B. TECH. PROJECT REPORT

On

Development of micro Phasor Measurement Unit

BY

Ankita Bansal

DISCIPLINE OF ELECTRICAL ENGINEERING

INDIAN INSTITUTE OF TECHNOLOGY INDORE

NOV 2017

Development of micro Phasor Measurement Unit

A PROJECT REPORT

Submitted in partial fulfillment of the requirements for the award of the degrees

of
BACHELOR OF TECHNOLOGY
in

ELECTRICAL ENGINEERING

Submitted by:
Ankita Bansal

Guided by:
Dr. Trapti Jain, Associate Professor EE

INDIAN INSTITUTE OF TECHNOLOGY INDORE
December, 2017

CANDIDATE'S DECLARATION

I hereby declare that the project entitled **"Development of micro Phasor Measurement Unit"** submitted in partial fulfillment for the award of the degree of Bachelor of Technology in 'Electrical Engineering' completed under the supervision of **Dr. Trapti Jain, Associate Professor Electrical Engineering,** IIT Indore is an authentic work.

Further, I declare that I have not submitted this work for the award of any other degree elsewhere.

Ankita Bansal

140002008

Discipline of Electrical Engineering

CERTIFICATE by BTP Guide

It is certified that the above statement made by the students is correct to the best of my knowledge.

Dr. Trapti Jain

Associate Professor Electrical Engineering

IIT INDORE

Preface

This report on "Development of micro Phasor Measurement Unit" is prepared under the guidance of Dr. Trapti Jain, Associate Professor, Electrical Engineering, IIT INDORE.

Through this report I have tried to explain the algorithm which can be used in Indian power system in distribution network for monitoring the system in a better manner.

I have tried to the best of my abilities and knowledge to explain the algorithm in detail. I have also added tables and figures to make it more illustrative.

Ankita Bansal
B.Tech. IV Year
Discipline of Electrical Engineering
IIT Indore

Acknowledgements

Apart from the efforts made by anyone, the success of project depends largely on the encouragement and guidelines of many others. I take this opportunity to express my gratitude to the people who have been instrumental in the successful completion of this project.

I would like to express sincere gratitude to Dr. Trapti Jain, Associate professor and HOD at IIT INDORE for giving me an opportunity to work on this interesting project.

I would like to express indebtedness to Mr. Vinay Kumar Tiwari, Phd scholar at IIT INDORE. This work was not possible without his valuable guidance.

Ankita Bansal
B.Tech. IV Year
Discipline of Electrical Engineering
IIT Indore

Abstract

Phasor frequency, magnitude, and angle describing a sinusoidal signal are widely used as critical variables in algorithms and performance indices in many power system applications.

For estimating the phasor magnitude and angle in real time at transmission level, we use phasor measurement unit (PMU).

For monitoring the system in a better way, we need to have information about magnitude, frequency and phase angle at distribution level also. So we need a device similar to PMU that can be placed at distribution level.

In this report, an algorithm for estimating phasor of any electrical waveform at the distribution level is presented. The problem with estimating the phasor at distribution level is that it is very sensitive to any noise and abrupt change in the waveform and also the voltage angle resolution at distribution level is much less than at transmission level. So we need to place several micro PMUs at distribution level.

This algorithm is based on Discrete Wavelet Transform (DWT) and Hilbert Transform.

Table of Contents

List of Figures

List of Tables

1. Introduction

Indian grid is the fourth biggest in the world and it is operating closer to its stability limits with a total generation capacity of more than 250GW. The non-conventional energy sources like, solar and wind power are rapidly increasing and generates 28% of total installed capacity and conventional generating stations generates the remaining72%. So it is essential to monitor the system as much accurately as possible otherwise there will be situation of instability and reverse power flow. Monitoring actually means to have an accurate knowledge of what the voltage is, what the frequency is and if the frequency changes then what the rate of change of frequency is for voltages and currents in the system.

We can analyze all this if we know the phasor of the signal. And we can calculate the phasor using a device called as Phasor Measurement Unit (PMU).

1.1 Phasor Measurement Unit:

PMUs are devices which sample substation voltages and currents through power and current transformers. The device then performs signal processing computations to generate voltage and current phasors referenced to the global time reference (using GPS).

The concept of PMU was introduced in 1980s. The need of synchronized sampling first appeared in the design of protection systems. In 1988, the first idea of PMU was introduced. Lots of demo projects focusing on the applications of PMU at transmission level were developed since then.

PMU architecture: Any PMU architecture consists of 4 basic components

a. Micro processing Unit: Component which estimates the phasor.
b. GPS: Provides a time stamp for the signal.
c. Analog to digital converter (A/D converter): The input to the PMU is an analog signal but micro processer needs digital input so we need an analog to digital converter.
d. Filter: To prevent aliasing, we need to filter input signal before feeding it into A/D converter.

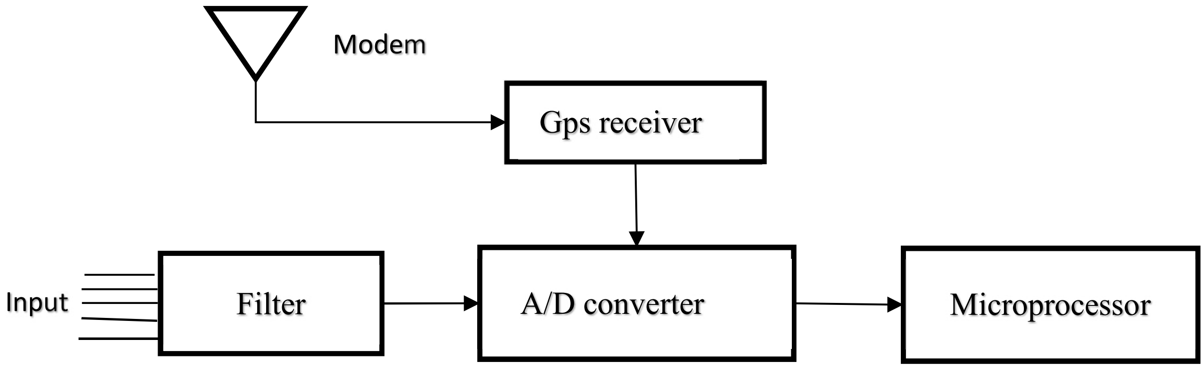


Fig 1.1 PMU architecture

1.2 Phasor Representation of Waveform:

Phasor is a way to describe synthetically a sinusoidal signal in AC networks, with the help of a complex number, made by its rms value and its phase angle.

The classical mathematical definition of the phasor relies on a generic AC signal $x(t)$.

Let $x(t) = Xm\cos(wt + \varphi)$ ..... Eq (1)

Where Xm is the signal peak value, w=2πf is the system angular frequency and φ is the initial phase of the signal, which depends on the definition of the time scale.

The corresponding phasor representation is

$X = \frac{Xm}{\sqrt{2}} * (\cos\varphi + i\sin\varphi) = Xr + i\,Xi$ ..... Eq (2)

Where Xr and Xi are real and imaginary components of the complex phasor representation.

The phasor angle is strictly connected to the initial time instant used as a reference (putting t=0 in Eq 1). For this reason, the phase angle is intrinsically a relative concept that has to be referred correctly to use initial time when a measurement of the phasor is needed.

1.3 Need of Micro PMU at distribution level:

As of now, the PMU is placed at transmission level only. But the distribution network of the power system is also very complex as there are many distributed energy resources now (continuously increasing).

With increasing distributed energy resources it is possible that the system may become unstable. So if we place PMU at distribution level, it will be helpful in Stability analysis & monitoring.

It will also be helpful in Protection, State estimation of power system, Protective relaying, Harmonics detection,Voltage/ Var/ Watt Control and to locate the fault.

But we cannot directly place the PMU at distribution level as the voltage angle resolution at distribution level is much less than that at transmission and also the waveform at distribution level is much affected by various errors and white noises. So we need to develop new algorithm for calculating phasor at distribution level which can calculate accurate phasor even in presence of noises and errors.

2. Theory and Design

We want to design an algorithm which should be able to calculate magnitude and phase accurately even in case of deviation of frequency from its nominal value, in presence of frequency harmonics and also in case of presence of noise in the signal.

When it comes to signal processing Fourier transform is usually our preferred candidate because its computational time is very less, it is least complex among other transforms and it is easy to implement as well but when it comes to processing real time data, its accuracy becomes very poor. For this reason we need to use wavelet transform (Discrete) which in spite of being poor in terms of computational time, complexity and implementation, has a very good accuracy. There are few other transforms also available but they are very complex and we cannot implement them on FPGA, they require a dedicated system for their operation. In this manner wavelet is best suited for our design.

Parameters	**FFT**	**Wavelet Transform**	**Other algorithms Such as MUSIC**
Computational Time	Least	> FFT	greatest
Computational Complexity	Least	>FFT	greatest
Required Resources for hardware implementation	Least	>FFT	greatest
Ease of implementation	Easiest	Complex	Highly Complex
Accuracy for real time data	Extremely less	Good	Good

Table 1. Comparison of wavelet transform with other transforms

2.1 Wavelet Transform:

Wavelet transform is similar to Fourier transform. The difference between the two is that Fourier transform decomposes the signal in terms of sinusoids whereas wavelet transform decomposes in terms of wavelets. Wavelets can either be symmetric or asymmetric, regular or irregular, smooth or sharp. We can say that sinusoid is a type of wavelet. Generally, wavelet transform of any signal can be expressed as

$$F(a,b) = \int_{-\infty}^{\infty} f(x)\, \psi^*_{(a,b)}(x)\, \mathrm{d}x$$

Where the * is the complex conjugate symbol and function ψ is some function.

There are two types of wavelet transform: Continuous wavelet transform (CWT) and Discrete wavelet transform (DWT).

We use CWT for signal analysis such as self-similarity analysis, time frequency analysis and DWT is used for signal analysis and signal processing such as noise reduction, peak detection and data compression.

Flow chart for calculating DWT and Flow chart of the proposed algorithm are given in Fig 2.1 and Fig 2.2 respectively.

In DWT after each step of filtering signal is decimated by two. In this way after each stage of filtration and decimation, the signal length is reduced to half.
After calculating DWT of the signal we get the fundamental component of the signal at the end which has no noise. We then use Hilbert transform to calculate the phase of the signal.

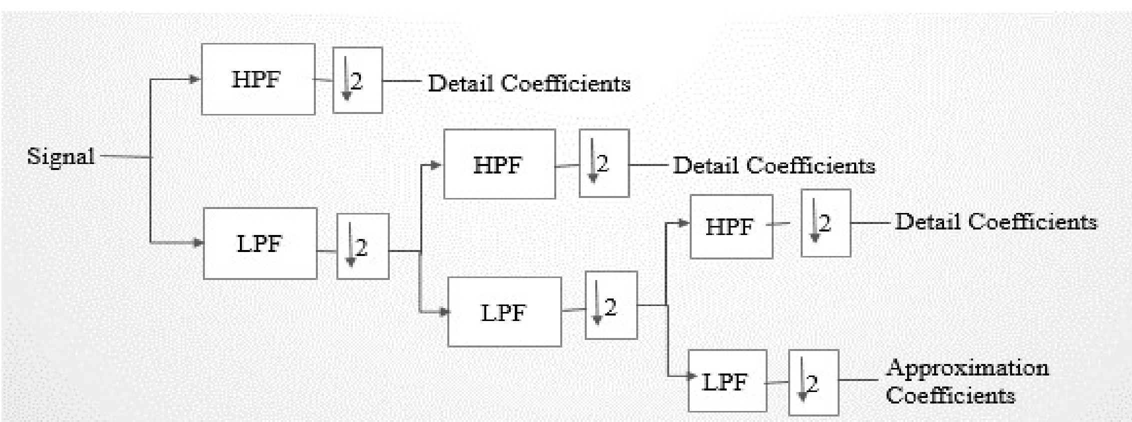


Fig 2.1 Flow chart for calculating DWT

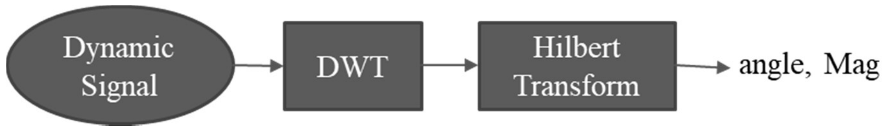


Fig 2.2 Flow chart for proposed algorithm

2.2 Hilbert Transform:

The Hilbert transform gives us the harmonic conjugate of a given signal in Fourier analysis. It is also known as harmonic analysis.

In time domain, Hilbert transform of a signal can be seen as convolution of the signal s(t) with ($1/\pi t$). But since $1/\pi t$ is non-integrable, we define Hilbert transform in frequency domain as

$H\{s(t)\}= S(w)*(-j*sgn(w))$

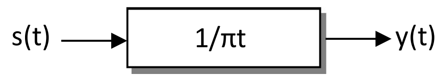
Where H{s(t)} denotes the Hilbert transform of signal s(t) and S(w) denotes Fourier transform of s(t) and sgn(w) is Signum function defined as:

$$sgn(w)= \begin{cases} 1, & \text{for } w>0 \\ 0, & \text{for } w=0 \\ -1, & \text{for } w<0 \end{cases}$$

Hence H{s(t)} can be written as:

$$H\{s(t)\}= \begin{cases} j* S(w) & \text{for } w<0 \\ 0 & \text{for } w=0 \\ -j*S(w) & \text{for } w>0 \end{cases}$$

From above expression for Hilbert transform of s(t) in frequency domain, we can say that Hilbert transform actually shifts the phase of the signal by 90^0 in frequency domain and has no effect on magnitude of the signal. In this way, the energy of signal s(t) and energy of its Hilbert transform is same and also the signal and its Hilbert transform are orthogonal to each other.



y(t) is Hilbert transform of s(t).

And if we take Hilbert transform of y(t), then we get –s(t) as output. Now if we take Hilbert transform of –s(t) then we will get –y(t) at output and again if we calculate Hilbert transform of –y(t) then we will get s(t) at output. Which means we get the signal s(t) back when we take Hilbert transform 4 times. That's why Hilbert transform is also known as Quadrature filter.

In signal analysis we define Hilbert transform of signal s(t) as $\widehat{s(t)} = s(t) + j * h(s(t))$ where h(s(t)) is actual Hilbert transform of s(t).

Let s(t) = A*sin(wt+ φ) then Hilbert transform $\widehat{s(t)}$ of the signal s(t) will be:

$$\widehat{s(t)} = A * \sin(wt + \varphi) + j * h(\mathrm{A} * \sin(wt + \varphi))$$

On calculating, h(A*sin(wt+ φ)) will be equal to A*sin(wt+ $\varphi - \pi/2$) or –A*cos(wt+ φ) for w>0

Hence $\widehat{s(t)} = \mathrm{A} * \sin(wt + \varphi) - A * j * \cos(wt + \varphi)$

Angle of the signal s(t) can be calculated by calculating angle of $\widehat{s(t)}$

Angle of $\widehat{s(t)}$ = $\tan^{-1}(-\frac{\cos(wt+\varphi)}{\sin(wt\quad\varphi)})$ = wt+ φ- $\pi/2$

Angle of s(t) = angle of $\widehat{s(t)}$ + $\pi/2$

3. Analysis, Discussion and Results

In our analysis, approximation coefficients at the 3rd level of DWT corresponds to the fundamental component of the signal.

Let say these coefficients are A (n), n=1, 2, …. N, where N represents number of coefficients.

3.1 Magnitude calculation:

Since A (n) represents fundamental component of the signal so the energy of the fundamental component will lie in these coefficients.

Thus rms value of the signal can be calculated as:

$$rms = \sqrt{\sum_n \mathrm{A}^2 / N}$$

3.2 Phase Calculation:

Let Hilbert transform of coefficients A (n) are given by H (n) = H_r (n) + j* H_i (n),

Angle of the signal can be calculated as: $angle = \tan^{-1} Hi/Hr$

But this angle corresponds to the total angle ($wt + \varphi$) of signal (Eq 1) at any instant t.

Hence, phase φ can be calculated as: $\varphi = angle - wt$

The nominal frequency of operation in Indian power system is 50 Hz. So in the design we keep f to be 50 Hz. In this way, the angle φ can be calculated as:

$$\varphi = angle - 2\pi * 50 * t$$

3.3 Phase angle correction:

Sometimes due to some unavoidable reasons, frequency may deviate from its nominal value and it can vary between 49.5 Hz to 50.5 Hz. In this case, we will not get correct result using above equation.

To estimate correct phase in this situation we first calculate φ keeping f=50 Hz and then correct the phase according to the deviation in frequency f from its nominal value. The algorithm for correcting phase in case of deviation in frequency is shown in Fig 3.1.a and final algorithm is as shown in Fig 3.1.b

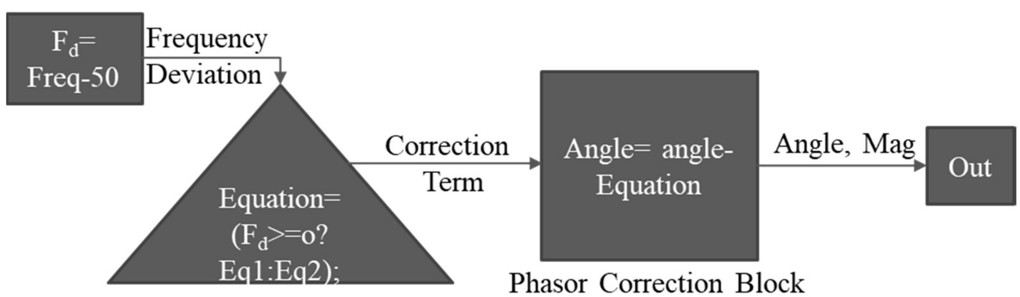


Fig 3.1.a Algorithm for correcting phase

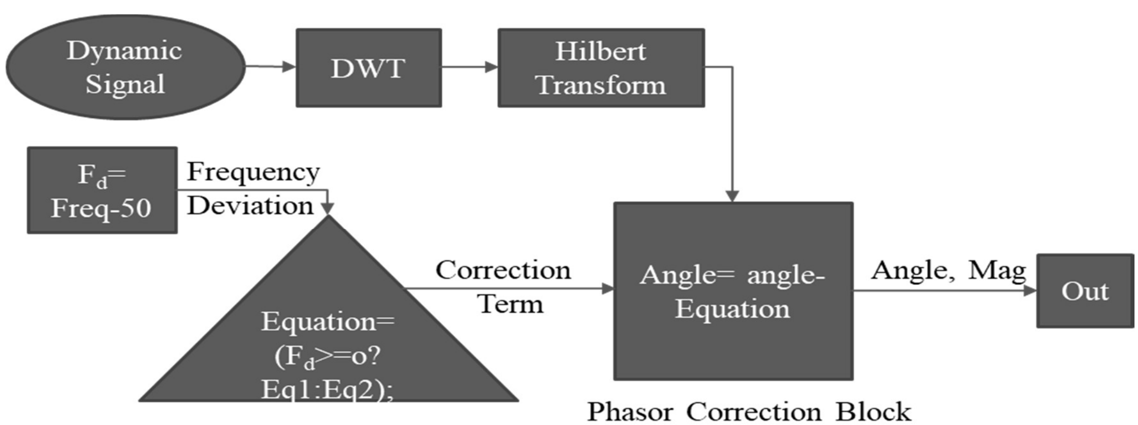


Fig 3.1.b Final algorithm

3.4 Expression of Dynamic Signal as input:

The dynamic signal used for simulation was V

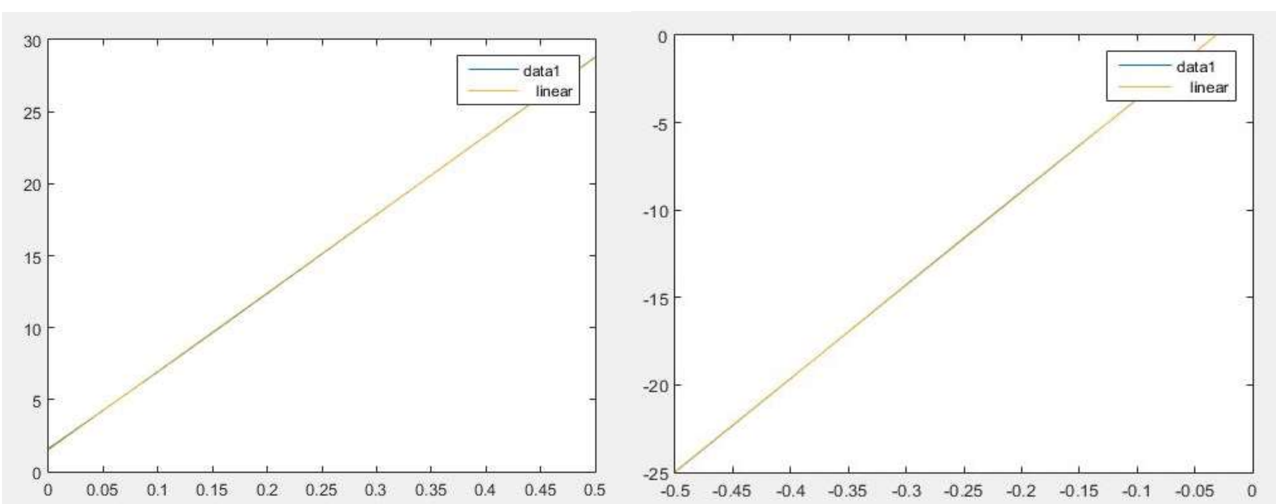
$$V = vrms * \sin(2\pi * f * t + \varphi) + 0.8 * \sin(2\pi * 3f * t - 70°) + 0.4 * \sin(2\pi * 5f * t) + 0.24 * \sin(2\pi * 9f * t) + 0.16 * \sin(2\pi * 11f * t + 20°) + \text{noise (SNR=30)}$$

Where φ is angle of input signal's fundamental component.

For this signal on varying frequency from 49.5 Hz to 50.5 Hz or in other words varying f_d from -0.5 Hz to 0.5 Hz for $\varphi = 0°$, the measured φ is as in Table 2.

f_d	-0.5	-0.45	-0.4	-0.35	-0.3	-0.25	-0.2	-0.15	-0.1	-0.05
φ	-24.97	-22.30	-19.63	-16.96	-14.28	-11.60	-8.9436	-6.2988	-3.6659	-1.04
f_d	0.05	0.1	0.15	0.2	0.25	0.3	0.35	0.4	0.45	0.5
φ	4.2607	6.9441	9.6525	12.3799	15.1171	17.8553	20.5893	23.3197	26.0509	28.7896

Table 2. Angle measured in case of frequency deviation



For f_d >0 For f_d <0

Fig 3.2 Curve fitting for phase correction

On fitting the data of Table 1 we are able to calculate φ approximately equal to 0° or 360° in case of linear curve fitting. The equations obtained for curve fitting were as follows

For 49.5<f<50; Eq1= 54.467*f_d+ 1.525

For 50<f<50.5; Eq2= 53.248*f_d+ 1.674

3.5 Results after phase angle correction:

After correcting the phase due to deviation in frequency, the results for different phase angles between 0° to 360° are as in Table 3.

Angle Input	Magnitude	Frequency (in HZ)										
		49.5	49.6	49.7	49.8	49.9	50	50.1	50.2	50.3	50.4	50.5
0	3.98789363	359.97741	359.9915	360.0239	360.032	359.985	0.0776032	-0.02776	-0.03865	-0.01009	0.00757	0.03069
35	4.01096683	34.639492	34.70903	34.80681	34.8849	34.911	35.075621	35.0271	35.0442	35.0729	35.0739	35.0583
50	4.01887125	49.483696	49.58832	49.71799	49.826	49.8833	50.074782	50.0354	50.0375	50.0361	50.0043	49.9519
80	4.01954437	79.316576	79.4765	79.63953	79.7713	79.8587	80.073514	80.0176	79.9549	79.8711	79.7693	79.6553
100	4.00820467	99.365058	99.52633	99.67024	99.7806	99.8611	100.07302	99.9943	99.886	99.7575	99.6298	99.5044
160	3.97306255	159.82449	159.806	159.7733	159.769	159.828	160.07395	160.021	159.92	159.825	159.789	159.796
200	3.99074432	199.70558	199.6483	199.6063	199.629	199.75	200.07603	200.106	200.09	200.077	200.104	200.144
240	4.01528762	239.43479	239.4546	239.4922	239.584	239.745	240.07823	240.106	240.092	240.063	240.028	239.976
300	3.99579194	299.73487	299.8073	299.8705	299.919	299.937	300.07955	299.96	299.871	299.793	299.699	299.602
320	3.98512457	319.93645	319.9783	320.0148	320.025	319.989	320.07923	319.937	319.857	319.805	319.742	319.688
340	3.98184406	340.03171	340.0478	340.0697	340.064	340.004	340.07856	339.943	339.894	340.865	340.496	339.514
360	3.98789185	359.97739	359.9915	360.0238	360.032	359.985	0.0775925	-0.02777	-0.03863	-0.0101	0.00757	0.03071

Table 3. Simulation results of the dynamic signal after phase correction

3.6 Power system model used in Simulink:

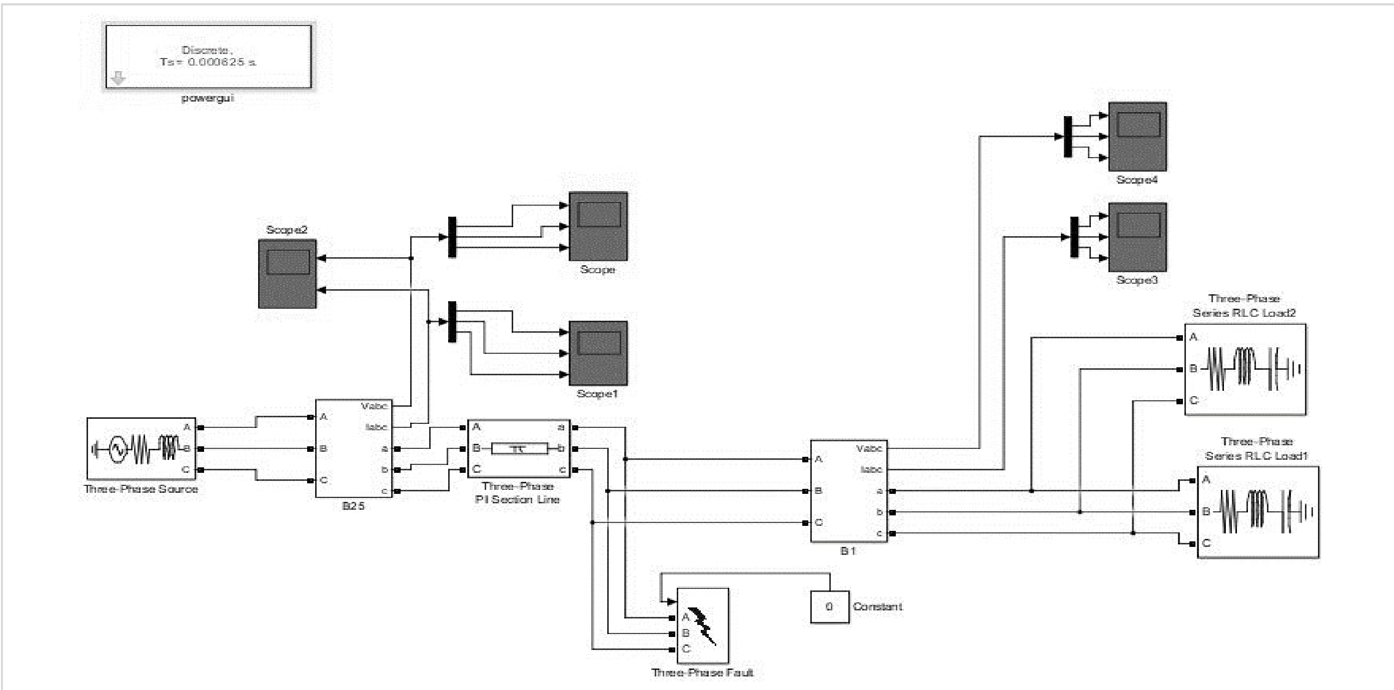


Fig 3.3 Power System Simulink Model

In the Simulink model, Source and load are Y connected and the specifications of the components are:

Voltage at three phase source = 35.35V (rms value)

Impedance Z_s of the source = $(1/7 + i)*10^{-8}\,\Omega$

Impedance Z_{tl} of the transmission line = $0.6365 + i\ 14.66\ \Omega$

Nominal phase to phase voltage (Load) = 1 KV

Power drawn by the load = P (varying)

Load type = Resistive

3.7 Simulation Results of power system model:

Table 4. Measured in Simulink

Power (KW)	Angle of Load Current	Mag of Load Current	Angle of Load Voltage	Mag of Load Voltage
5	-2.6089	0.1025	-2.6015	20.32
10	-6.7328	0.205	-6.7291	20.095
15	-10.7512	0.2975	-10.7463	19.7775
20	-14.6221	0.3875	-14.6203	19.38
22	-16.1252	0.4225	-16.1235	19.2
25	-18.3217	0.4725	-18.3248	18.915
50	-33.9171	0.805	-33.9164	16.1225
100	-52.5637	1.1225	-52.5633	11.25

Table 5. Manually calculated

Power (KW)c	Angle of Load Current	Mag of Load Current	Angle of Load Voltage	Mag of Load Voltage
5	-4.179	0.10145	-4.179	20.29
10	-8.288	0.2007	-8.288	20.07
15	-12.2882	0.29625	-12.2882	19.75
20	-16.147	0.387	-16.147	19.35
22	-17.644	0.421975	-17.644	19.17875
25	-19.837	0.47225	-19.837	18.89
50	-35.39	0.80625	-35.39	16.125
100	-54.0374	1.126375	-54.0374	11.26375

4. Conclusion and Future Scope

This report emphasizes on the algorithm for designing micro phasor measurement unit (micro PMU). The proposed algorithm is verified using synthetic signals generated in MATLAB.

From the results we see that it is capable of calculating phasors accurately even in case of presence of noise, frequency harmonics and also when frequency is deviated from its nominal value. And we are able to calculate phasor correctly of a power system model in Simulink also.

In future, we can now implement the algorithm on hardware and develop a prototype. This can be placed in distribution network to analyze and monitor the power lines in real time.

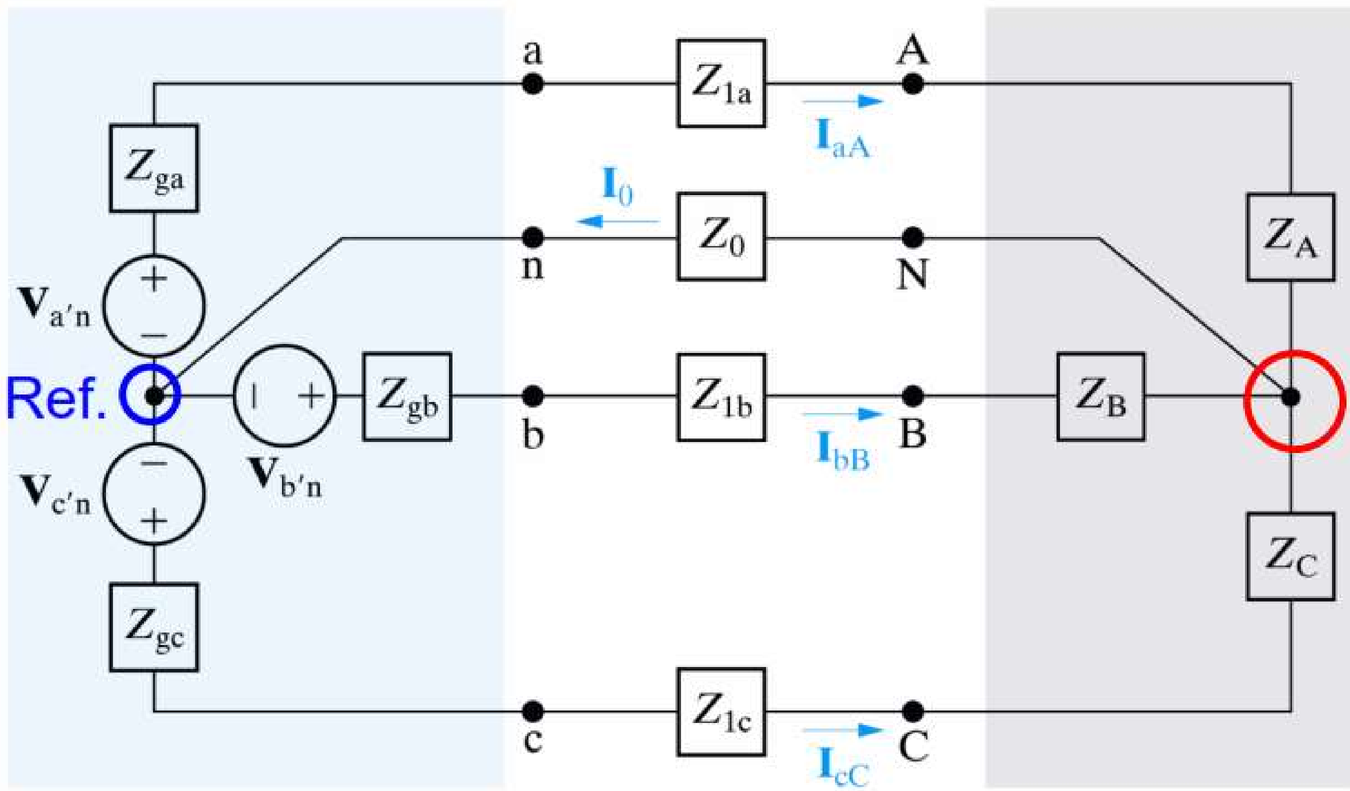
The Micro PMU can be used for Diagnostic Applications such as unintentional island detection, reverse power flow detection, state estimation, fault location, oscillation detection and control applications such as protective relaying, volt-VAR optimization, micro grid coordination etc.

Appendix 1

Analysis of Y-Y circuit:

In case of balanced system, three phase system can be realized using 1 phase system as following:

Three phase model:



For Y-connected load, line current equals phase current.

From above figure: $I_0 = I_{aA} + I_{bB} + I_{cC}$

$$\frac{V_N}{Z_0} = \frac{V_{a'n} - V_N}{Z_{ga} + Z_{1a} + Z_A} + \frac{V_{b'n} - V_N}{Z_{gb} + Z_{1b} + Z_B} + \frac{V_{c'n} - V_N}{Z_{gc} + Z_{1c} + Z_C} \qquad \text{……. Eq (a)}$$

For balanced three phase circuits, $\{V_{a'n}, V_{b'n}, V_{c'n}\}$ have equal magnitude and 120° relative phases.

Since $Z_{ga} = Z_{gb} = Z_{gc}$ and $Z_{1a} = Z_{1b} = Z_{1c}$ and $Z_A = Z_B = Z_C$, Total impedance Z_L along any line is same

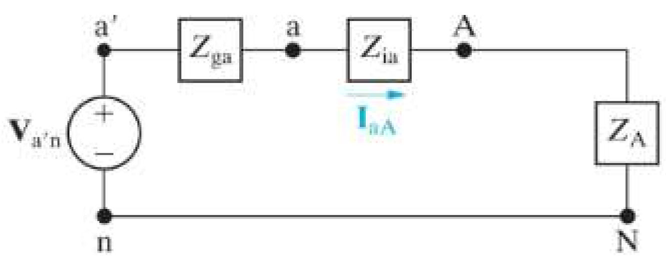
$Z_L = Z_{ga} + Z_{1a} + Z_A$

Eq (a) becomes $\frac{V_N}{Z_0} = \frac{V_{a'n}-V_N}{Z_L} + \frac{V_{b'n}-V_N}{Z_L} + \frac{V_{c'n}-V_N}{Z_L} \Longrightarrow$ $V_N = 0$

$V_N = 0$ means no voltage difference between nodes n and N in the presence of $Z_0 \Longrightarrow$ Neutral line is both short (v=0) and open (i=0).

The three-phase circuit can be separated into 3 one-phase circuits (open), while each of them has a short between nodes n and N.

Phase A:



Load voltage V_{AN}, Load current I_{aA} can be calculated as:

$I_{aA} = \frac{V_{a'n}}{Z_{ga}+Z_{ia}+Z_A}$ and $V_{AN} = I_{aA} * Z_A$

For constant Z type load in three phase load in Simulink; Z_A (resistive in our case) can be calculated as:

$Z_A = V^2/P$; V is nominal phase to phase voltage of load and P is the power drawn by the load.

Appendix 2

Sampling frequency of the signal is fs taken as 1600 Hz here. And the signal frequency is f1= 50 Hz. We are observing signal for tf = 1 second. And the snr (signal to noise ratio is taken as 30), freq is varied from 49.5 Hz to 50.5 Hz to study the effect of frequency deviation, p1 and p2 represents the coefficients of the linear curve fitted for freq > 50 Hz and for freq < 50 Hz respectively.

```
fs=1600; ts=1/fs; f1=50; tf=1;f2=49.75;c=1;snr=30; n=21; freq = linspace(49.5,50.5,n);

p1= [54.467481642254924, 1.525092936177868];

p2= [53.248763638928680, 1.6740066113463];
```

Generating voltage signal with odd harmonics and having noise and signal's frequency is different in different 10 time intervals which is done for dynamic signal analysis.

```
for i1=1:n

f3= freq(i1); k=0;

for t=0:ts:tf-ts;

if t<.1

    v=5*sin(2*pi*f1*t+pi/180*0)+1*sin(2*pi*3*f1*t-pi/180*70) +0.5*sin(2*pi*5*f1*t)+
       0.3*sin(2*pi*9*f1*t)+0.2*sin(2*pi*11*f1*t+pi/180*20);

    v=awgn(v,snr);

 elseif t<.2

    v=4*sin(2*pi*f3*t+pi/180*360)+.8*sin(2*pi*3*f3*t-  pi/180*70)+ 0.4*sin(2*pi*5*f3*t)+
       0.24*sin(2*pi*9*f3*t)+0.16*sin(2*pi*11*f3*t+pi/180*20);

    v=awgn(v,snr);

  elseif t<.3

    v=5*sin(2*pi*f3*t)+1*sin(2*pi*3*f3*t-pi/180*70)+0.5*sin(2*pi*5*f3*t)+ 0.3*sin(2*pi*9*f3*t)+
       0.2*sin(2*pi*11*f3*t+pi/180*20);
```

```
v=awgn(v,snr);

 elseif t<.4

    v=6*sin(2*pi*f1*t)+1.2*sin(2*pi*3*f1*t-pi/180*70)+0.6*sin(2*pi*5*f1*t)+0.36*sin(2*pi*9*f1*t)+
       0.24*sin(2*pi*11*f1*t+pi/180*20);

    v=awgn(v,snr);

elseif t<.5

    v=4*sin(2*pi*f3*t)+.8*sin(2*pi*3*f3*t-pi/180*70)+0.4*sin(2*pi*5*f3*t)+0.24*sin(2*pi*9*f3*t)+
       0.16*sin(2*pi*11*f3*t+pi/180*20);

    v=awgn(v,snr);

elseif t<.6

    v=5*sin(2*pi*f1*t)+1*sin(2*pi*3*f1*t-pi/180*70)+0.5*sin(2*pi*5*f1*t)+0.3*sin(2*pi*9*f1*t)+
        0.2*sin(2*pi*11*f1*t+pi/180*20);

    v=awgn(v,snr);

elseif t<.7

    v=6*sin(2*pi*f2*t)+1.2*sin(2*pi*3*f2*t-pi/180*70)+0.6*sin(2*pi*5*f2*t)+0.36*sin(2*pi*9*f2*t)+
        0.24*sin(2*pi*11*f2*t+pi/180*20);

    v=awgn(v,snr);

elseif t<.8

    v=4*sin(2*pi*f1*t)+.8*sin(2*pi*3*f1*t-pi/180*70)+0.4*sin(2*pi*5*f1*t)+0.24*sin(2*pi*9*f1*t)+
       0.16*sin(2*pi*11*f1*t+pi/180*20);

    v=awgn(v,snr);

elseif t<.9
```

```
    v=5*sin(2*pi*f1*t)+1*sin(2*pi*3*f1*t-pi/180*70)+0.5*sin(2*pi*5*f1*t)+ 0.3*sin(2*pi*9*f1*t)+
      0.2*sin(2*pi*11*f1*t+pi/180*20);

    v=awgn(v,snr);

  else

    v=4*sin(2*pi*f2*t)+.8*sin(2*pi*3*f2*t-pi/180*70)+0.4*sin(2*pi*5*f2*t)+0.24*sin(2*pi*9*f2*t)+
      0.16*sin(2*pi*11*f2*t+pi/180*20);

    v=awgn(v,snr);

  end

   k=k+1;

   sig(k)=v;

   xi(k)=i;

end
```

Analyzing the signal using **dynamic_analysis** function for 2nd time interval (160 samples in 1st time interval and 161 to 320 samples in 2nd time interval). **dynamic_analysis** function has been defined later.

```
[angle_out,rms_out]= dynamic_analysis (sig,50,161,320);

ferr(i1)=f3-f1; ang_2(i1)= sum(angle_out)/length(angle_out);  // observing average angle
```

Phase coorection in case of frequency deviation

```
if ferr(i1)>=0

angl3(i1)=ang_2(i1)-(p1(1)*ferr(i1)+p1(2));

end
```

```
if ferr(i1)<0
  angl3(i1)=ang_2(i1)-(p2(1)*ferr(i1)+p2(2));
end
end
```

dynamic_analysis function to calculate phase and magnitude of the signal

dwt_start and dwt_end are used to find the actual time in the time interval in which we are analyzing the signal and H0 and H1 are LPF and HPF coefficients used for calculating DWT of the signal (Fig 2.1)

```
function [angle_out,rms_out]= dynamic_analysis(sig,freq,sig_start,sig_end)
sig2=sig(sig_start:sig_end); dwt_start=ceil(sig_start/8); dwt_end=ceil(sig_end/8);
 H0= [-2.99883648961829e-10,4.05612705555044e-09,-1.81484324829912e-08,2.01432202353783e-
    10,2.63392422626911e-07,-6.84707959699876e-07,-1.01199401001899e-06,7.24124828766967e-06,-
    4.37614386218458e-06,-3.71058618339259e-05,6.77428082838443e-05,0.000101532889737123,-
    0.000385104748697454,-5.34975984353524e-05,0.00139255961933283,-0.000831562172806687,-
    0.00358149425960073,0.00442054238700206,0.00672162730208939,-0.0138105261375186,-
    0.00878932492447835,0.0322942995301110,0.00587468181135674,-0.0617228996246833,
    0.00563224685775138,0.102291719175138,-0.0247168273378751,-0.155458750706696,
    0.0398502464580176,0.228291050819922,-0.0167270883090561,-0.326786800433922,-
    0.139212088011442,0.361502298739192,0.610493238938368,0.472696185310729,0.219942113551317,
    0.0634237804590585,0.0105493946249466,0.000779953613666564];
 H1= [-0.000779953613666564,0.0105493946249466,-0.0634237804590585,0.219942113551317,-
    0.472696185310729,0.610493238938368,-0.361502298739192,-0.139212088011442,
    0.326786800433922,-0.0167270883090561,-0.228291050819922, 0.0398502464580176,
    0.155458750706696,-0.0247168273378751,-0.102291719175138, 0.00563224685775138,
```

0.0617228996246833,0.00587468181135674,-0.0322942995301110,-0.00878932492447835, 0.0138105261375186,0.00672162730208939,-0.00442054238700206,-0.00358149425960073, 0.000831562172806687,0.00139255961933283,5.34975984353524e-05,-0.000385104748697454,-0.000101532889737123,6.77428082838443e-05,3.71058618339259e-05,-4.37614386218458e-06,-7.24124828766967e-06,-1.01199401001899e-06,6.84707959699876e-07,2.63392422626911e-07,-2.01432202353783e-10,-1.81484324829912e-08,-4.05612705555044e-09,-2.99883648961829e-10];

Calculating DWT of the signal up to 3 stages and set the DWT mode to periodization otherwise we will not get exactly half signal coefficients after each stage.

```
dwtmode('per') ;

[t1 t2]= dwt(sig2, H0,H1); [t3 t4]= dwt(t1, H0,H1); [t5 t6]= dwt(t3, H0,H1);  t7=t5/2.828427;
```

Calculating rms value and phase of the signal

```
rms_out=sqrt(sum(t7.^2)*2/length(t7));        t8=hilbert(t7);

for i= dwt_start:dwt_end

    ang(i-dwt_start+1)= angle(t8(i-dwt_start+1))-2*pi*freq*(i-1)/200;        // φ = angle − 2π ∗ 50 ∗ t

end

angl1 = (ang*180/pi)+180; x=angl1;                              // converting phase angle from radian to degree

for i=1:length(x)          k=1;

while(k>0)

   if (x(i)>=0 && x(i)<= 360)       k=1;

   else      k=0;

   end

   if(k)    x1(i)=x(i); k=0;

   else     x1(i)= x(i)+360;  k=1; x(i)=x1(i);
```

```
end

end

end

   angle_out= x1;

end
```

Appendix 3

Time Complexity of proposed algorithm using wavelet:

In the algorithm, I used Db20 filter coefficients $\Rightarrow$ no of filter coefficients N = 40

No of samples to be realized at 1^{st} stage = 160

No of samples to be realized at 2^{nd} stage = 80

No of samples to be realized at 3^{rd} stage = 40

The computational complexity of addition at each stage is given by N-1 per samples

The computational complexity of multiplication at each stage is given by N per sample

$\Rightarrow$ total computational complexity of multiplication = 40*160 + 40*80 + 40*40 = 11200

and total computational complexity of addition = 39*160 + 39*80 + 39*40 = 10920

Time Complexity of Fast Fourier transform (fft):

The computational complexity of multiplication = $N\log_2 N$

The computational complexity of addition = $2N\log_2 N$

Where N is number of samples

⇨ total computational complexity of multiplication = $160*\log_2 160 = 1171.51$

and total computational complexity of addition = $2*160*\log_2 160 = 2343.02$

We see that the complexity of wavelet transform is much more than the complexity of fft.

References

1. A. G. Phadke, "Synchronized Phasor Measurement in Power Systems," IEEE Computer Applications in Power, April 1993,
2. Development of a Micro- Phasor Measurement Unit for Distribution System Applications, *Hari Prasanna Das and Ashok Kumr Pradhan, IIT kharagpur.*
3. Phasor Measurement Units and Wide Area Monitoring Systems, *Antonello Monti, Carlo,Musca Ferdinanda Ponci.*
4. The Framework and Algorithm of a New Phasor Measurement Unit, *Jim Li; Xiaorong Xie, Member, IEEE; Jmyu Xiao; Jmgtao Wu.*
5. The Design of Approximate Hilbert Transform Pairs of Wavelet Bases, Ivan W. Selesnick*, Member, IEEE.*
6. Hardware Implementation of Polyphase Decomposition Based Wavelet Filters for Power System Harmonics Estimation**,** *Vinay K. Tiwari, Student Member, IEEE, and Sachin K. Jain, Senior Member, IEEE.*
7. Real- Time Power System Frequency and Phasors Estimation, Jinfeng Ren*, Student Member, IEEE*, and Mladen Kezunovic*, Fellow, IEEE.*
8. Automated Anomaly Detection in Distribution Grids Using _PMU Measurements, *Mahdi Jamei, Anna Scaglione, Ciaran Roberts, Emma Stewart.*
9. I*nternational Conference on Emerging Trends in Electrical, Electronics and Sustainable Energy Systems (ICETEESES–16) 978-1-5090-2118-5/16/$31.00 ©2016 IEEE*

   Synchrophasor Measurement Network and its Applications in Indian Grid, *R.B. Sharma1 and G.M. Dhole2.*
10. Low Voltage Micro-Phasor Measurement Unit (μPMU), *Bogdan Pinte, Michael Quinlan, and Karl Reinhard.*