top of a minute. Step

test 1 will have a magnitude step that occurs at 15 seconds into the test. Step test 2 will have a magnitude step that occurs at 15.001667 seconds into the test. The following step tests will start at 15 seconds and a multiple of 1.667 ms into the test. The 1.667 ms is one tenth of the PMU reporting rate of 16.67 ms between samples. Running ten tests with a signal start time offset by multiples of 1.667 ms will allow for evenly spaced samples that will provide a higher resolution curve. Table II below summarizes this test.

Table II: Magnitude Step Tests – Step On Start Times

	Magnitude Step Start Time
Step Test 1	15.000000 Seconds
Step Test 2	15.001667 Seconds
Step Test 3	15.003334 Seconds
Step Test 4	15.005001 Seconds
Step Test 5	15.006668 Seconds
Step Test 6	15.008335 Seconds
Step Test 7	15.010002 Seconds
Step Test 8	15.011669 Seconds
Step Test 9	15.013336 Seconds
Step Test 10	15.015003 Seconds
Step Test 11	15.016670 Seconds

The structures for setting these tests are similar to the structures described above for a magnitude step test. It is important to note that the magnitude step that occurs at 15 seconds will be a 10% increase from the initial magnitude of the input signal. More specifically the signal will step from 100 volts rms to 110 volts rms. Figure 32 illustrates Step Test 1 before amplification.

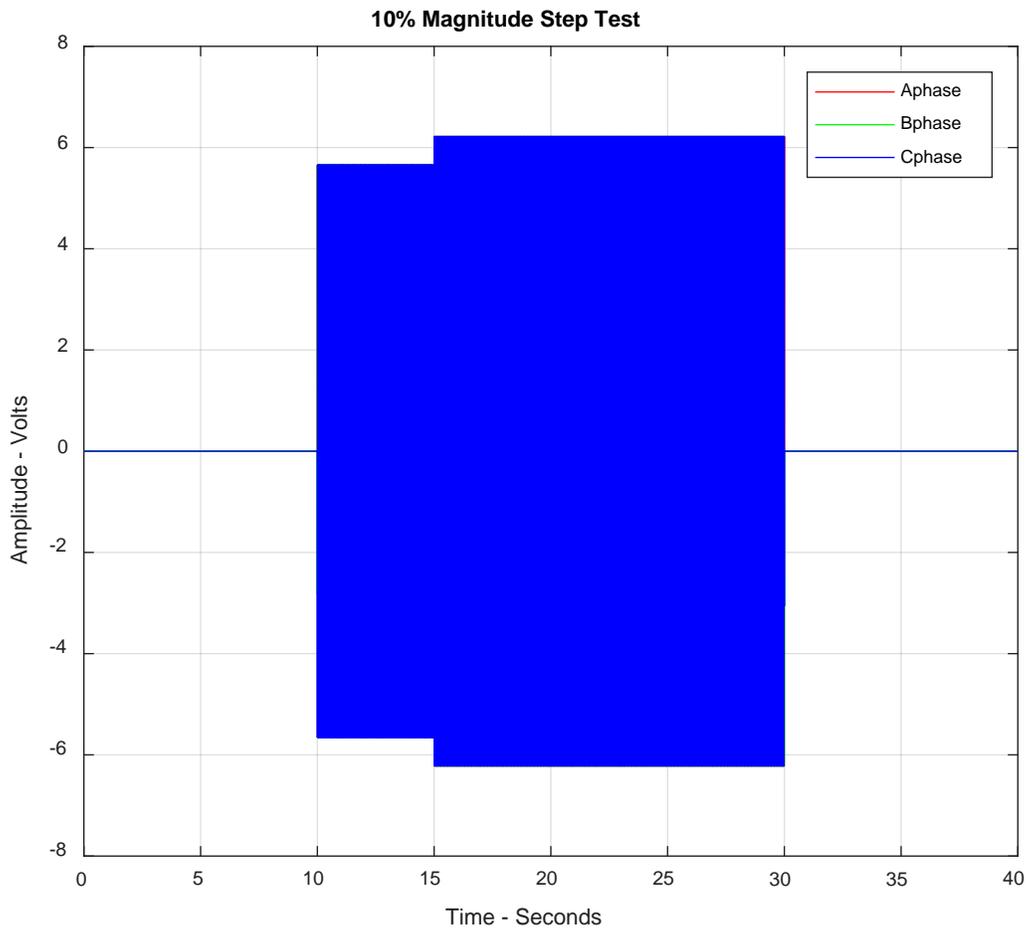


Figure 32: Magnitude Step Test 1 – 10% Step at 15 Seconds

Each magnitude step test was saved to a .csv and run in LabVIEW to produce the analog input to the PMU. PMU output data was recorded for each test and all data was parsed and analyzed.

5.3.1. High Resolution PMU Step Response - Results

Figure 33 illustrates the results from the step testing with all PMU step responses plotted together.

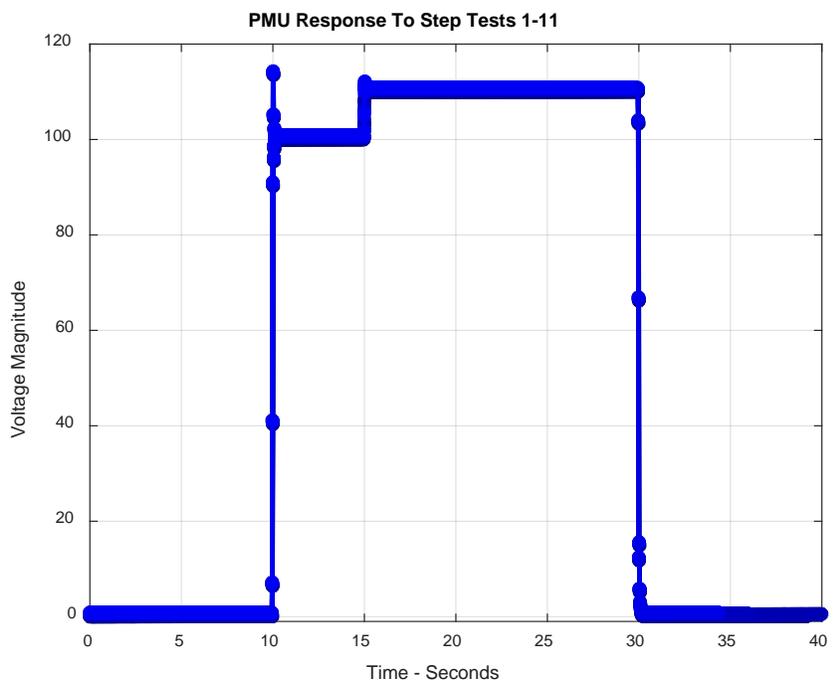


Figure 33: PMU Response to Magnitude Step Tests 1-11 All Plotted Together

Zooming in on the PMU response to the steps at 15 seconds plus offset yields the following Figure 34.

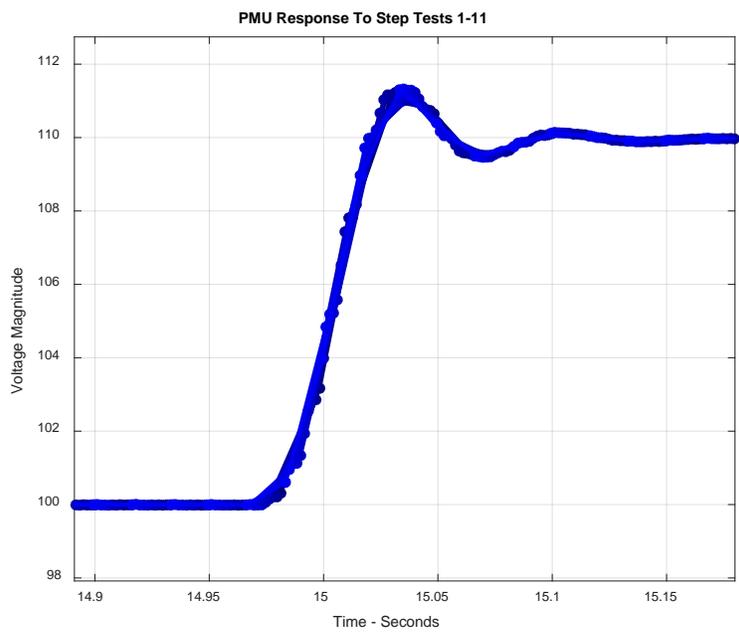


Figure 34: Zoomed PMU Response to Magnitude Step Tests 1-11 All Plotted Together

Figure 34 above shows the raw data from the PMU response to magnitude step tests 1-11 all plotted together. To better understand how to analyze this data a plot of three step test sets will be analyzed. The data will be taken from step test 1, step test 5, and step test 9. Just looking at the raw data will show that the time vectors for all three step tests are identical. Since each test was started at the top of a minute, this shows that the PMU under test is sampling the same for each step test. Hence the samples for the three PMU response curves will lie right on top of each other with respect to time as shown below.

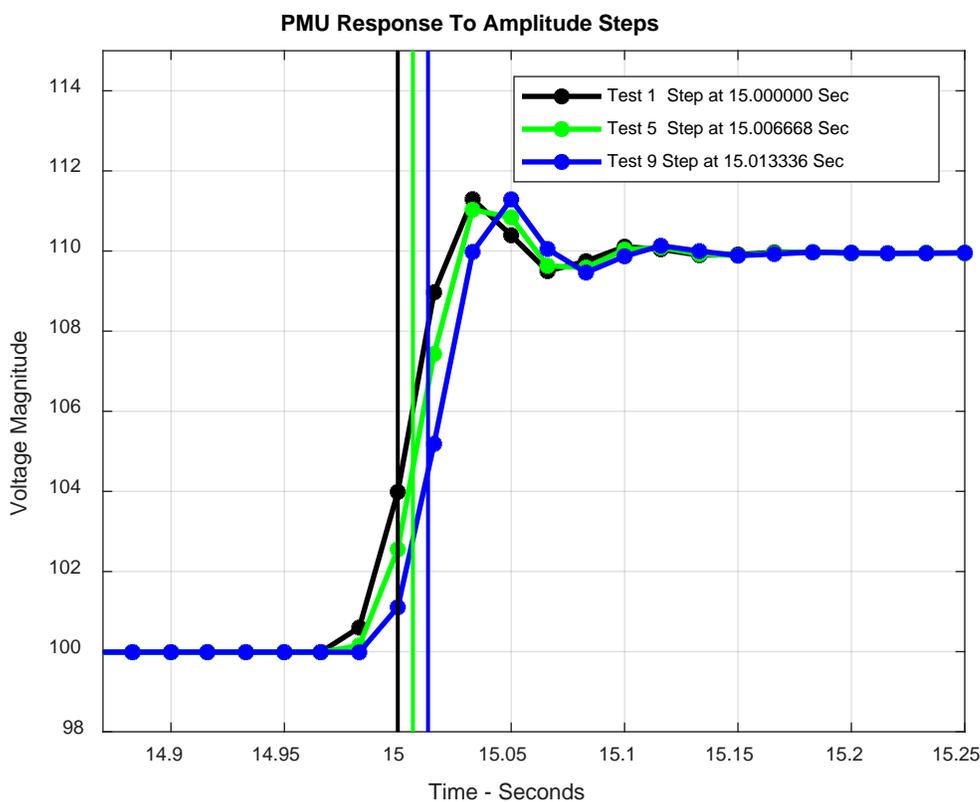


Figure 35: PMU Response to Amplitude Steps – No Time Adjustment

The vertical traces indicate the step time for each test, as described in Table II above. The goal of this test is to now construct a curve from the 11 step tests that will represent the PMU response to a step input with greater resolution. To accomplish this the time vectors for each step test will be shifted back so that each step occurs at 15 seconds. This would in effect make it so

that the vertical traces shown in Figure 35 all lie on top of each other at 15 seconds. This time shift is shown in Figure 36 below.

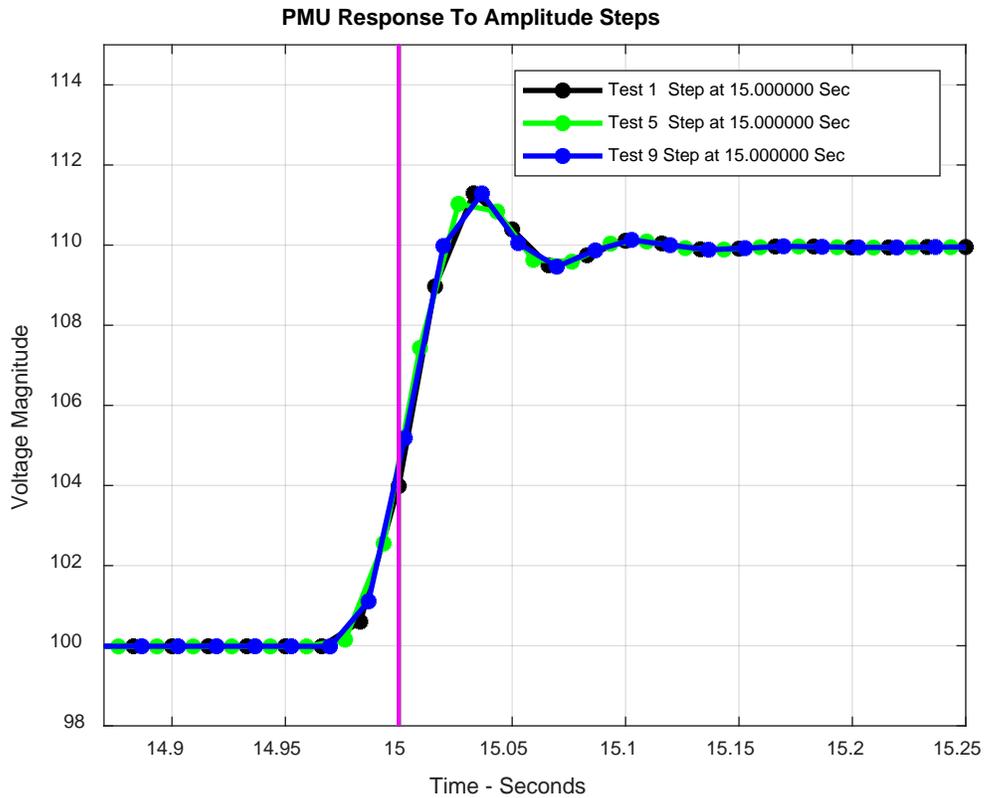


Figure 36: PMU Response to Amplitude Steps – Time Adjusted

Notice from Figure 36 above that the samples for each curve are now offset from each other with respect to time. The data for each of the 11 step tests was aggregated together to form a single curve. This curve will have better resolution with samples having 1.667 ms between them. A curve with this resolution will be more informative about PMU behavior to a step input and will yield a close approximation to how time stamping is done inside the FIR filter window of the PMU under test. After aggregating the data the following curve was produced.

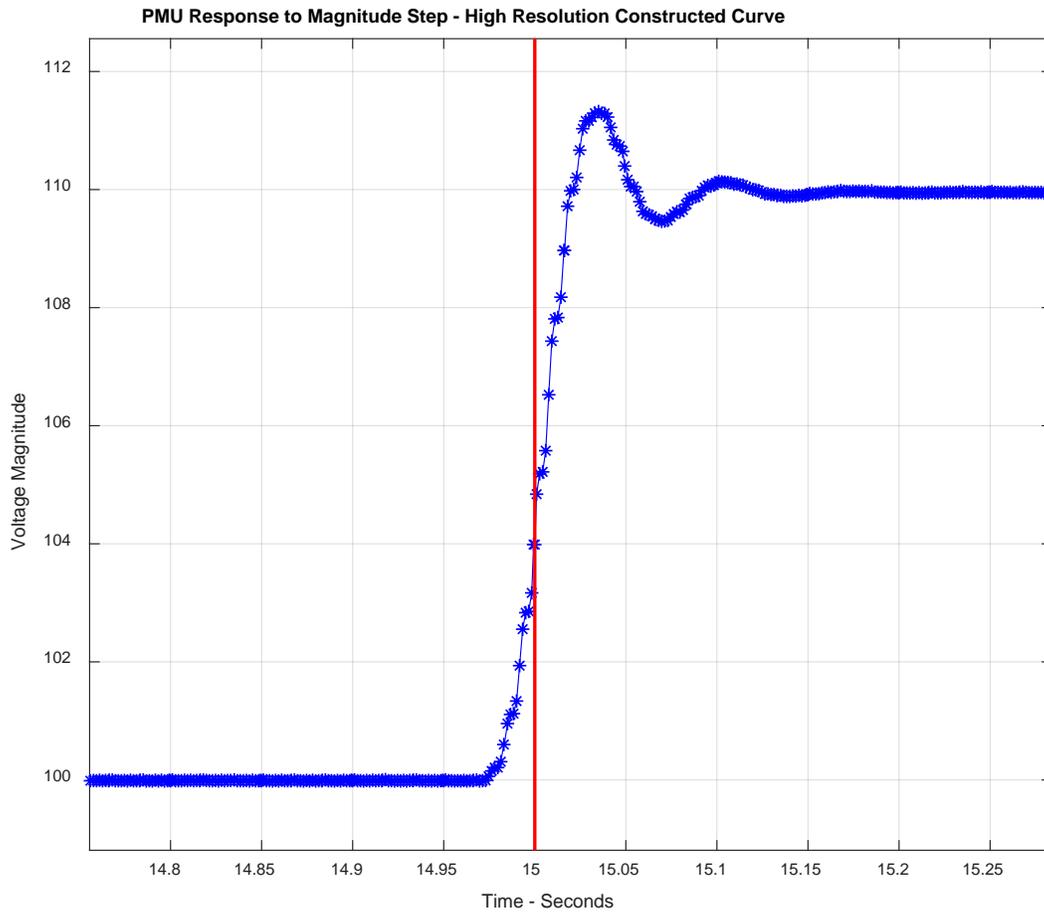


Figure 37: PMU Response to Amplitude Steps – High Resolution Constructed Curve

Of interest is a zoomed image of when the step occurs and when the PMU responds non-causally. The time difference between the step input at 15 seconds and the time when the PMU first responds will yield the advance in time seen from PMU step testing, and thus show how time stamping is performed within the FIR filter window.

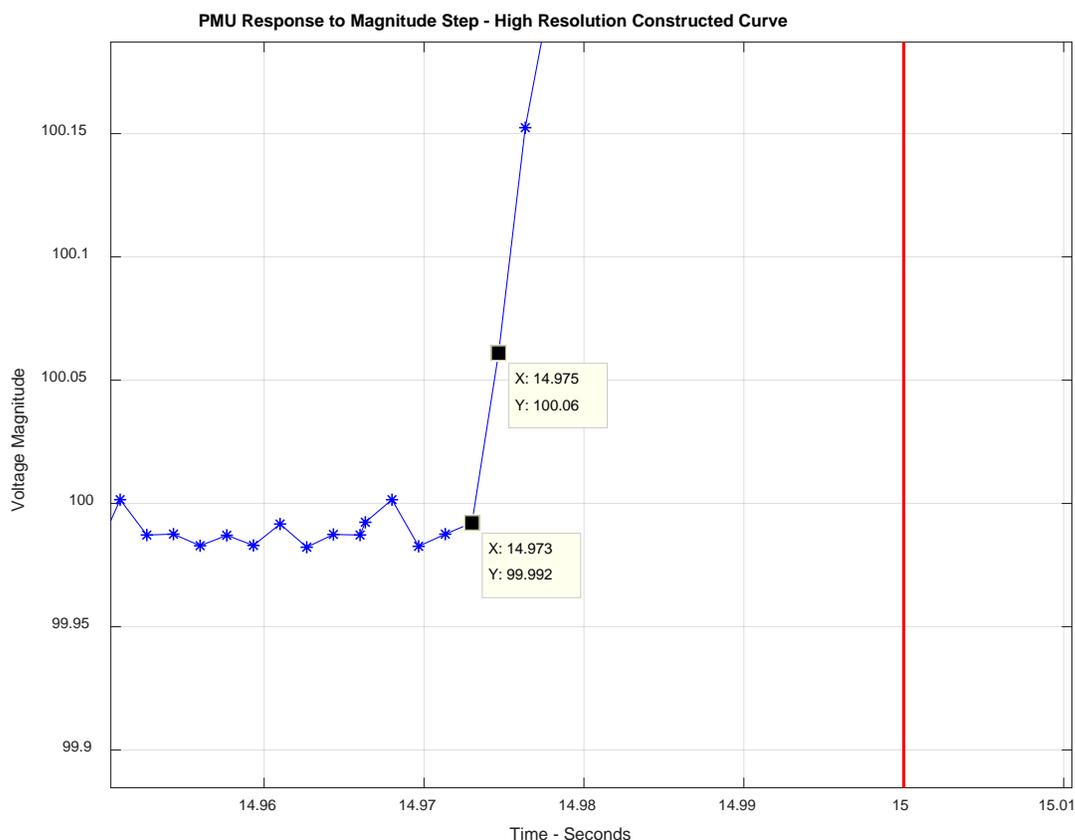


Figure 38: High Resolution Step Response – Advanced Timestamping

This plot shows the PMU has an advanced response to a step input between 25 ms and 27 ms. This test was further verified by contact with the PMU manufacturer. Correspondence with the manufacturer revealed that there is a hardcoded advance in the filtering algorithm of 27.5ms which is very close to the experimental results. This advance is based on filter settings present in the PMU.

Further analysis of this test is done regarding the guidelines set in the C37.118 standard. The Standard describes allowable overshoot and delay time for a 10% magnitude step. The step testing above was designed around a 10% step as well. The standard describes overshoot as shown in Figure 4 and specifies a maximum overshoot of 5% of step magnitude for a P class PMU, and a maximum overshoot of 10% of step magnitude for an M class PMU. For a step

magnitude of 10 V_{rms} this corresponds to an overshoot of 0.5 V_{rms} for P class PMUs and an overshoot of 1.0 V_{rms} for M Class PMUs. Figure 39 shows the max overshoot for the magnitude step test.

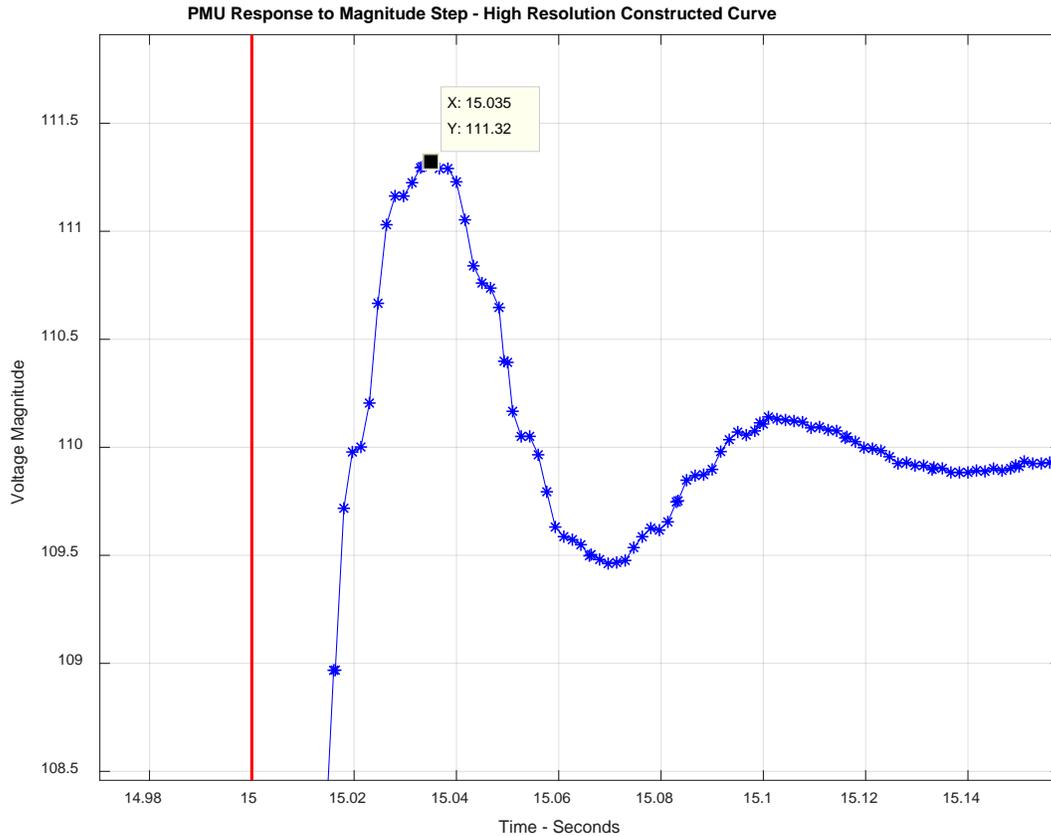


Figure 39: PMU Response to Magnitude Step Test - Overshoot

Figure 39 above shows that for a 10% magnitude step the PMU violates the standards maximum allowable overshoot as the percentage of step magnitude is at 13.2%. Documentation for the PMU under test does not specify filter settings for P class or M class PMUs. Delay time is illustrated in Figure 4. The C37.118 standard specifies maximum delay time as follows.

$$\begin{aligned}
 \text{Max Delay Time} &= \frac{1}{4(\text{Frequency of Measurement Data Reporting})} \\
 &= \frac{1}{4(60 \text{ sps})} = 4.1667 \text{ ms}
 \end{aligned}
 \tag{24}$$

The following figure zooms on the time in which the PMU response crosses 50% of the final step value, or $105 \text{ V}_{\text{rms}}$. The time difference between this crossing point and the step time at 15 seconds will yield the delay time of the PMU under test.

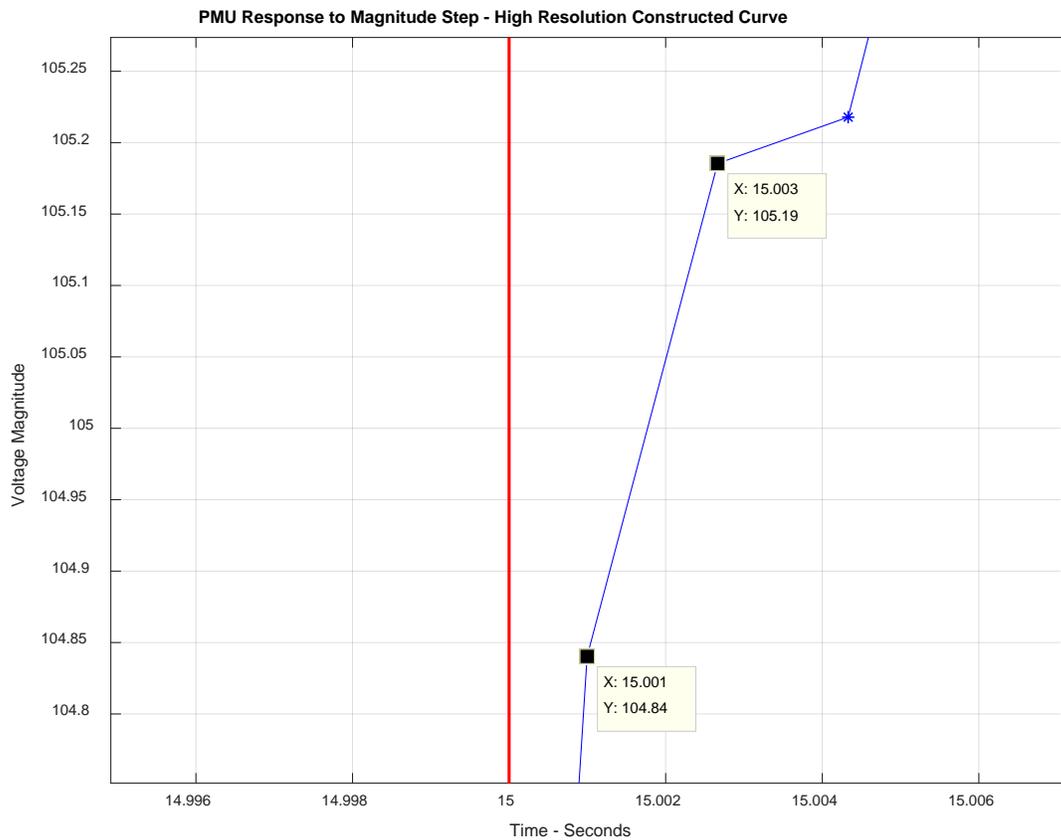


Figure 40: PMU Step Response – Delay Time

The delay time for the PMU under test is between 1 ms and 3 ms. This delay time is within the limit set by the C37.118 standard as seen in (24).

6. Phase Modulation Testing

This section will begin by describing the signals used for phase modulation testing and the signal generation methods. Results and analysis techniques will be covered to show how the Bode plot was produced for the PMU under test. A brief discussion on the shape of the Bode plot will lead into the discussion section where these ideas will then be expanded on.

6.1. Phase Modulation Signal Description

Signal generation method 1, described in Figure 17, was initially used to produce phase modulation signals for PMU testing. There are different analysis techniques and test signals that could be used for estimating the frequency response of a device under test. The method chosen for this test bench ensured accurate estimation of the PMUs frequency response by running one phase modulation frequency at a time. This method then requires running a battery of phase modulation tests. For each test the phase modulation frequency term, f_m in (17), was set to a specified value ranging from 0.1 Hz to 60 Hz. A single test would therefore produce a single point on the Bode plot. The next test would consist of incrementing the phase modulation frequency term which would then produce the next point on the Bode plot, and this process would be repeated until a full Bode plot was produced. To produce a full Bode plot the phase modulation frequency terms for this battery of tests were run as follows.

Table III: Phase Modulation Testing – Modulation Frequency Battery Test

	Phase Modulation Frequency f_m		Phase Modulation Frequency f_m
Test 1	0.1 Hz	Test 36	3.6 Hz
Test 2	0.2 Hz	Test 37	3.7 Hz
Test 3	0.3 Hz	Test 38	3.8 Hz
Test 4	0.4 Hz	Test 39	3.9 Hz
Test 5	0.5 Hz	Test 40	4.0 Hz
Test 6	0.6 Hz	Test 41	4.1 Hz
Test 7	0.7 Hz	Test 42	4.2 Hz
Test 8	0.8 Hz	Test 43	4.3 Hz
Test 9	0.9 Hz	Test 44	4.4 Hz
Test 10	1.0 Hz	Test 45	4.5 Hz
Test 11	1.1 Hz	Test 46	4.6 Hz
Test 12	1.2 Hz	Test 47	4.7 Hz
Test 13	1.3 Hz	Test 48	4.8 Hz
Test 14	1.4 Hz	Test 49	4.9 Hz
Test 15	1.5 Hz	Test 50	5.0 Hz
Test 16	1.6 Hz	Test 51	6.0 Hz
Test 17	1.7 Hz	Test 52	7.0 Hz
Test 18	1.8 Hz	Test 53	8.0 Hz
Test 19	1.9 Hz	Test 54	9.0 Hz
Test 20	2.0 Hz	Test 55	10.0 Hz
Test 21	2.1 Hz	Test 56	12.0 Hz
Test 22	2.2 Hz	Test 57	14.0 Hz
Test 23	2.3 Hz	Test 58	16.0 Hz
Test 24	2.4 Hz	Test 59	18.0 Hz
Test 25	2.5 Hz	Test 60	20.0 Hz
Test 26	2.6 Hz	Test 61	22.0 Hz
Test 27	2.7 Hz	Test 62	24.0 Hz
Test 28	2.8 Hz	Test 63	26.0 Hz
Test 29	2.9 Hz	Test 64	28.0 Hz
Test 30	3.0 Hz	Test 65	30.0 Hz
Test 31	3.1 Hz	Test 66	35.0 Hz
Test 32	3.2 Hz	Test 67	45.0 Hz
Test 33	3.3 Hz	Test 68	50.0 Hz
Test 34	3.4 Hz	Test 69	55.0 Hz
Test 35	3.5 Hz	Test 70	60.0 Hz

Running a battery of 70 tests, collecting 70 sets of PMU output data, and parsing the data sets individually is a long and arduous task. Combining all phase modulation testing into a single signal would make it possible to run a single test and collect a single set of PMU output data for analysis. This test signal consists of a train of phase modulated signals. This will in effect take all 70 of the tests described above and run them one by one consecutively in a single signal. The phase modulated signals will be separated by 2 seconds of a plain 60 Hz signal. The test signal

will also start off with 2 seconds of a plain 60 Hz signal before the first modulation burst. Each iteration of plain 60 Hz and the following burst of phase modulation will start with a new time vector starting at zero seconds. This will aid in the analysis of the test results. The analysis techniques used will require an adequate number of periods of signal data. The sinusoidal data obtained from the PMU has a high signal to noise ratio, considering that the PMU test bench was carefully designed to input user specified signals with low noise content. This means that the effect of measurement noise is likely insignificant to the analysis if a handful of periods of data are available. At the lowest phase modulation frequency approximately two periods of signal data are obtained for analysis. Running the test several times reveals that the same results are obtained each time. The results have little to no variance indicating that noise is not an issue and the number of periods used in the analysis is sufficient. For these reasons, the modulated signals with a slower frequency will be run longer to capture more periods of signal data. The faster frequencies will be run for a shorter period of time to improve overall test efficiency. Faster frequencies will not require as much time to capture many periods for analysis. Phase modulation bursts with f_m ranging from 0.1 Hz to 1.0 Hz will be run for 20 seconds each and will be separated by 2 seconds of a plain 60 Hz signal. Phase modulation bursts with f_m ranging from 1.1 Hz to 60.0 Hz will be run for 3 seconds each and will be separated by 2 seconds of a plain 60 Hz signal. The overall test signal duration will last for 525 seconds, or 8 minutes and 45 seconds. The signal will be sampled at 10 kHz. In Matlab a signal of this magnitude will be memory heavy and will take considerable time to load into LabVIEW. The delayed start time for a signal of this memory size will also be long. For these reasons the test signal was developed in LabVIEW. This is an implementation of signal generation method 2 illustrated in Figure 17. 'ThreePhaseFreqSweep.vi' is LabVIEW software that was developed for PMU phase modulation

testing. Figure 41 below shows the front panel of 'ThreePhaseSweep.vi'. The software has been configured so that all the user needs to do is specify a start time for the test in the 'Pulse Start Time' window. Total signal run time is 8 min and 45 seconds so PMU archive files will need to be adjusted to at least 9-10 minutes per file in PDC Assistant.

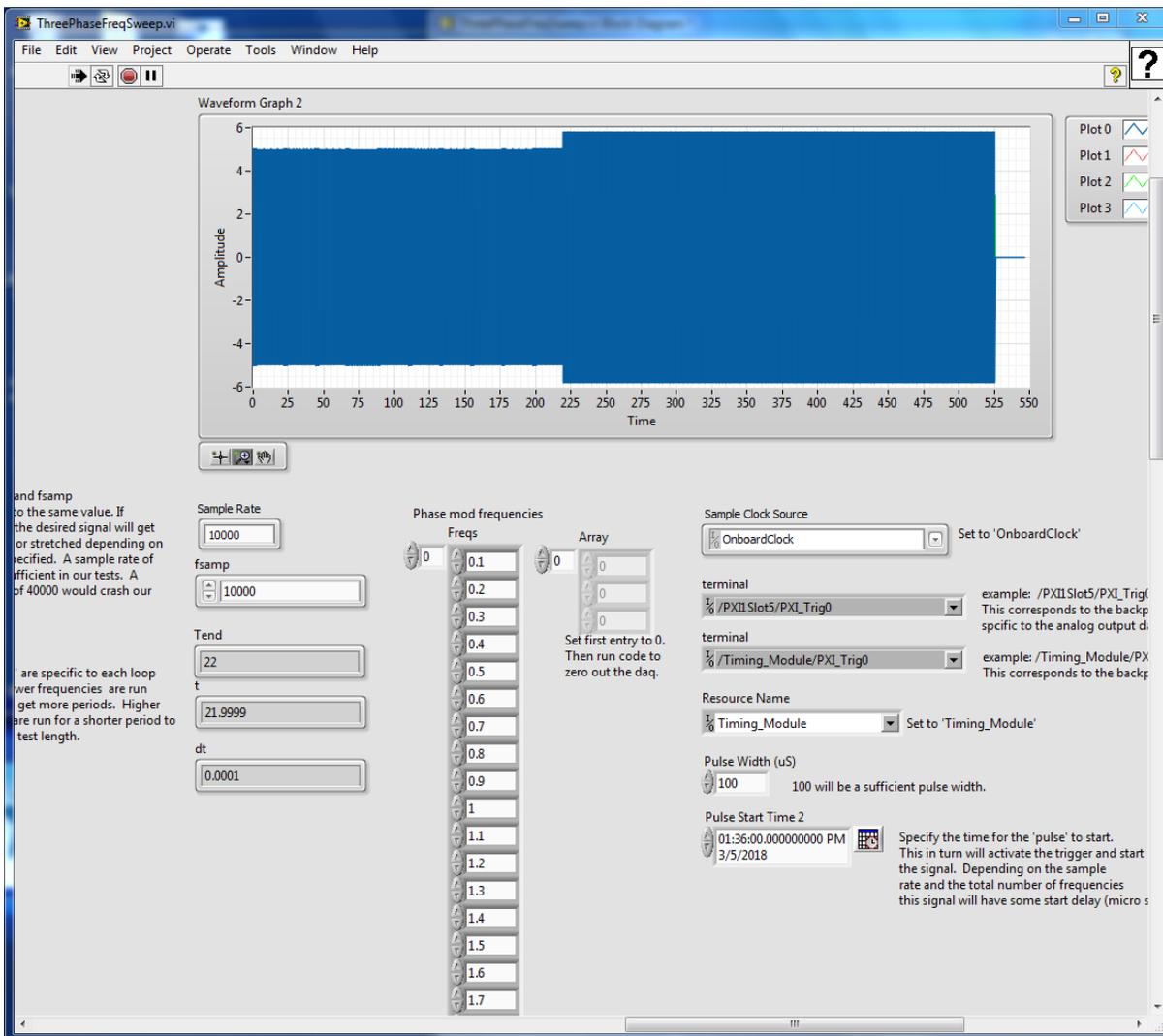


Figure 41: Phase Modulation Signal Generation Software

6.2. Analysis – Producing the Bode Plot - Overview

To begin this section, consider a plot of the phase angle output from the PMU against time for the phase modulation testing described in the previous section. The final analysis of this data and knowledge of the input signal will produce a Bode plot for the PMU under test.

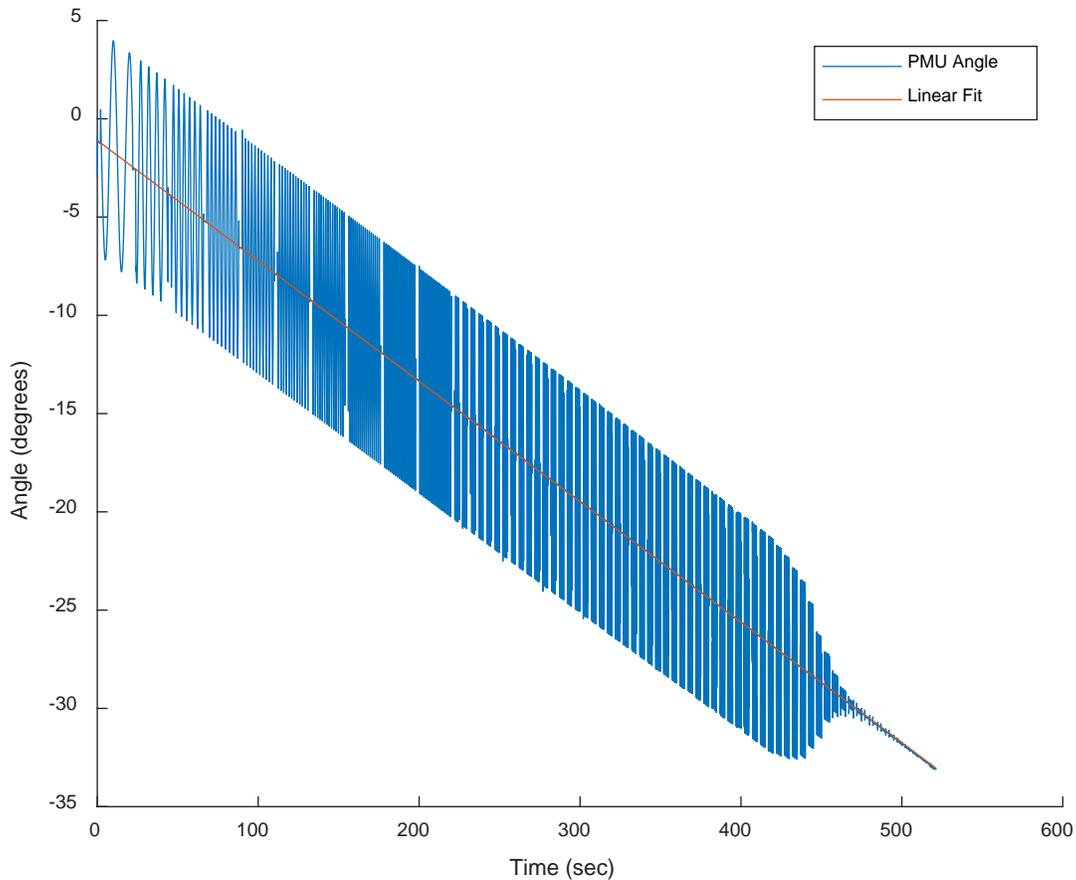


Figure 42: PMU Phase Angle Output

To obtain the Bode plot post processing must be done on the phase angle data shown above. This post processing will introduce error that must be accounted for. There is also error that is inherent from the test bench. The goal of this section will be to perform a transfer function calculation that accounts for all sources of error and in turn produces a Bode plot. The following equation will serve as a starting point for analysis. As the section progresses equation (25) will

be modified slightly in form but not function. Correction factors will also be added to the overall equation to account for test bench and post processing errors.

$$G(f_m) = \frac{\mathcal{F}\{\hat{f}(t)\}}{\mathcal{F}\{f(t)\}} \quad (25)$$

where $G(f_m)$ is the PMU transfer function calculated for each phase modulation frequency f_m . The Fast Fourier Transform (FFT) is denoted by $\mathcal{F}\{ \}$. Input frequency to the PMU is defined as $f(t)$ and the output frequency is defined as $\hat{f}(t)$. The transfer function $G(f_m)$ in (25) is built from PMU output phase angle data and input signal information. The term $\hat{f}(t)$ is derived by taking the derivative of PMU phase angle output. To better understand this analysis, equation (25) will be explored here, as the concept of doing a FFT on frequency can be confusing.

Consider a phase modulation input signal as defined by (15) and (17). The phase modulation input signal can be condensed into the following equation which includes only the terms needed for phase modulation.

$$v(t) = A \cos[2\pi f_o t + \phi(t)] = A \cos[2\pi f_o t + k_a \cos(2\pi f_m t)] \quad (26)$$

Consider that instantaneous signal frequency is the fundamental frequency plus a delta frequency term as in (9). Taking the derivative of the phase term in (26) will yield the delta frequency term. Instantaneous signal frequency then varies sinusoidally around the fundamental frequency of 60 Hz. This equation shows the derivative of $\phi(t)$ first in terms of sine and then cosine.

$$\Delta f(t) = \frac{d}{dt}(\phi(t)) = -2\pi f_m k_a \sin[2\pi f_m t] = 2\pi f_m k_a \cos[2\pi f_m t + 90^\circ] \quad (27)$$

Plots of the phase term $\phi(t)$ and instantaneous signal frequency, $f(t)$, against time are shown below where f_m is set to 10 Hz and k_a is set to 0.1.

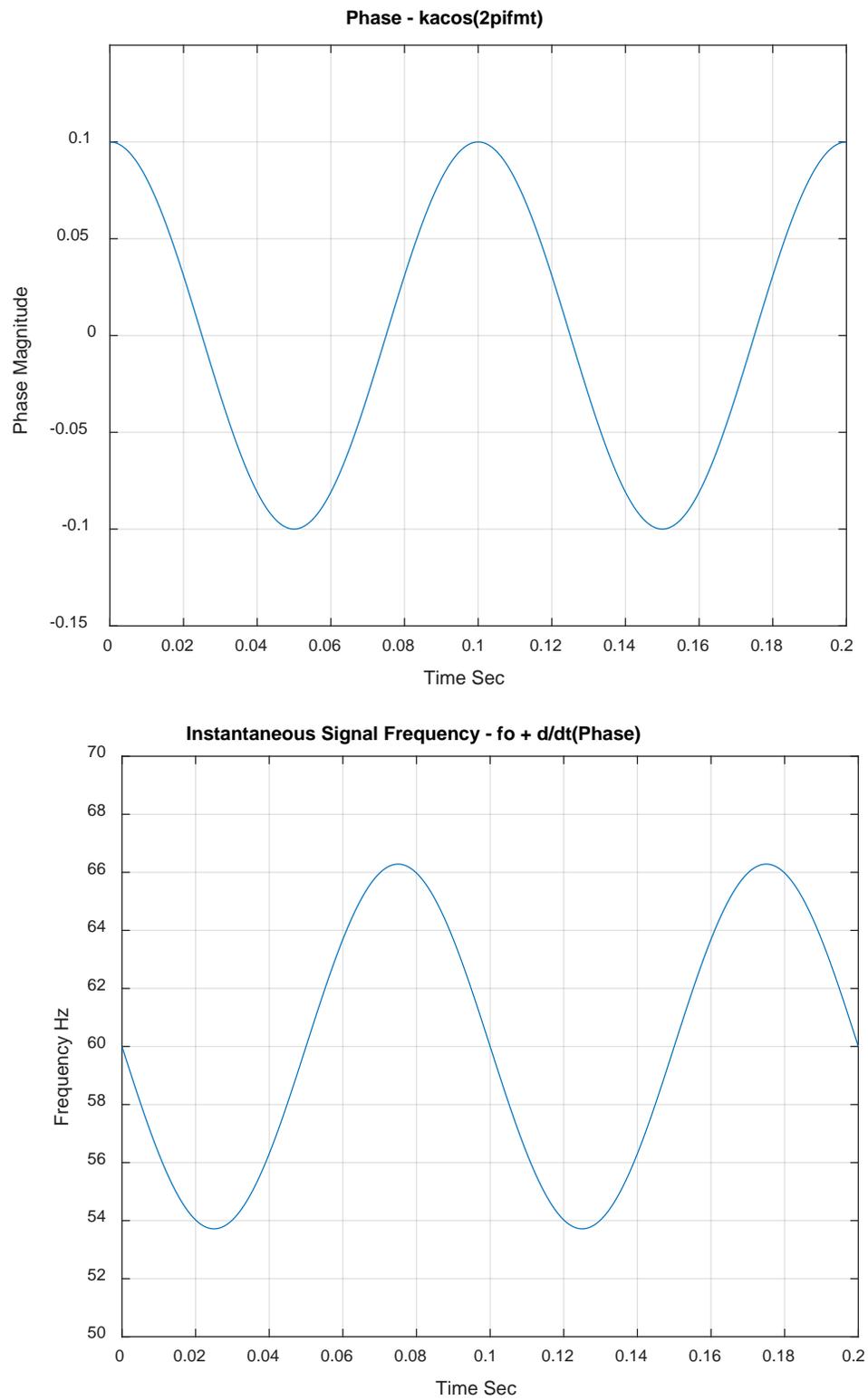


Figure 43: Phase and Frequency

Now consider what happens when a FFT is performed on the instantaneous signal frequency. The figure below shows the magnitude of the frequency content of the FFT. It is important to note that the FFT was performed on an integer number of periods of $f(t)$ and thus there is no leakage.

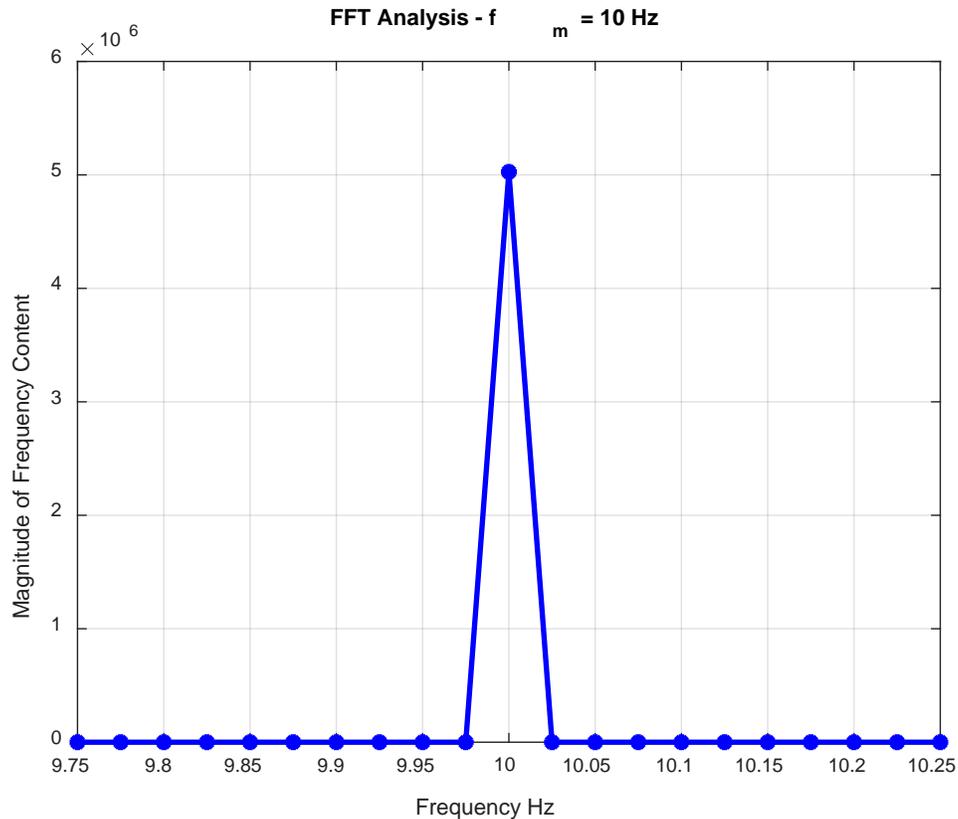


Figure 44: FFT Analysis of Frequency

Looking at this quick analysis sheds light on the purpose of phase modulation testing and the analysis of (25). The analysis performed in (25) is going to produce a magnitude and phase angle relationship between the changing frequency of the input and output of the PMU. Figure 45 below illustrates the input and output of the PMU under phase modulation. The sinusoidal behavior of the phase term in Figure 43 above would translate to a phasor wiggling back and forth around 0 degrees on a phasor diagram as shown below. The corresponding signal frequency

as shown in Figure 43 above could also be represented on a phasor diagram with the phasor wiggling back and forth about 60 Hz.

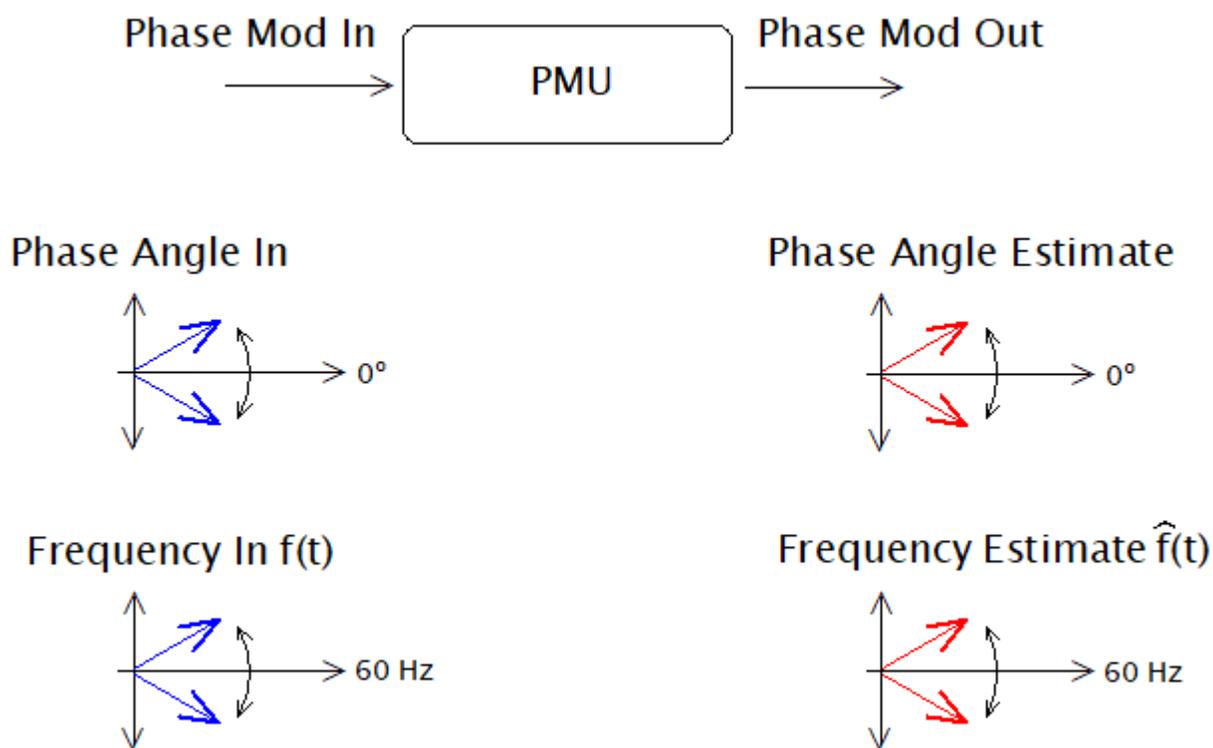


Figure 45: PMU Phase Modulation – Phase and Frequency Tracking

The ultimate goal of (25) is to show how well the PMU tracks the input phasor and thus frequency of a phase modulated input. If the input phasor were plotted together with the output phasor, ideally, they would be identical both in magnitude and phase. Or in other words the input and output would wiggle about zero degrees on top of each other. Similarly, the input and output phasors representing frequency would also wiggle about 60 Hz on top of each other. To estimate phasors PMUs must filter input signals and process the data. Filtering and processing takes time and so the PMU will not be able to track the dynamic input frequency perfectly. This delay will vary depending on modulation frequency, f_m . Lower phase modulation frequencies will cause

the frequency to swing slowly about 60 Hz. Increasing the modulation frequency will cause the input frequency to swing faster about 60 Hz.

6.3. Analysis – Producing the Bode Plot – Deriving Frequency

The term $\hat{f}(t)$ in (25) above is obtained by taking the derivative of PMU phase angle output data. This post processing will introduce errors that need to be corrected for in the final analysis. The derivative will be performed by taking the backward Euler of the PMU phase angle data. The following block diagram summarizes the implementation of the backward Euler. To find the delay associated with this process it is useful to think of the backward Euler as an FIR filter.

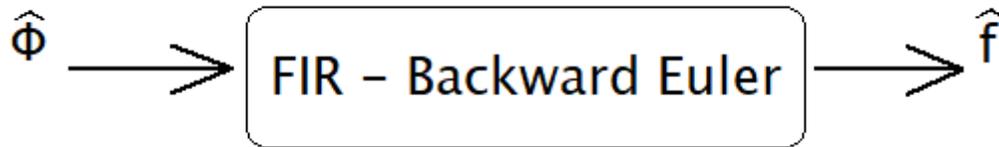


Figure 46: Backward Euler as FIR Filter

The following mathematics represent the backward Euler derivative that approximates $\hat{f}(t)$ in (25) above.

$$\hat{f}_k = \left[\frac{\theta_k - \theta_{k-1}}{dt} \right] \quad (28)$$

Performing the Z transform

$$\hat{F}(z) = \frac{1 - z^{-1}}{dt} \quad (29)$$

Evaluating frequency

$$\hat{F}(\Omega) = \left[\frac{1}{dt} \right] [1 - e^{-j\Omega}] = \left[\frac{1}{dt} \right] [1 - \cos\Omega + j\sin\Omega] \quad (30)$$

The phase angle associated with (31) is shown below.

$$\angle(\hat{F}(\Omega)) = \tan^{-1} \left[\frac{\sin\theta}{1 - \cos\theta} \right] \quad (31)$$

Plotting this phase angle from 0 to pi radians per sample yields the following curve. The delay of the FIR backward Euler is related to the slope of this curve.

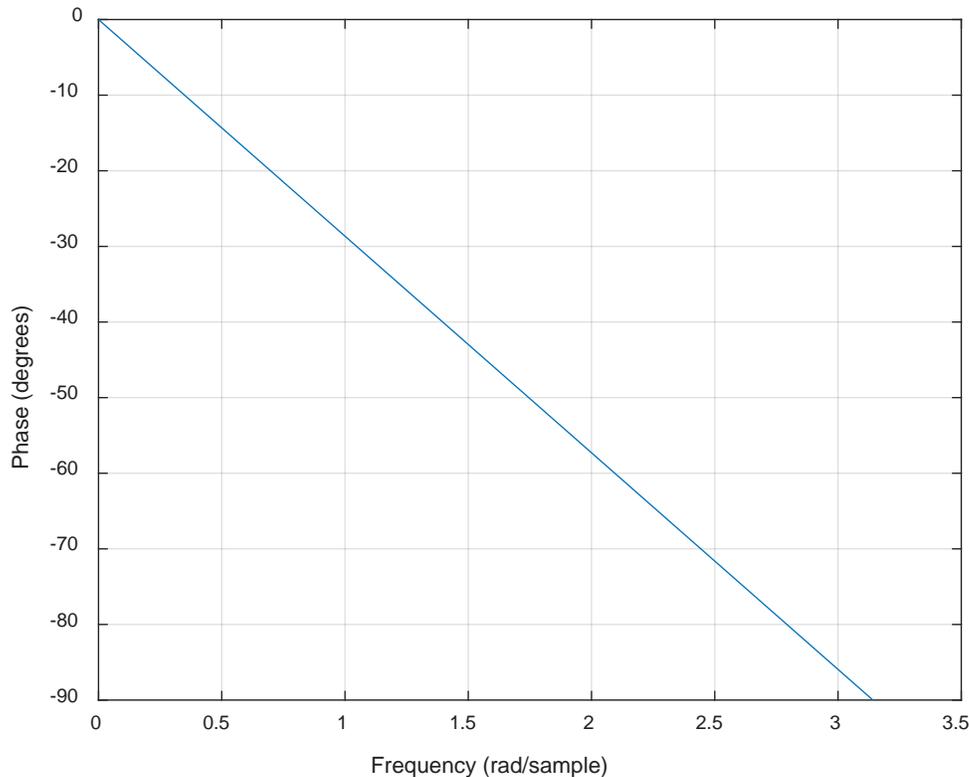


Figure 47: Backward Euler Delay

Finding the slope of this line gives a delay of 0.5 samples. This post processing error will be accounted for in the final analysis as θ_{errBE} .

6.4. Analysis – Producing the Bode Plot – Replacing the FFT with NLLS

Early iterations of test bench analysis software made use of the FFT as shown in (25). Capturing an integer number of periods of each phase modulation burst in $\hat{f}(t)$ proved to be difficult and failing to do so resulted in a significant amount of leakage. This warranted a new solution to produce the frequency content of $\hat{f}(t)$. Non-linear least squares analysis was

performed on $\hat{f}(t)$. This analysis would fit the sinusoid data in $\hat{f}(t)$ to the parameters of the following sinusoidal model.

$$\hat{f}(t) = DC + A\cos(2\pi f_m t) + B\sin(2\pi f_m t) \quad (32)$$

By the nature of the phase modulation test design this model is an excellent fit to the sinusoidal data. The parameters DC , A , and B of the model enter linearly and it is easy to find values of these parameters that minimize the function in (33). A concern would be the parameter f_m , but since f_m is known throughout the test an excellent initial guess can be passed to the minimization algorithm. The model is fitted to the data by adjusting the parameters until the following function is minimized.

$$\sum_{k=1}^N (f_k - \hat{f}_k)^2 \quad (33)$$

Where f_k is the derived output frequency from the PMU and \hat{f}_k is the model form for the output frequency. The subscript k represents the samples of $\hat{f}(t)$. The minimization is accomplished using a gradient based method called Levenberg-Marquardt [4]. The function 'lsqnonlin' is a built-in function in Matlab that accomplishes the NLLS fitting. With the parameters of this model accurately representing the sinusoidal data in $\hat{f}(t)$ it is possible to determine the frequency content of $\hat{f}(t)$. To obtain the magnitude and phase of (32) the following trigonometric operation is performed.

$$\hat{f}(t) = DC + \sqrt{A^2 + B^2} \cos\left(2\pi f_m t + \tan^{-1}\left(\frac{B}{A}\right)\right) \quad (34)$$

6.4.1. Accounting for Initial Phase

An important note about the non-linear least squares analysis is to consider the phase modulation input signal. Each set of plain 60 Hz and the succeeding phase modulation burst is

programmed to have a new time vector starting at zero. To avoid transients in the analysis the first second and last tenth of a second of each phase modulation burst are ignored. This means that the non-linear least squares analysis is being performed on a transient free portion of each phase modulation burst in $\hat{f}(t)$. The non-linear least squares analysis then starts at $t = 3$ seconds for each new burst. Consider a zoomed-in image of the PMU phase angle output. This image shows where the first phase modulation burst occurs with a modulation frequency of 0.1 Hz.

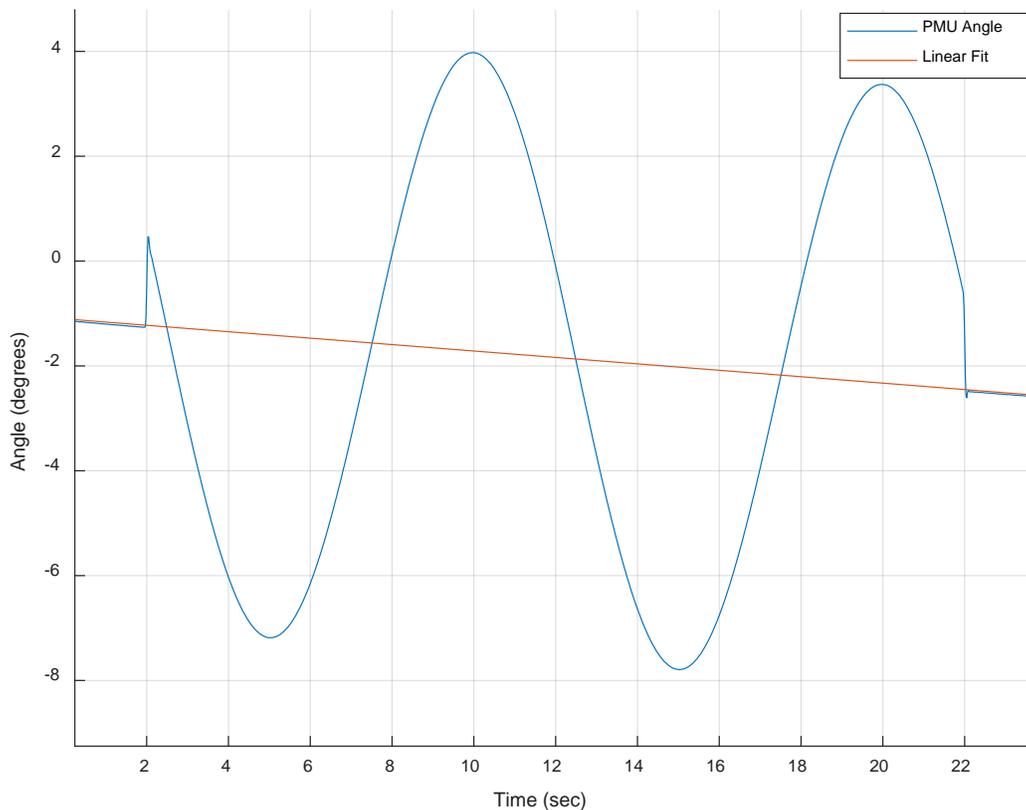


Figure 48: Zoom of PMU 0.1 Hz Phase Modulation Output

Consider equation (26) above where the phase term of the input signal is described by

$$\phi(t) = k_a \cos(2\pi f_m t) \quad (35)$$

When phase modulation starts at two seconds there are some transients in the data. To ensure that each burst of phase modulation is transient free for analysis a second of data is

discarded from the beginning of each burst as well as a tenth of a second at the end of the burst. This means that the non-linear least squares analysis will be performed only on data ranging from three seconds to 21.9 seconds for the 0.1 Hz phase modulation burst shown in Figure 48. At three seconds the phase modulation burst does not start at a peak as would be expected in (34). The phase modulation burst in Figure 48 is then described by the following equation.

$$\phi(t) = k_a \cos(2\pi f_m t - \theta_{errorff}) \quad (36)$$

To adjust for the phase shift $\theta_{errorff}$ in (35) some correction analysis must be done. This is achieved by setting the argument of the cosine function in (35) equal to zero and then solving for $\theta_{errorff}$. The correction factor for this phase shift then becomes.

$$\theta_{errorff} = 2\pi f_m t \quad (37)$$

Each phase modulation burst will be analyzed at three seconds into the burst. This means that $\theta_{errorff}$ above will be analyzed for each burst at $t = 3$ seconds.

6.5. Analysis – Producing the Bode Plot – Accounting for Imperfect Sample Rate

Notice the negative sloping trend of the PMU phase angle output against time as shown in Figure 42. This slope is an effect of the imperfect sample rate discussed in section 3.9. The fundamental frequency of the phase modulation test signal is programmed to be 60 Hz, but this negative sloping trend suggests that the underlying fundamental frequency is actually slower than 60 Hz. In the analysis performed in section 3.9 it was a straight forward task to find the slope in Figure 13. Considering the plot of PMU phase angle data against time as shown in Figure 42 it now becomes more challenging to find the overall negative slope of this data. One solution is to fit a straight line to the data in a least squares sense. This line is illustrated in Figure 42 as a red trace that cuts through the middle of the data. Now finding the true underlying

fundamental frequency is the same as discussed in section 3.9. This error in sample rate will be factored into the final analysis as θ_{errDt} .

6.6. Summary - The Transfer Function Calculation with Correction Factors and the Bode Plot

To summarize the previous sections the following figures will illustrate a block diagram of the phase modulation testing process. Only relevant blocks in the overall process will be included in the diagram. The beginning of the process is shown in Figure 49 below.

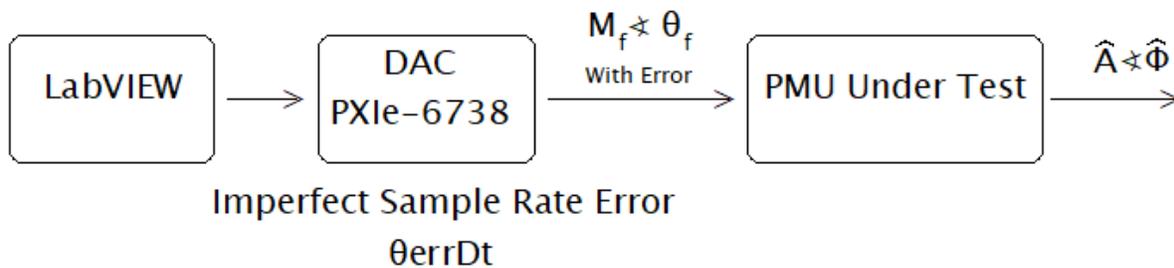


Figure 49: Phase Modulation Testing – PMU Input and Test Bench Error

The block diagram above starts with signal generation in LabVIEW and ends with the A phase magnitude and angle output from the PMU. The imperfect sample rate error as described in section 3.9 above is shown here. With the imperfect sample rate included the true input to the PMU is described as follows.

$$\Delta f(t) = \frac{d}{dt}(\phi(t)) = 2\pi f_m k_a \cos(2\pi f_m t + 90^\circ + \theta_{errDt}) \quad (38)$$

The phasor input to the PMU as depicted in Figure 49 then becomes.

$$\begin{aligned} M_f &= 2\pi f_m k_a \\ \theta_f &= 90^\circ + \theta_{errDt} \end{aligned} \quad (39)$$

The following figure will highlight the post processing of the PMU output phase angle. Accumulated post processing errors will be shown with their corresponding block.

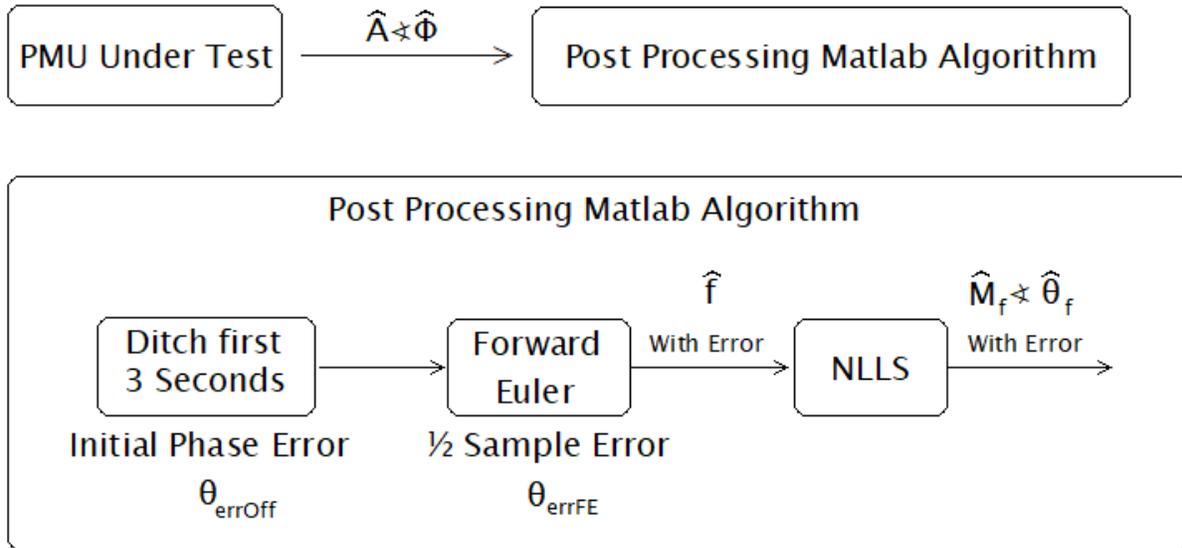


Figure 50: Phase Modulation Testing – PMU Output and Post Processing Errors

The post processing shown in Figure 50 ends with the output from the non-linear least squares analysis. The output from the NLLS algorithm contains the true phase relationship that this testing wishes to discover but is offset by post processing error. The error is accumulated in the phase angle and can therefore be described as follows.

$$NLLS \text{ Output Angle} = [\hat{\theta}_f + \theta_{errOff} - \theta_{errFE}] \quad (40)$$

Previous sections have described how these errors in both the test bench accuracy and post processing techniques can be quantified in phase. The final analysis will account for all errors in the form of correction factors. These correction factors will subtract the accumulated phase error where appropriate. The overall computation for the frequency response is summarized as follows.

$$G(f_m) = \frac{\hat{M}_f \nabla [\hat{\theta}_f - \theta_{errOff} + \theta_{errFE}]}{M_f \nabla [\theta_f - \theta_{errDt}]} \quad (41)$$

The analysis software developed for phase modulation testing, ‘AnalyzeData.m’, uses a for loop to step through the post processing as shown in Figure 50 for each phase modulation

burst. The post processing algorithm produces a single point on the Bode plot for each iteration of the for loop. Once all phase modulation bursts have been analyzed the magnitude and phase results are plotted on a logarithmic scale to produce the Bode plot. These results are shown below.

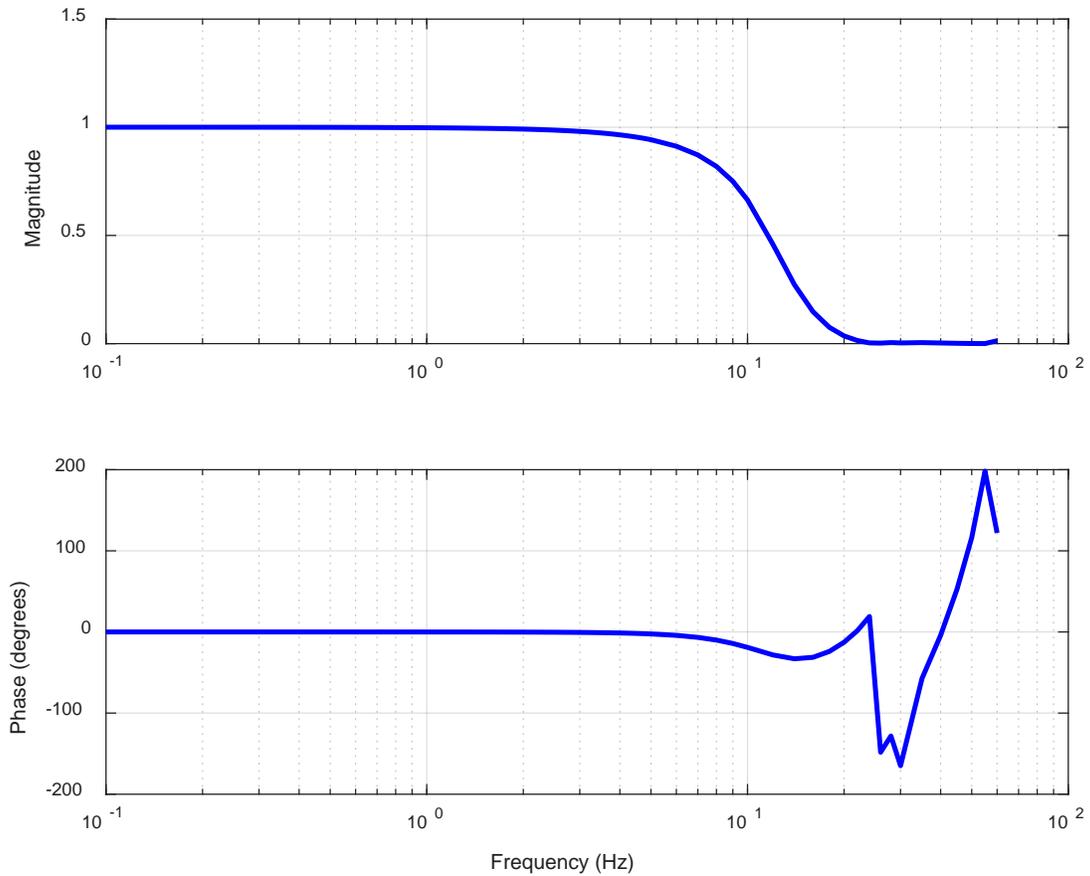


Figure 51: PMU Reported Bode Plot

As explained at the beginning of section 6, this Bode plot is built from PMU output data and knowledge about the input. Therefore, this plot should be thought of as the frequency response **as reported by the PMU**. From section 5 it is known that filter group delay has been compensated for by the time stamping within the FIR filter window. This should be taken into

consideration when looking at the shape of the phase Bode plot and will be expanded on further in the discussion section.

7. Discussion

The results from section 6 show a Bode plot for the PMU under test. This however is not the real time frequency response that this thesis wishes to find. Analysis of step test results in section 5 factored with the analysis of the phase modulation testing results in section 6 bring up a new discussion. The topic of reported frequency response versus the real time frequency response of PMUs will be covered in the following subsections.

7.1. The Bode Plot

The slope of the phase Bode plot shows that there is very little output delay from the PMU in the frequency band of 0 Hz up to 10 Hz. This should be expected as PMU manufacturers account for group delay in the FIR filtering process by time stamping around the middle of the filter window. Section 5 showed that a PMUs reported response to step inputs make the device appear to behave non-causally. This non-causal behavior was proved to be an artifact of the timestamping within the FIR filter window and that it is possible to determine the timestamping advance as shown in section 5. The C37.118 standard allows for timestamping this way in the form of measurement reporting latency guidelines as discussed in section 2.2. The C37.118 standard discusses how this time stamping compensates for the group delay that is inherent in FIR filtering. Manufactures can then employ different phasor estimation filtering algorithms in their PMUs. When compared side by side with their competitors the PMUs will appear to behave similarly. The C37.118 standard discusses the benefits of group delay compensation by unifying PMU performance. The C37.118 standard puts limits on PMU manufactures to be within the accepted TVE, FE, and RFE guidelines while also meeting the requirements for measurement reporting latency. These standard requirements put limits on FIR filtering and timestamping, while also setting guidelines for filter performance. In certain real time applications researchers

may wish to verify PMU performance. This is especially true if the manufacturer does not claim compliance to the C37.118 standard.

7.2. PMU Dynamics for Real Time Applications

PMU applications may require detailed knowledge about the devices real time behavior characteristics. This knowledge includes the group delay compensation or how far advanced the timestamping is within the FIR filter window. By performing the testing described in section 5 PMU users will have access to this timestamping information. Adding the group delay compensation back into to the PMU Bode plot produced in section 6 will introduce a constant slope to the phase Bode shown in Figure 51. The following figure illustrates this new real time Bode comparatively against the old Bode.

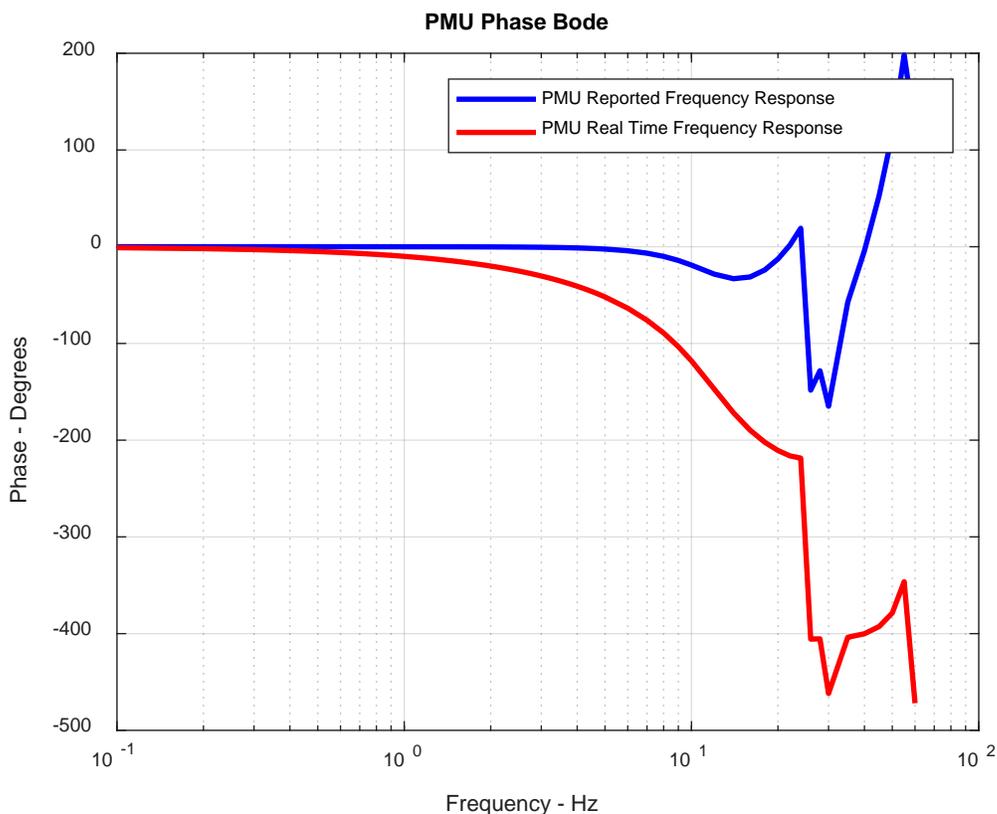


Figure 52: PMU Frequency Response – Reported vs Real Time

This new Bode will give PMU users a detailed real time frequency response characteristic to design their specific application around. Noted power system dynamics expert Dr. Dan Trudnowski has outlined PMU performance requirements that are specified for real time PMU applications. As a final note this thesis will discuss how Dr. Trudnowski's phase requirements compare in relation to the FE requirements of the C37.118 standard. Lastly, Dr. Trudnowski's phase requirements will be compared against the PMU under test.

7.2.1. Dr. Trudnowski's Phase Requirements Compared to the C37.118 Standard Frequency Error Requirements

The focus of this section will be to compare the recommended phase requirements outlined by Dr. Trudnowski against the corresponding frequency error requirements set forth by the C37.118 standard. The C37.118 standard allows for a $FE = 0.3$ Hz for M class PMUs and $FE = 0.06$ Hz for P class PMUs. This FE assumes a modulation frequency of $f_m = 5$ Hz for M class PMUs and $f_m = 2$ Hz for P class PMUs. The following equation shows how this error in frequency is translated to an error in phase.

$$FE = |f - \hat{f}| \quad (42)$$

$$FE = 2\pi f_m k_a \cos(2\pi f_m t) - 2\pi f_m k_a \cos(2\pi f_m t + \varphi_e)$$

The analysis performed in appendix A is summarized in the following table.

Table IV: Appendix A Summary

	M Class PMU	P Class PMU
Allowable Latency in Time	5/60 seconds	2/60 seconds
Allowable Latency in Phase	-60° at $f_m = 2$ Hz	-24° at $f_m = 2$ Hz
Allowable FE	0.3 Hz at $f_m = 5$ Hz	0.06 Hz at $f_m = 2$ Hz
φ_e reported $f_m = 5$ Hz	-5.473°	NA
φ_e reported $f_m = 2$ Hz	-2.189°	-2.736°
φ_e real time $f_m = 2$ Hz	-62.189°	-26.736°

These results show that the P class PMU is closest to meeting Dr. Trudnowski's phase requirement of -20° , and is off by -6.736° .

7.2.2. Dr. Trudnowski's Requirements Compared to the PMU Under Test

To analyze the PMU under test and compare it to the requirements of Dr. Trudnowski the real time Bode plot will be analyzed. The following figure shows how the PMU under test performs at a modulation frequency of 2 Hz.

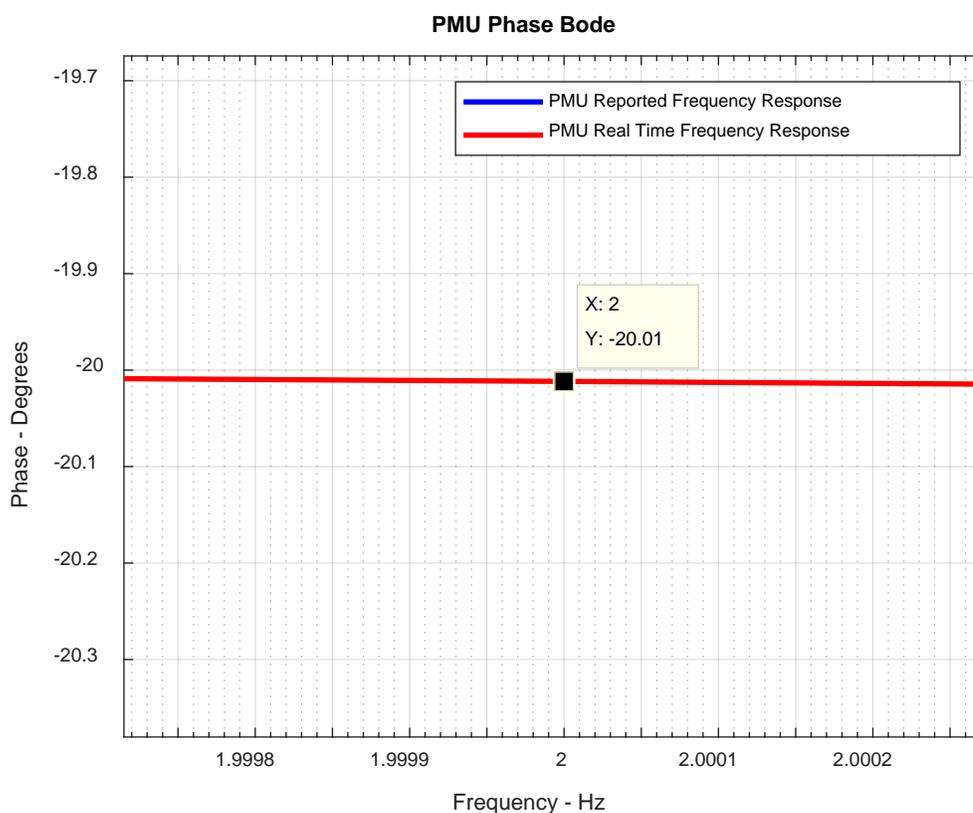


Figure 53: PMU Under Test versus Dr. Trudnowski's Real Time Requirements

As can be seen in Figure 53 above the PMU under test performs well and has a phase of -20.01 degrees at a phase modulation frequency of 2 Hz. This means the PMU under test would come very close to passing Dr. Trudnowski's phase requirement.

8. Conclusions

The contents of this thesis describe how to build a PMU test bench, how to use this test bench to test a PMU with dynamic input signals, and how to analyze PMU test data to produce both reported PMU responses and real time PMU responses. The results from this testing are specific to the PMU under test for this thesis but the principles and practices can be applied to any PMU that measures voltage phasors. Power system researchers who wish to perform testing on PMUs before implementing them in their system will find this knowledge helpful and informative. More specifically, researchers who are interested in real time PMU applications may wish to use this thesis as a road map, or a starting point for their own PMU testing and characterization.

9. References Cited (or Bibliography)

- [1] F. Aminifar, M. Fotuhi-Firuzabad, A. Safdarian, A. Davoudi and M. Shahidehpour, "Synchrophasor Measurement Technology in Power Systems: Panorama and State-of-the-Art," in *IEEE Access*, vol. 2, pp. 1607-1628, 2014
- [2] J.D. Glover, T.J. Overbye, M.S. Sarma, *Power System Analysis & Design*, Stamford, CT: Cengage Learning, 2012
- [3] D. N. Kosterev, C. W. Taylor and W. A. Mittelstadt, "Model validation for the August 10, 1996 WSCC system outage," in *IEEE Transactions on Power Systems*, vol. 14, no. 3, pp. 967-979, Aug 1999.
- [4] Moré, J. J. "The Levenberg-Marquardt Algorithm: Implementation and Theory." Numerical Analysis, ed. G. A. Watson, Lecture Notes in Mathematics 630, Springer Verlag, 1977, pp. 105–116.
- [5] A.G. Phadke, J.S. Throp, *History and Applications of Phasor Measurements*. Presented at IEEE PES Power Systems Conference and Exposition, 2006
- [6] A.G. Phadke, J.S. Throp, *Synchronized Phasor Measurements and Their Applications*. Cham: Springer. 2017
- [7] Brian J. Pierre, Felipe Wilches-Bernal, Ryan T. Elliott, David A. Schoenwald, Jason C. Neely, Raymond H. Byrne, Daniel J. Trudnowski, "Simulation results for the pacific DC intertie wide area damping controller", *Power & Energy Society General Meeting 2017 IEEE*, pp. 1-5, 2017, ISSN 1944-9933.
- [8] Sharma, Rajesh & G.M. Dhole, (2016). Wide Area Measurement Technology in Power Systems. *Procedia Technology*. 25. 718-725. 10.1016/j.protcy.2016.08.165.
- [9] IEEE Standard for Synchrophasor Measurements for Power Systems," in *IEEE Std C37.118.1-2011(Revision of IEEE Std C37.118-2005)*, vol., No., pp. 1-61, Dec. 28 2011 doi: 10.1109/IEEESTD.2011.6111219

10. Appendix A: Dr. Trudnowski's Phase Requirements Compared to C37.118 Standard FE Requirements

For phase modulation frequencies of $f_m = 5$ Hz for M class PMUs and $f_m = 2$ Hz for P class PMUs, the C37.118 standard requires that the frequency error, FE, be no more than 0.3 Hz for M class PMUs and 0.06 Hz for P class PMUs. The following mathematics will translate the C37.118 standard FE to a phase error. The allowable phase error will then be compared against the real time phase requirements outlined by Dr. Trudnowski.

Frequency error is defined as follows.

$$FE = |f - \hat{f}| \quad (43)$$

Substituting values for, \hat{f} and f .

$$FE = (0.1)(2\pi f_m) \cos(2\pi f_m t) - (0.1)(2\pi f_m)(1 - M_e) \cos(2\pi f_m t + \varphi_e) \quad (44)$$

where M_e and φ_e are the error terms in the estimated output frequency \hat{f} . The output frequency estimation, \hat{f} , assumes that the PMU is a linear system, or that the output frequency of the PMU is going to have a scaled magnitude of the input frequency and a phase shift from the input frequency. If all error is assumed to occur in phase, φ_e , then the error in magnitude, M_e , is set to zero. This will give an upper limit to the amount of phase error that can occur with a maximum FE of 0.3 Hz for M class PMUs and 0.06 Hz for P class PMUs. This will give the following equation.

$$FE = (0.1)(2\pi f_m) [\cos(2\pi f_m t) - \cos(2\pi f_m t + \varphi_e)] \quad (45)$$

Using the following trig identity.

$$\cos(\alpha + \beta) = \cos(\alpha) \cos(\beta) - \sin(\alpha) \sin(\beta) \quad (46)$$

Substituting trig identity

$$FE = (0.1)(2\pi f_m) [\cos(2\pi f_m t) - (\cos(2\pi f_m t) \cos(\varphi_e) - \sin(2\pi f_m t) \sin(\varphi_e))]$$

$$\frac{FE}{(0.1)(2\pi f_m)} = \cos(2\pi f_m t) - \cos(2\pi f_m t) \cos(\varphi_e) + \sin(2\pi f_m t) \sin(\varphi_e)$$

$$\frac{FE}{(0.1)(2\pi f_m)} = \cos(2\pi f_m t) [1 - \cos(\varphi_e)] + \sin(2\pi f_m t) \sin(\varphi_e)$$

Think of $[1 - \cos(\varphi_e)]$ and $\sin(\varphi_e)$ as amplitude terms. The right-hand side above can therefore be written as one cosine with a new magnitude and phase angle.

$$\frac{FE}{(0.1)(2\pi f_m)} = \left[\sqrt{[1 - \cos(\varphi_e)]^2 + [\sin(\varphi_e)]^2} \right] \cos(2\pi f_m t + \delta) \quad (47)$$

The goal is to find φ_e when the right-hand side of the equation reaches a maximum, therefore $\cos(2\pi f_m t + \delta)$ will only be considered when equal to 1.

$$\frac{FE}{(0.1)(2\pi f_m)} = \left[\sqrt{[1 - \cos(\varphi_e)]^2 + [\sin(\varphi_e)]^2} \right] \quad (1)$$

$$\left[\frac{FE}{(0.1)(2\pi f_m)} \right]^2 = [1 - \cos(\varphi_e)]^2 + [\sin(\varphi_e)]^2$$

$$\left[\frac{FE}{(0.1)(2\pi f_m)} \right]^2 = 1 - 2 \cos(\varphi_e) + \cos^2(\varphi_e) + \sin^2(\varphi_e)$$

$$\left[\frac{FE}{(0.1)(2\pi f_m)} \right]^2 = 2 - 2\cos(\varphi_e)$$

$$1 - \frac{\left[\frac{FE}{(0.1)(2\pi f_m)} \right]^2}{2} = \cos(\varphi_e)$$

$$\varphi_e = \cos^{-1} \left[1 - \frac{\left[\frac{FE}{(0.1)(2\pi f_m)} \right]^2}{2} \right]$$

Substituting $f_m = 5$ Hz for M class, $f_m = 2$ Hz for P class, $FE = 0.3$ Hz for M class, and $FE = 0.06$ Hz for P class yields the following maximum allowable phase

$$\begin{aligned} \varphi_{e \text{ Mclass}} &= -5.473^\circ \text{ for } f_m \text{ of } 5 \text{ Hz} \\ \varphi_{e \text{ Pclass}} &= -2.736^\circ \text{ for } f_m \text{ of } 2 \text{ Hz} \end{aligned} \quad (48)$$

Assuming the FIR filtering is linear in phase it is possible to use linear interpolation to determine φ_e at $f_m = 2$ Hz for M class PMUs.

$$\begin{aligned}\varphi_{e\ Mclass} &= -2.189^\circ \text{ for } f_m \text{ of } 2 \text{ Hz} \\ \varphi_{e\ Pclass} &= -2.736^\circ \text{ for } f_m \text{ of } 2 \text{ Hz}\end{aligned}\quad (49)$$

This result is based on the C37.118 standard's frequency error. This error does not take into account the advanced time stamping being performed by the FIR filtering process. The C37.118 standard allows for a maximum reporting latency of $2/F_s$ for P class PMUs, and $5/F_s$ for M class PMUs. With a reporting rate of $F_s = 60$ sps, this corresponds to a maximum of $T_L = 2/60$ seconds of advance for P class PMUs, and $T_L = 5/60$ seconds of advance for M class PMUs. The following equation will convert this advance timestamping into a phase quantity in degrees.

$$\begin{aligned}\theta_L &= T_L 2\pi f_m \left(\frac{180}{\pi}\right) \\ \theta_{L\ Pclass} &= -\left(\frac{2}{60}\right)(2\pi)(2)\left(\frac{180}{\pi}\right) = -24^\circ \\ \theta_{L\ Mclass} &= -\left(\frac{5}{60}\right)(2\pi)(2)\left(\frac{180}{\pi}\right) = -60^\circ\end{aligned}\quad (50)$$

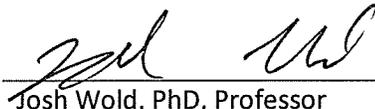
Therefore, total phase at $f_m = 2$ Hz is given by.

$$\begin{aligned}\varphi_{e\ P\ Class} &= -24^\circ - 2.736^\circ = -26.736^\circ \\ \varphi_{e\ M\ Class} &= -60^\circ - 2.189^\circ = -62.189^\circ\end{aligned}\quad (51)$$

This result falls short of Dr. Trudnowski's phase requirement of -20° by -6.736° for P Class PMUs and -42.189° for M class PMUs. The results of this analysis are summarized in Table IV in section 7.2.1.

SIGNATURE PAGE

This is to certify that the thesis prepared by Joe Perigo entitled "Phasor Measurement Unit Testing and Characterization" has been examined and approved for acceptance by the Department of Electrical Engineering, Montana Tech of The University of Montana, on this 1st day of May, 2018.



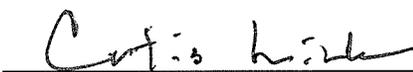
Josh Wold, PhD, Professor
Department of Electrical Engineering
Chair, Examination Committee



Dan Trudnowski, PhD, Dean
School of Mines & Engineering
Member, Examination Committee



Matt Donnelly, PhD, Professor
Department of Electrical Engineering
Member, Examination Committee



Curtis Ling, PhD, Professor
Department of Freshman Engineering Program
Member, Examination Committee