EXPERIMENTAL INVESTIGATION OF ADAPTIVE SAMPLING PERIOD BASED PLL

A DISSERTATION

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

of

MASTER OF TECHNOLOGY

in

ELECTRICAL ENGINEERING

(With Specialization in Instrumentation and Signal Processing)

By

VINEET VAJPAYEE

DEPARTMENT OF ELECTRICAL ENGINEERING
INDIAN INSTITUTE OF TECHNOLOGY ROORKEE
ROORKEE-247 667 (INDIA)

JUNE, 2013

No.– MT/256/P.S/2013

CANDIDATE'S DECLARATION

I hereby declare that this thesis report entitled **EXPERIMENTAL INVESTIGATION OF ADAPTIVE SAMPLING PERIOD BASED PLL**, submitted to the Department of Electrical Engineering, Indian Institute of Technology, Roorkee, India, in partial fulfillment of the requirements for the award of the Degree of Master of Technology in Electrical Engineering with specialization in Instrumentation and Signal Processing is an authentic record of the work carried out by me during the period from September 2012 to June 2013 under the supervision of **Dr. P SUMATHI, Department of Electrical Engineering, Indian Institute of Technology, Roorkee.** The matter presented in this thesis report has not been submitted by me for the award of any other degree of this institute or any other institute.

Date: 14-06-13
Place: Roorkee

Vineet
Vineet Vajpayee

CERTIFICATE

This is to certify that the above statement made by the candidate is true to the best of my knowledge and belief.

14/June/2013
Dr. P SUMATHI
Assistant Professor
Department of Electrical Engineering
Indian Institute of Technology Roorkee

"I am the master of my fate: I am the captain of my soul."

William E Henley

ABSTRACT

The proposed phase locking scheme is based on the principle of adaptive sampling period adjustment. It variably changes its sampling period to generate the output signal in synchronism with the incoming input signal in terms of phase and frequency. The phase error between input signal and the look up table generated signal is calculated by a multiplier. The phase locked loop acts to minimize this phase error by generating sampling pulses which are having variable pulse width.

The purpose of PLL is very basic and its application area varies from power system to communication system. This report covers grid synchronization and FM demodulation as two main applications. The scheme can be used on any frequency range and it is tested on 50 Hz and 10 kHz. The proposed scheme is also tested in various fault situation. The important parameters like lock-in range, pull-in range, and hold-in range are calculated for proposed scheme. The simulation results shows that the proposed scheme has wide lock-in range and it exhibits quick acquisition of input signal. The simulation and hardware implementation of the proposed scheme are done on FPGA.

The objectives of this work include:

1. Study of different types of phase locked loops.
2. To propose a novel phase locked loop.
3. Simulate the adaptive sampling period based PLL under different conditions.
4. Implement the proposed scheme on FPGA.
5. To apply the scheme in FM demodulation and grid synchronization.

Acknowledgements

I would like to express my deep sense of gratitude and sincere thanks to my beloved guide **Dr. P Sumathi**, Department of Electrical Engineering, Indian Institute of Technology Roorkee, for being helpful and a great source of inspiration. Her keen interest and constant encouragement gave me the confidence to complete my work. I wish to extend my sincere thanks for her excellent guidance and suggestions for the successful completion of my work. Despite of her busy schedule, she used to review my dissertation progress, give her valuable suggestions and made corrections. Her dedication to work and self belief will always inspire me, and I hope to continue to work with her noble thoughts.

I gratefully acknowledged Dr. R. P. Maheshwari, Dr. Vinod Kumar, and Dr. R. S. Anand for providing necessary research facilities to carry out this work and valuable suggestions and motivations provided by other faculty members of the department are duly acknowledged.

A special thanks to my fiance Divya Devaliya for her love, care and moral support.

I would like to extend my gratitude to my friends and class mates for their support throughout the past two years. My special acknowledgments go to Archit, Belvin, Sunil, Sarita, Ankit, Ankur, Nitin and Vivek.

I owe a lot to my family who unremittingly supported me during my years of study. I would like to pay high regards to my mother and father for being a true motivator through out my life. Their relentless backing made this work possible and I would like to dedicate it to them.

Above all, I owe it all to Almighty God for granting me health, wisdom, and strength.

Vineet Vajpayee

Contents

List of Figures

List of Tables

Abbreviations

ADPLL	**All Digital Phase Locked Loop**
APF	**All Pass Filter**
APLL	**Analog Phase Locked Loop**
DCO	**Digital Controlled Oscillator**
DFT	**Discrete Fourier Transform**
DPGS	**Distributed Power Generation System**
DPLL	**Digital Phase Locked Loop**
EPLL	**Enhanced Phase Locked Loop**
FIR	**Finite Impulse Response**
FPGA	**Field Programmable Gate Array**
HDL	**Hardware Discriptive Language**
HTB-PLL	**Hilbert Transform based Phase Locked Loop**
IC	**Integrated Circuit**
JTAG	**Joint Test Action Group**
LUT	**Look Up Table**
NCO	**Numerically Controlled Oscillator**
PD	**Phase Detector**
PI	**Proportional Integral**
PLL	**Phase Locked Loop**
park-PLL	**Park transform based Phase Locked Loop**
pPLL	**Power Phase Locked Loop**
QAM	**Quadrature Amplitude Modulation**
QPLL	**Quadrature Phase Locked Loop**
QPSk	**Quadrature Phase Shift Keying**

SC-PLL	**S**ynthesis **C**ircuit based **P**hase **L**ocked **L**oop
SGT	**S**liding **G**oertzel **T**ransform
SOGI-PLL	**S**econd **O**rder **G**eneralized **I**ntegrator **P**hase **L**ocked **L**oop
SRF-PLL	**S**ynchronous **R**eference **F**rame based **P**hase **L**ocked **L**oop
TCL	**T**ool **C**ommand **L**anguage
TDB-PLL	**T**ransport **D**elay based **P**hase **L**ocked **L**oop
VCO	**V**oltage **C**ontrolled **O**scillator
VHDL	**V**ery **H**igh Speed **I**ntegrated **C**ircuit **H**ardware **D**iscriptive **L**anguage
VSPF-PLL	**V**ariable **S**ampling **P**eriod **F**ilter based **P**hase **L**ocked **L**oop
XOR	**E**xclusive **OR**

To my grandmother Shanti Devi Devaliya

Chapter 1

Introduction

Before a decade the main sources of energy were non-renewable sources like fossil fuel, coal, petroleum, natural gas but due to the limited resource availability, day-to-day growing cost and the environmental problems associated with their use it forced people to shift their focus to renewable sources. Presently to satisfy the growing demand of energy there is very much need of alternative and renewable sources of energy. The renewable sources of energy are becoming popular because of their clean energy generation, environmental friendly behavior, low cost production, less maintenance requirement, reliable nature and energy independent nature [1]. The renewable sources that are very much popular are solar energy, photo-voltaic, hydro-power, and wind power. This shift in trend from non-renewable sources towards renewable sources has changed the scenario of utility networks. The increase in capacity of the distributed power generation system (DPGS) has introduced problems like power outages, grid instability, and connected systems instability. Therefore there is need of highly accurate controlled strategies to deal with the problems of synchronization.

The power converters, AC-DC converters, cyclo-converters, and voltage compensator these are all part of DPGS. These all have to be connected to the grid and they have to be in synchronism with the utility grid. The information provided by the phase angle of voltage vector is also very important. This information can be used for the purpose of derivation of output signal, to develop control mechanism, to generate feedback variables,

to measure current content, to find out harmonics, and to calculate power flow [2]. There are different strategies to find out the phase information from voltage vector. The technique that is used now a days and very popular is Phase Locked Loop (PLL).

The presence of harmonics degrades the system performance and creates problems like line dips and line notches. The situation becomes worst in variable frequency environment. In variable frequency environment, frequency varying continuously with respect to time, step change in frequency, sudden step variation in phase angle and the presence of decaying DC component are very much likely happening situations. The phase locked loop must be able to synchronize in the presence of all these undesired condition and this synchronization must be very fast [3].

The PLLs are widely used in frequency demodulation. The PLL is a coherent way of communication and one thing to be noted here that coherent demodulators are better than non-coherent demodulators in terms of noise performance [4]. The PLL extracts the message signal from sinusoidal frequency modulated signal. The phase-locked loop (PLL) is attractive as an FM demodulator because of its threshold extension capabilities compared to the conventional FM discriminator [5].

This dissertation report employs adaptive sampling period based PLL for grid synchronization and FM demodulation as two main aspects. To extract fundamental signal it becomes imperative to use phase-locking scheme. The PLL is not only limited to grid synchronization, it is having wide area of operation. It is used in motor control, automatic controllers, frequency/phase demodulation, frequency/phase estimation, tracking filters, frequency dividers, frequency synthesis, clock recovery, and carrier recovery circuits like Costas loop, squaring loop and many more [6].

This dissertation report is divided into six different chapters. Chapter 2 gives a brief overview of different PLL techniques used in the literature. Chapter 3 aims towards explaining the basics of PLL, describing mathematical equations, block diagrams, and important parameters. Chapter 4 explains the proposed scheme with detailed description of each block. Chapter 5 shows the simulated as well as implemented results, the discussion on each and every result is also presented. The future scope of the scheme and applications are discussed in chapter 6. Finally conclusions are drawn in chapter 7.

Chapter 2

Literature Survey

"Dont read success stories, you will only get a message.
Read failure stories, you will get some ideas to get success."

-Dr A. P. J. Abdul Kalam

2.1 Background

The introduction to the concept of phase locked loop (PLL) was made in 1932 by de Bellescize. Before the introduction of PLL, super-heterodyne receivers were used up to 1920 for synchronous reception of radio signals. Another way to simplify the approach for signal reception was to use synchronous homo-dyne receiver. During 1940 the use of PLL becomes very popular with television receivers. The main purpose was to synchronize the sweep oscillators like horizontal and vertical oscillators with the sync pulses. The complexity associated with the use of PLL in discrete component poses hindrance in their popularity. The introduction of PLL integrated circuits (ICs) in 1960 suddenly changed the situation. The low cost provided by monolithic PLL ICs has initiated new field of applications [7]. As the IC technology starts growing the new advancement in terms of increase in speed, reliability, and performance have made digital PLL (DPLL) widely useful. The analog PLL (APLL) has the problems of component saturation, initial calibration, periodic recalibration, and sensitivity to DC drifts. These problems have been removed by digital PLL and it made real time signal processing more accurate [8]. In the field of control in 1970s the use of PLL was limited to speed control of synchronous motor. Since then the advent of IC technology and microprocessors have increase their

application to digital controllers with advanced features [9]. The PLLs also have wide use in modern communication system and in consumer's electronic products. In short, main applications of PLL include data recovery, clock recovery, demodulation techniques, tracking filters, frequency synthesis, phase estimation, and clock synchronization [10].

2.2 Versions of PLL

This section discusses different single-phase PLL techniques. The different PLLs can be categorized on the basis of phase detector. The phase detector can be a simple multiplier or it can be a combination of different multiplier, integrator, gain, and adder. Another category of phase detector is depending upon the generation of input signal. One of the most used PLL topology is synchronous reference frame based PLL (SRF-PLL). The work mentioned in [11] describes three single phase digital PLLs. The first one is single phase power-based PLL (pPLL). As the name suggests it is based on fictitious electrical power. The phase detector is a simple multiplier circuit. The behavior of pPLL can be easily explained by electric power analogy. The input voltage signal is multiplied with the fictitious current signal to produce mean power. The produced fictitious power mean is made zero and that implies the current signal will be in quadrature to voltage signal. The low pass filter extracts the mean power. The phase detector dynamics totally depends on filter design. The output of phase detector generates second harmonics and to discard the harmonics the low pass filter must have low cut off frequency but this will degrade system response. This drawback can be removed by increasing filter order and at the same time attenuating the second harmonics. Another thing that it produces subharmonic components of very low frequency near fundamental. To overcome, this requires slight attenuation near fundamental frequency. This also requires careful designing of filter and compensator.

The second single-phase PLL is inverse park transformed based PLL (parkPLL). This is based on the concept of synchronous reference frame PLL (SRF-PLL). V_α is the input voltage signal and another input is V_β, latter is generated internally after doing inverse park transformation. These two inputs are applied to the park transformation block, which converts $\alpha\beta$ to dq. There are two first order filters used which have τ_d and τ_q as their respective time constants. As a consequence there is a trade off between rejection of harmonics and speed of response. The presence of any DC shift in input will introduces oscillation in dq component. The parkPLL has a feature of inherent filtering.

An enhanced PLL (EPLL) discussed in [12] is based on the principle of adaptive filtering. The output of EPLL is able to lock in terms of phase as well as amplitude. Instead of directly multiplying the output of voltage controlled oscillator (VCO) with the input signal, an intermediary signal is generated. This intermediary signal is derived from VCO output. Then this intermediary signal is multiplied with the VCO output to generate phase error. The phase detector circuitry consists of three multiplier, one subtractor, one integrator, and one $\pi/2$ phase shifter. It has an advantage that direct estimation of phase and amplitude of input signal is possible. Another advantage is that the system is robust with respect to noise and structure. A small variation in internal parameters can be easily tolerated without affecting the performance.

The quadrature PLL (QPLL) discussed in [13] is based on gradient-descent algorithm and employs the concept of estimating the in-phase and quadrature-phase component. The QPLL is an enhanced version of EPLL and Costas loop. The QPLL estimates the frequency component directly while amplitude and phase estimation needs calculation. The phase detector of QPLL directly estimates quadrature-phase signal and time derivative of phase angle of input signal so it can be mainly used in application related to quadrature signal generation like quadrature amplitude modulation (QAM), quadrature phase shift keying (QPSK). This PLL offers wide pull-in and lock-in range and fast convergence. The phase detector consists of six multipliers, three adders, four gains, and two integrators which overall makes it a complex circuitry and increase computational complexity.

Another way to classify PLLs is on the basis of quadrature signal generation. The Transport delay based PLL (TDB-PLL), Hilbert transform based PLL (HTB-PLL) and second order generalized integrator PLL (SOGI-PLL) falls in the same category. The TDB-PLL uses a transport delay to generate a 90 degree phase-shifted input signal. Here all the harmonic content of input signal is undergone with the same delay. It is implemented by the use of first-in-first-out buffer. The size of the buffer is made equal to one fourth the number of samples contained in one cycle of fundamental component [14].

The HTB-PLL [15] uses Hilbert transform to generate the quadrature signal. Only the way of generation of quadrature signal is different as compared to previous method, the rest of the structure is same. The Hilbert transform is non causal in nature so it can not be implemented practically. It can be realized with the help of finite impulse response (FIR) filter. In case of low order FIR filter the PLL output is highly distorted. The higher

order filter can solve this problem for off-line application but in on-line applications it would not be a good solution. The harmonic components can be estimated in transient as well as in steady state. In terms of computational complexity the HTB-PLL has higher complexity and TDB-PLL has one by fourth improvement on the HTB-PLL.

There are some limitations with previous quadrature signal generation methods like frequency dependency, associated nonlinearity, poor filtering, and complexity. The method discussed in [16] has a simple structure and generates already filtered signal without any delay. The approach is based on second order generalized integrator PLL (SOGI-PLL). This system can be tuned adaptively by adjusting the resonance frequency. An all-pass filter (APF) discussed in [17] utilizes the same approach as used in SOGI-PLL but it does not attenuate higher order terms. Both of these methods are able to generate accurate orthogonal signal but in case of variable frequency environment an additional circuit is required to make sure that the generated signal is 90 degree phase shift of the input signal.

The method described in [18] employs a synthesis circuit (SC) to generate the orthogonal component. This method ensures 90 degree phase shift in variable frequency environment. The previous discussed methods generated an orthogonal signal by shifting input signal. Here the generation is done by detecting amplitude and phase from PLL output. The synthesis circuit consists of a multiplier, a low pass filter, and a sine block. This PLL fails under the presence of harmonics. One way to improve SC-PLL performance is to add one more SC-PLL for each harmonics to eliminate, but it increases the computational effort.

Another way to achieve synchronization is on the basis of variable sampling period approach. The work shown in [19] represents the same principle and proposes two compensation schemes using discrete Fourier transform (DFT) for power converter. The approach is based on correcting the sampling period. The sampling period of digital filter is adjusted in such a way to match the time window with the grid period. But this method is not suitable for single micro-controllers application. The other approach is based on addition of phase offset to cancel out the phase error produced by DFT. This method is less efficient than previous since it does not ensure rejection of harmonic components and attenuation of sub-harmonic component. There is also presence of synchronization jitter when DFT sampling period is asynchronous with the grid period.

The method presented in [20] is based on the same principle of variable sampling period in which the rectangular window filter adjusts the sampling period. The designed code of this method is simple and mathematical approach is used to show different dynamics. The hardware results are also taken in rigorous condition and which shows robustness of the proposed algorithm. The limitation with this system is that the model is not optimized and bandwidth got reduced due to the presence of controller.

The research work shown in [21] is also based on the principle of variable sampling period and it employs Sliding Goertzel transform (SGT) based filter. The authors have also mentioned in the research work using the same approach for three phase system in [22]. They compared the results of different PLL schemes. The compared PLL results clearly show the best result given by variable sampling period filter based PLL (VSPF-PLL). The SGT filter places zero in the multiples of grid frequency and this will remove the entire harmonic component. But placing zero in the multiples of grid frequency reduces the bandwidth. The SGT filter increases complexity of the system.

Chapter 3

Basics of PLL

"Live as if you were to die tomorrow.
Learn as if you were to live forever."

-Mahatma Gandhi

The PLL is a device or a mechanism by which output signal remains in synchronization with the input signal with respect to phase and frequency. In PLL one signal is used to track or follow the other signal. This is done by reducing the phase error between the output signal and the reference or input signal. The objective is to minimize the phase error between input and output signal. The basic components of a PLL system are Phase Detector (PD), a Voltage Controlled Oscillator (VCO) and a Loop Filter.

3.1 Basics Governing Equation for PLL

Let $V_i(t)$ be the applied input voltage and $V_o(t)$ be the produced output voltage.

$$V_i(t) = ASin(\omega_i t + \theta_i) \tag{3.1}$$

$$V_o(t) = Cos(\omega_o t + \theta_o) \tag{3.2}$$

where ω_i, ω_o are angular frequencies of the input and VCO output respectively and θ_i, θ_o are phases of input and VCO output respectively. If the loop is initially unlocked and the PD has sinusoidal characteristic then the error signal $V_e(t)$ is given by

$$V_e(t) = AK_d Sin(\omega_i t + \theta_i) Cos(\omega_o t + \theta_o)$$

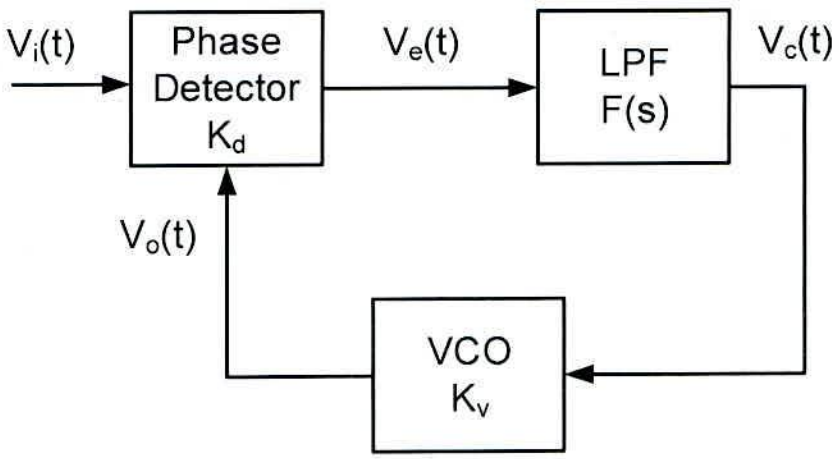

FIGURE 3.1: Block diagram of a classical PLL.

$$V_e(t) = \frac{AK_d}{2}[Sin((\omega_i + \omega_o)t + \theta_i + \theta_o) + Sin((\omega_i - \omega_o)t + \theta_i - \theta_o)] \tag{3.3}$$

where K_d is the gain of phase detector. Now the error signal is passed through a low pass filter. This low pass filter removes the higher frequency component. From equation 3.3 it is clear that the low pass filter attenuates the first component and pass only the second component. The output of low pass filter is given by $V_c(t)$

$$V_c(t) = \frac{AK_d}{2}[Sin((\omega_i - \omega_o)t + \theta_i - \theta_o)] \tag{3.4}$$

After a period the VCO output becomes synchronous with the input so we can assume $\omega_i \approx \omega_o$ and $\theta_i - \theta_o = \theta_d$. Then output of filter will be

$$V_c(t) = \frac{AK_d}{2}Sin(\theta_d) \tag{3.5}$$

It is clear from above equations that the VCO is acting as an integrator. One way to linearize this PLL is that for a slowly varying signal we can assume $Sin(\theta_d) \approx \theta_d$. This assumption becomes invalid when θ_d is large.

3.2 Linear Analysis of Classical PLL

A linearized PLL model can be drawn from . The widely used linearized model of PLL is shown in Figure 3.4. There are different important parameter that can be calculated on

the basis of linear model. One of the basic model about a system is its transfer function model.

3.2.1 Transfer function

The transfer function between the input signal and the VCO output signal $G_o(s)$, can be obtained as

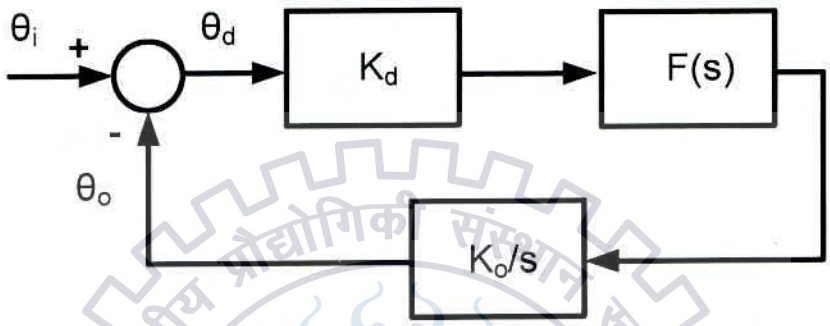

FIGURE 3.2: Block diagram representing linear analysis of PLL.

$$G_o(s) = \frac{\theta_o(s)}{\theta_i(s)} = \frac{K_d K_v F(s)}{s + K_d K_v F(s)} \tag{3.6}$$

The transfer function between the input signal and the phase error detector output signal $G_d(s)$, can be obtained as

$$G_d(s) = \frac{\theta_d(s)}{\theta_i(s)} = \frac{s}{s + K_d K_v F(s)} \tag{3.7}$$

The order of PLL can be obtained from equation 3.6 and the stability can be analyzed using Bode plot and Nyquist plot.

3.2.2 Hold Range

The Hold range is defined as the frequency range in which PLL is able to statically maintain phase tracking. If PLL is in locked state and the reference signal is varied then it will go out of synchronism at the edges of the hold range. The Hold range is obtained

by calculating frequency offset at the input which is responsible for forcing phase error to go beyond linear range [23].

$$\Delta\omega_H = K_d K_o F(0) \tag{3.8}$$

3.2.3 Lock-in Range

The Lock-in range is defined as the range in which the PLL locks within one single beat note between input and output frequency [9]. If relatively numerator and denominator are of first order then system is going to behave like a first order system and lock range can be calculated as

$$\Delta\omega_L = \pm K_d K_o F(\infty) \tag{3.9}$$

3.2.4 Pull-in Range

The Pull-in range describes the PLL in an acquisition or dynamic mode and the PLL will always be locked in this range. If frequency is applied outside the pull-in range it would not be able to get locked.

3.2.5 Pull-out Range

The Pull-out range describes the PLL in static state and is the dynamic limit for stable operation. The PLL is initially in locked state and then a frequency step is applied which causes the PLL to get unlocked. If a frequency step of value less than pull-out range is applied then the PLL remains locked and if step is of more value as compared to pull-out range then the PLL fell out of the lock [23].

3.2.6 Steady-State Error

As we know from the final value theorem

$$\lim_{t\to\infty} \theta_d(t) = \lim_{s\to 0} s\theta_d(s) \tag{3.10}$$

$$\lim_{t\to\infty} \theta_d(t) = \lim_{t\to\infty} s\theta_i(s)G_d(s) \tag{3.11}$$

3.3 Nonlinear Analysis of PLL

The first method for analyzing PLL in nonlinear way is Phase Plane method. It is a graphical method of phase plane design. This method is suitable only for first and second order PLL. The second method is Lyapunov Method. In this method we try to find out the Lyapunov Energy function and then carry out the analysis. There are some more methods like Circle and Popov Criteria. These two methods are used mainly for the stability analysis of higher order PLL [9].

3.4 Digital PLL

The PLL is a basic component in implementation of digital coherent communication without Doppler shift effect and in its proper reception.The problems that we have in analog PLL (APLL) are like sensitive to DC drifts, component saturation, how to increase the order, need of initial calibration and periodic adjustment. The basic component of an APLL are balanced mixer, Low pass RC-filter and a variable tuned capacitor. In the earliest work on DPLL a sample and hold circuit is used and after that digital equipment like flip flop for phase detector implementation, digital filter and digital VCO.

Digital PLL may consist of digital phase detector while rest of the circuitry is analog. If all the component are digital in nature like a digital phase detector, a digital filter, and a numerically controlled oscillator then we classify it as an all digital PLL (ADPLL). An all software PLL implements everything in computer code and these are entirely digitalized [23].

3.4.1 Classical DPLL

The Classical DPLL exploit only the digital phase detector while filter and VCO are analog in nature. The output produced by this phase detector will be a continuous time voltage. This PLL have the advantage that it can be implemented at very high frequencies. The disadvantages associated with classical DPLL are similar to those with APLL.

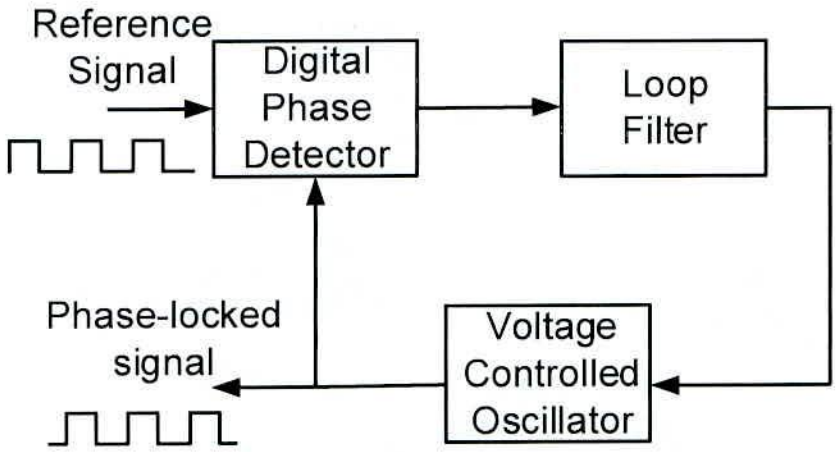

FIGURE 3.3: Block diagram of digital phase locked loop.

3.4.2 All Digital PLL

The main difference between classical DPLL and ADPLL is in the digital phase detector. In classical DPLL the purpose of phase detector was to produce a continuous time voltage. The phase detector output in ADPLL is digital one and it generates either pulses or multi bit values. The ADPLL exploits digital filter and digital controlled oscillator (DCO). There are different variant of ADPLL. In a way, a digital phase detector which produces pulses and these pulses are according to either low or high error and the counter serves the purpose of filter. The generated pulses will be continuous so the counter can be driven by a clock and finally this combination produces a sample data. Another way is that when digital phase detector produces multi bit values then a digital filter can be employed [23].

3.4.3 Software PLL

If the data is sampled at a very faster rate then the PLL operation can be implemented in software. This method offers flexibility and it can be used for any PLL provided that the sample rate chosen is high enough. One thing to be noted here that the software PLL deals with real time data so there can be problems related to sampling. The software PLL also finds application in post processing of data [10].

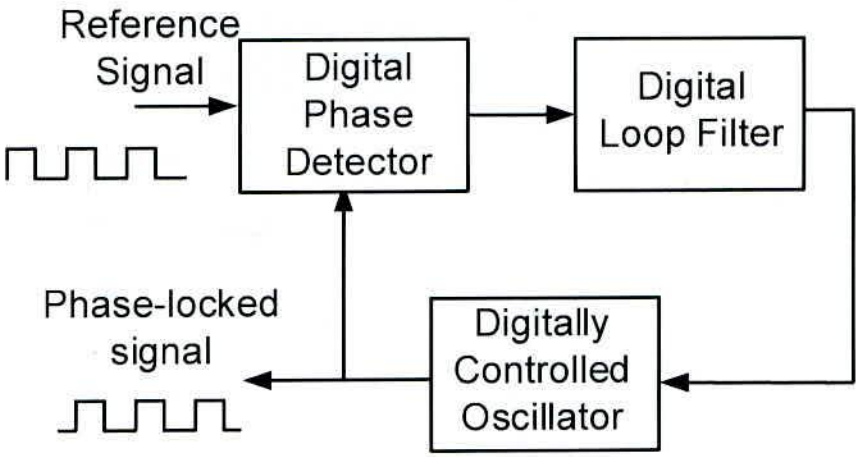

FIGURE 3.4: Block diagram all digital phase locked loop.

3.5 Phase Detector

Phase detector can be designed either by analog component or digital component. In analog domain a sinusoidal phase detector and square signal phase detector are available. The sinusoidal phase detector works as a multiplier and it is not having any memory. The phase detection interval is from $-\pi/2$ to $\pi/2$. The square signal generator is also called as digital phase detector. The implementation of square signal phase detector is done by sequential circuits. It contains memory. They operate with binary input and therefore built up from digital circuits. For triangular phase detector the range of linear operation is from $-\pi/2$ to $\pi/2$. For sawtooth phase detector the range is from $-\pi$ to π, for sequential phase or frequency detector it varies from -2π to 2π. The following section deals with them in detail.

3.5.1 Multiplier

Multiplier is a mixer phase detector which is of nonlinear in nature and has very simple structure. This phase detector multiplies the incoming input signal with the VCO output to produce phase error. This produces output by considering the full magnitude of input and VCO output so it has superior performance as compared to other phase detector which works on quantizing the input. The problem with this is that the loop gain will change as amplitude changes [10].

3.5.2 XOR based detector

If amplitudes of input signals to the phase detector is very large then the amplifier will saturate and the shape of output signal will be like rectangular pulses. For such type of output Exclusive-OR (XOR) circuit can be applied. This XOR phase detector has loop gain which is now independent of amplitude of input signals. It has an advantage that it has large linear range as compared to multiplier. The disadvantage is that the linearity of baseband input signal will become a function of duty cycles of incoming inputs [10].

3.5.3 Phase-Frequency detector

One of the widely used phase detector is the addition of tri-state phase frequency detector with charge pump. The purpose of charge pump is to remove loading effect produced by circuit and it also help to smoothen the response. This method can be very advantageous in some situation and it can give wrong output in some situation [10].

3.6 Loop Filter

The presence of a loop filter makes a PLL at least type II. The loop filter is basically an integrator and its digital equivalent will be an accumulator. As VCO is acting as an integrator and another integration action is done by loop filter so there will be two integrators. To ensure stability in such type of system the loop filter structure will always be minimum phase. As analog loop filter, a simple RC filter can be used to serve the purpose of low pass filtering. It can be designed actively or passively. Now analog filters have been replaced by their digital equivalent. One thing to be noted here that the digital filters are very much depend upon the design of phase detector and DCO. The order of analog and digital filter can be increased by increasing number of integrators and accumulators respectively. One different type of filter is sequential filter. This filter produces output only when a certain threshold is crossed by input. It does not produce output after each observation although it keeps on observing input and when it cross a certain threshold level then only it produces output. An example of such type of filter is N-before-M filter. The Lead and the lag inputs are accumulated in the N and the M counters separately providing $N < M < 2N$. Assume that initially all three counters are reset. The random sequence of binary inputs continues until one of two conditions is met. (i) If one of the N counters in fills up before or simultaneously with the M counter,

then the corresponding output is produced, all three counters are reset, and the cycle begins anew. (ii) If a total of M lead + lag inputs occur before N inputs of either type (lead or lag), all registers are reset and no output is produced. The latter condition is most likely when the phase error is near zero [8].

3.7 Voltage Controlled Oscillator

The voltage controlled oscillator (VCO) can be categorized into two category as the classical VCO and digital controlled oscillator (DCO). There is one more type of oscillator called as numerically controlled oscillator (NCO) [24].

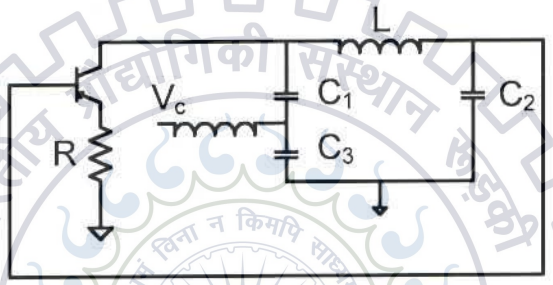

FIGURE 3.5: Block diagram of a voltage controlled oscillator.

The basic nature of a VCO is like an integrator and by changing the voltage on a variable capacitor the output frequency of operation can be varied. The VCO will provide stability in phase, high modulation sensitivity, large frequency deviation, and it will also be required for wide band modulation. The voltage controlled crystal oscillators have higher stability as compared to resonator oscillators, RC multi-vibrators and YIG tuned oscillators. The YIG tuned oscillators are used at microwave frequency. For wider operating range an LC oscillator will serve the purpose. One more requirement from VCO is phase stability and that can be achieved using high Q value, stabilizing temperature and maintaining low noise[8].

One of the simplest way of making a VCO is by connecting odd number of inverters in a feedback loop. A Schmitt-trigger is used in case of relaxation oscillator with a resonant circuits in positive feedback. A DCO is a digital equivalent of VCO and it acts as a divide by N counter and is shown in Figure 3.6 on page 17. This counter is driven by a stable oscillator which produces frequency M times the intermediate frequency. Then a

comparator is fed by a reference input and output of counter. The DCO output will be produced after comparison. The period of DCO is controlled by input N [23].

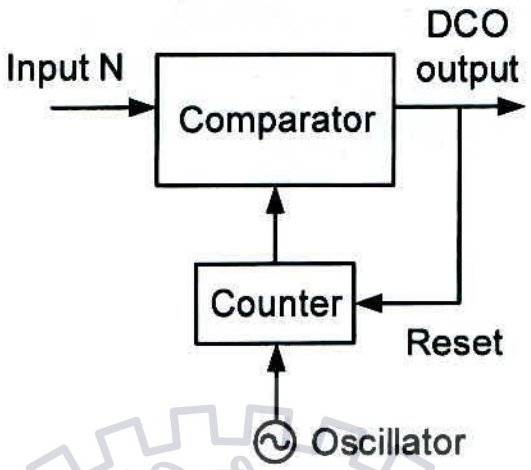

FIGURE 3.6: Block diagram of a digitally controlled oscillator.

Chapter 4

Proposed Scheme

"If we knew what it was we were doing,
it would not be called research, would it?"

-Albert Einstein

The proposed scheme is based on the principle of adaptive sampling period adjustment. The same principle has been used in [19–22]. The block diagram of the proposed phase locking scheme is shown in Figure 4.1 on page 19. The phase detector employs a single multiplier, which produces the error signal between the incoming signal and the look up table (LUT) generated reference signal. The error signal is fed to the moving average filter to discard the unwanted frequency components. The filtered error signal is applied to the PI controller for further smoothening and control action. The PI controller maximizes the bandwidth of the PLL with reducing the steady state error. The output of the PI controller is varied with the respect to the error signal, which is further applied to the numerically controlled oscillator. The NCO outputs the sampling pulses which are used to determine the phase of the reference signal. These sampling pulses are fed to the N-counter to obtain the phase $\phi(n)$. The NCO adaptively adjusts the sampling pulses required for generating the reference signal. The N-counter is a reference generator produces sawtooth waveform which describes the phase of the locally generated reference signal. The output of the N-counter, i.e. instantaneous value of phase is applied to the look up table (LUT) block and then the generated sine wave is fed to the phase error detector to complete the loop.

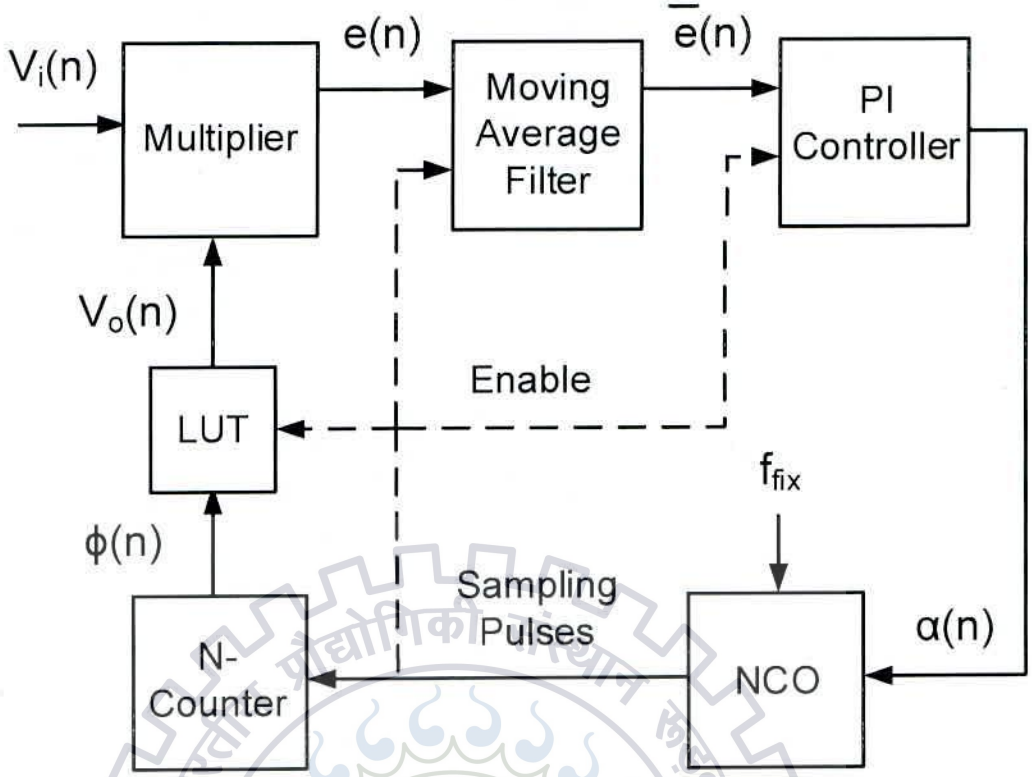

FIGURE 4.1: Block diagram representation of adaptive sampling frequency based PLL.

4.1 Phase Error Detector

The simplicity of the proposed scheme lies in the structure of the phase error detector. The multiplier phase detector, which is non-linear nature, correlates the input signal with the locally generated reference signal. The fundamental input voltage signal is a cosine vector and is contaminated by harmonics. The input signal $V_i(n)$can be represented as

$$V_i(n) = \hat{V}Cos(\theta_i(n)) + \sum_{k=2}^{N} \hat{V}_k Cos(k\theta_i(n) + \theta_k) \tag{4.1}$$

where the first term represents the fundamental input voltage vector and second term represents the sum of harmonics. $\hat{V}$ and $\theta_i(n)$ denotes the peak amplitude and phase of fundamental input signal respectively. $\hat{V}_k$ and θ_k denote peak amplitude and phase of k^{th} harmonics respectively. N is the length of window of the moving average filter. The output of phase error detector $e(n)$ can be represented as

$$e(n) = V_i(n)Sin(\phi(n)) \tag{4.2}$$

Substituting equation 4.1 in 4.2

$$\begin{aligned} e(n) = \frac{\hat{V}}{2}Sin[\phi(n) - \theta_i(n)] + \frac{\hat{V}}{2}Sin[\phi(n) + \theta_i(n)] \\ + \sum_{k=2}^{N}\left[\frac{\hat{V}_k}{2}Sin[\phi(n) + k\theta_i(n) + \theta_k]\right] \\ + \sum_{k=2}^{N}\left[\frac{\hat{V}_k}{2}Sin[\phi(n) - k\theta_i(n) - \theta_k]\right] \end{aligned} \tag{4.3}$$

Under ideal condition the phase of the input signal and that of LUT generated reference signal becomes equal, $\phi(n) = \theta_i(n)$. Hence equation 4.3 becomes

$$\begin{aligned} e(n) = \frac{\hat{V}}{2}Sin[2\theta_i(n)] + \sum_{k=2}^{N}\left[\frac{\hat{V}_k}{2}Sin[(1+k)\theta_i(n) + \theta_k]\right] \\ + \sum_{k=2}^{N}\left[\frac{\hat{V}_k}{2}Sin[(1-k)\theta_i(n) - \theta_k]\right] \end{aligned} \tag{4.4}$$

From equation 4.4, it could be observed that oscillations are present in error signal. The *e(n)* does not represent the real difference between the phase of line voltage and that of reference signal. The oscillating signal *e(n)* is 90^0 shifted version of input signal which has frequency component of second order and has half the magnitude than that of the fundamental component. There are two more oscillating signals generated by the harmonic component which has frequencies of 1-k and 1+k times than that of the input signal.

4.2 Moving average Filter

The transfer function of moving average filter in the z-domain is represented by

$$G_{MA}(Z) = \left(\frac{1}{N}\right)\left(\frac{1 - r^N Z^{-N}}{1 - rZ^{-1}}\right) \tag{4.5}$$

The moving average filter consists of N zeros equally spaced around Z-domain's unit circle and a single pole at unit circle. The pole-zero plot of the filter in Z-domain is shown in Figure 4.3 on page 22. The plot is drawn for N=128. The 128 zeros are positioned all around the unit circle and one pole is located at Z=1. It has pole-zero cancellation at $Z = e^{\pm j2\pi k/N}$ for k=0 [25]. The block diagram of moving average filter is shown in Figure 4.2 on page 21. 'r' is a damping factor introduced to guarantee the stability of system and its value is set close to one. This filter is also called as a Recursive

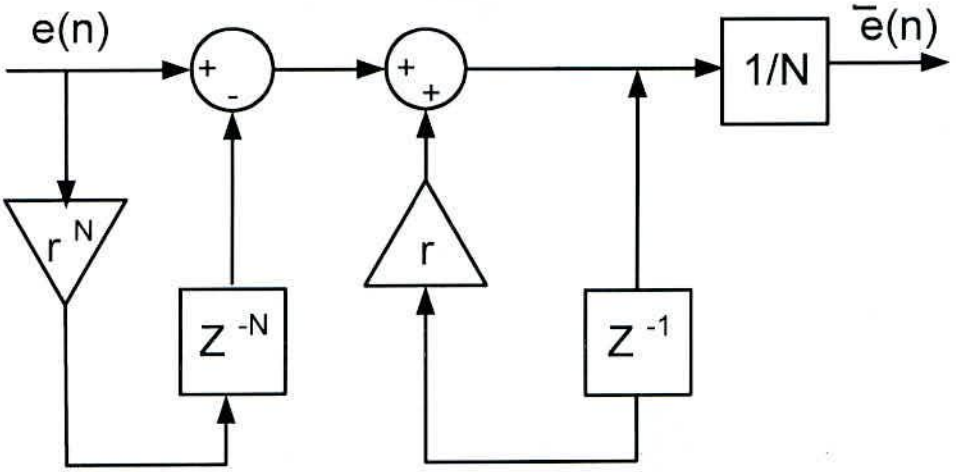

FIGURE 4.2: Block diagram of moving average filter.

Running Sum or Boxcar Averager. It performs one addition and one subtraction per output sample and it is independent of value of N. It works very well to reduce white noise and at the same time preserves step response. It is the fastest digital filter available [26]. The moving average filter easily extracts the DC value present in the error signal. It blocks completely the fundamental frequency and harmonics. The value of N is set at 128 and that of r is at 0.9997. The magnitude response of the moving average filter is plotted in Figure 4.4 on page 22.

4.3 PI Controller

The purpose of PI controller is to maximize bandwidth and it is also responsible for the quick locking. The filtered phase error signal is applied to its input and it generates a steady DC. The transfer function in Z-domain is derived by backward difference approach and is given by

$$G_{PI}(Z) = K_p + K_i T_S \left(\frac{1}{1 - Z^{-1}} \right) \tag{4.6}$$

where K_p, K_i and T_S are proportional gain and integral gain and sampling period respectively. A limiter is introduced to take care of $\alpha(n)$ which should be between $-1 \leq \alpha(n) \leq 1$.

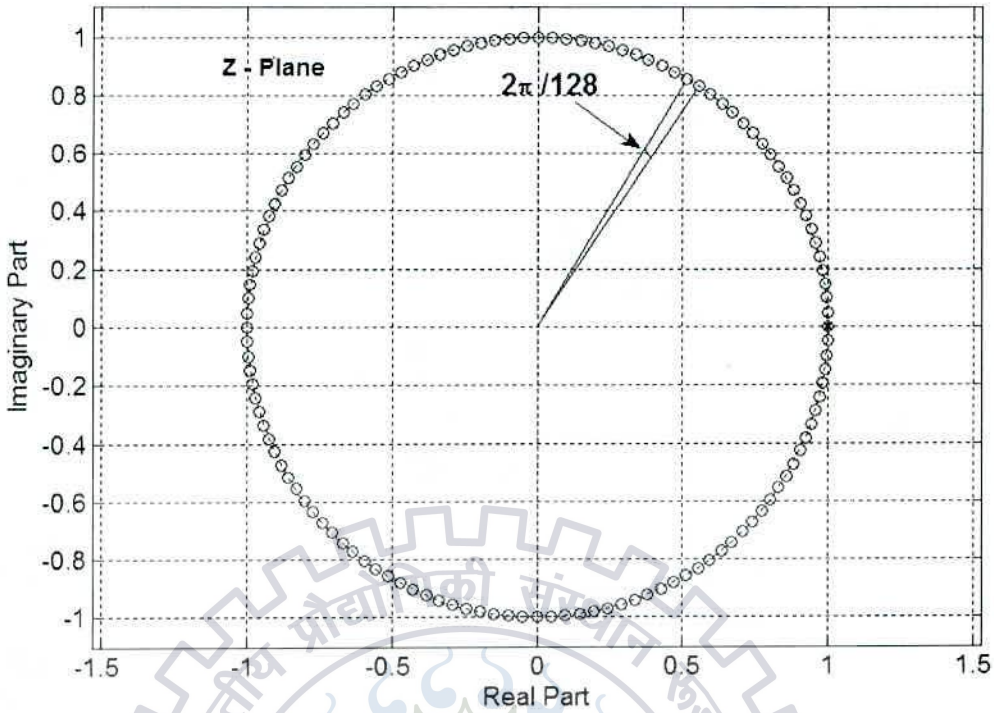

FIGURE 4.3: Pole-Zero plot of moving average filter.

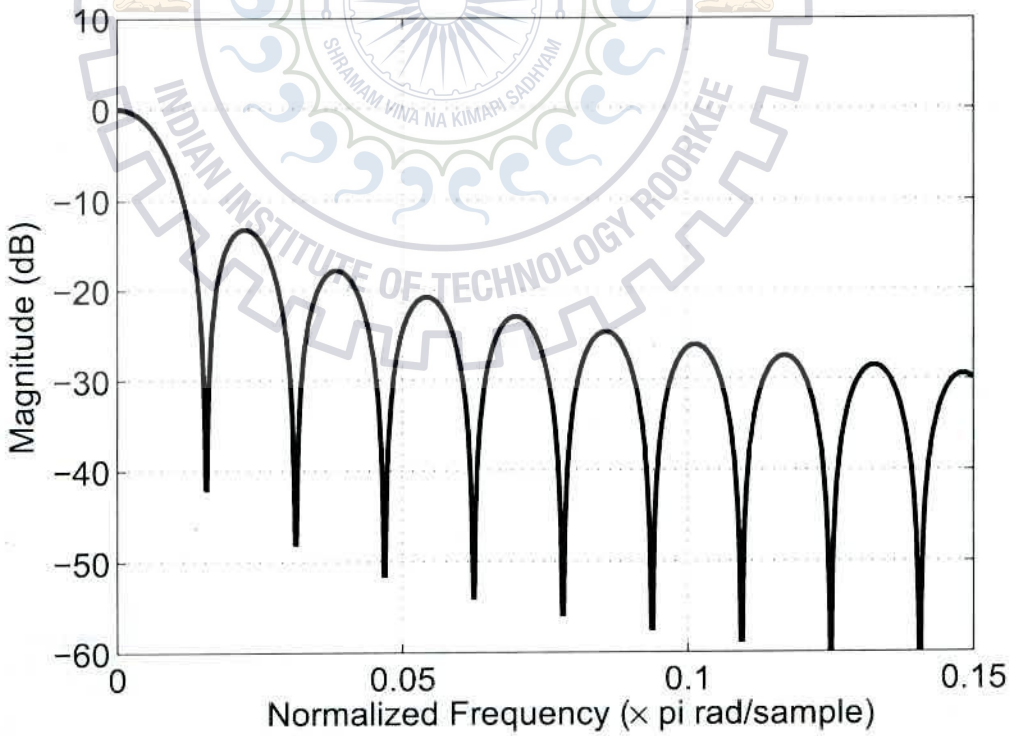

FIGURE 4.4: Pole-Zero plot of moving average filter.

4.4 Numerically Controlled Oscillator

The numerically controlled [24] oscillator produces sampling pulses which are further applied to the reference generator for the generation of reference phase signal. The generated pulses have different pulse width and it depends upon the controlled signal $\alpha(n)$ which is acting as input to the NCO [27]. The application of the NCO provides the wide operating range of the PLL, which distinguishes the proposed PLL from the existing PLLs. The block diagram of numerically controlled oscillator is shown in Figure 4.5 on page 24. The state equation obeyed by NCO could be expressed as

$$\begin{bmatrix} X_1(n+1) \\ X_2(n+1) \end{bmatrix} = \begin{bmatrix} \alpha & \alpha - 1 \\ \alpha + 1 & \alpha \end{bmatrix} \begin{bmatrix} X_1(n) \\ X_2(n) \end{bmatrix} \tag{4.7}$$

$$X_1(0) = 1; X_2(0) = 0$$

The value of α is given by

$$\alpha = Cos(\zeta) \tag{4.8}$$

and as $Cos(\zeta)$ lies between -1 to 1 so the condition $-1 \leq \alpha(n) \leq 1$ is self explanatory.

The value of ζ is given by

$$\zeta = \frac{2 * \pi * f_S}{f_{fix}} \tag{4.9}$$

where f_S and f_{fix} are the sampling frequency and NCO enabling frequency respectively. The value of f_{fix} is set to be $4 \times f_S$. The internal structure of NCO divides the enabling frequency by four so that the output frequency will be f_S and it is used for enabling LUT, moving averager, and PI controller. The controlled signal α behavior can be analyzed from the equation 4.9. If we use the above stated condition $f_{fix} = 4 \times f_S$ then equation 4.9 will results $\zeta = \frac{\pi}{2}$ and $Cos\left(\frac{\pi}{2}\right) = 0$ i.e. $\alpha = 0$. This is an important point because as $\alpha = 0$ so it will extract the fundamental signal.

Now consider if frequency of input signal is increasing so f_S is also going to be increasing and the value of $Cos(\zeta)$ will be more than $\frac{\pi}{2}$. If $\frac{\pi}{2} < Cos(\zeta) < \pi$ then it will be negative and α will also be negative. Similarly if frequency of input signal is decreasing so $0 < Cos(\zeta) < \frac{\pi}{2}$, α will be positive. The proposed scheme is easily able to synchronize with the input signal under its variation from DC to second harmonics [28].

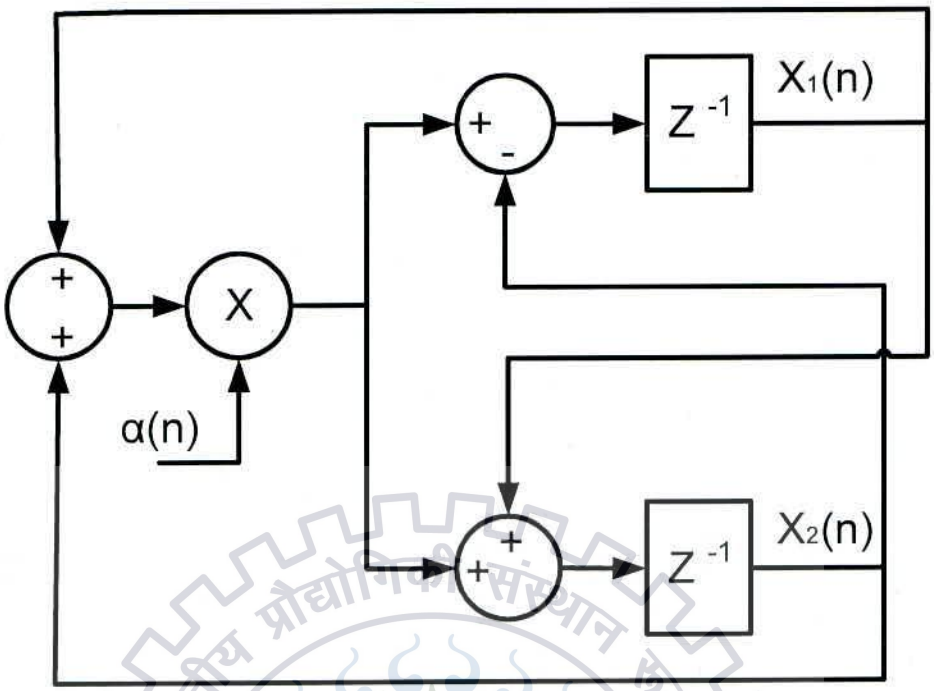

FIGURE 4.5: Block diagram of numerically controlled oscillator.

4.5 Reference Signal Generator

The reference generator is an up counter which increments its value by $2\pi/N$ for each sampling pulse. The maximum value up to which it can count depends upon N. Here the value of N is chosen to be 128. It starts with 0 and keeps on counting up to 2π with each period incrementing its value by $2\pi/N$. The block diagram representation is shown in Figure 4.6 on page 25. The unipolar sampling pulses are acting as an input to reference generator and then it produces sawtooth wave showing the phase of the reference signal.

4.6 Look Up Table

The Sine look up table gives the corresponding sine value indicated by the index given by counter. The increased magnitude of LUT improves quick acquisition behavior and lock-in time. The sampling pulses drives the counter which acts as an input to the LUT.

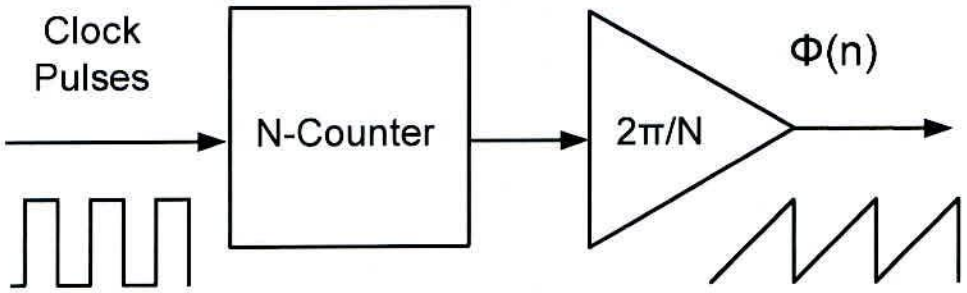

FIGURE 4.6: Block diagram of reference generator.

The LUT is enabled by the sampling pulses coming out of the NCO. The LUT also helps in reducing steady state error as it acts as a PI controller [29].

Chapter 5

Results and Discussion

"Yesterday is gone. Tomorrow has not yet come. We have only today. Let us begin."

-Mother Teresa

This chapter is divided into three sections, the first section 5.1 presents all the simulation results while the second section 5.2 gives an overview about DSP Builder and shows simulation results and finally the third section 5.3 presents all the results obtained in hardware on the field programmable gate array (FPGA).

5.1 Simulation Results

The proposed scheme is implemented and simulated in MATLAB® Simulink environment. The scheme is also tested in different conditions. The applied input to the PLL is of sine nature and it is represented as

$$V_i = Sin(2 * \pi * 50 * n) \tag{5.1}$$

This input is applied to the phase detector with the reference signal. The phase detector is a multiplier and it produces multiplication of the input signal with the output signal. The input is sampled at a rate of 25.6 kHz and it can be found out very easily. The 50 Hz input is multiplied by 128 and then it is multiplied by 4 because of the presence of NCO. Then the sampling frequency comes out to be 50*4*128 = 25.6 kHz. The value

of N is set to 128. The values of PI controller are tuned in such a way to maximize bandwidth and to track the input signal as soon as possible.

5.1.1 Analysis of Fundamental Signal

The Figure 5.1 on page 27 is showing the tracking of 50 Hz signal in ideal environment i.e. in absence of harmonics. The values of P and I are chosen to 1.36 and 2.16 respectively. The output signal takes less than one cycle or 0.0135 s to completely track the input signal. The controlled signal variation is shown in Figure 5.2 on page 28. The variation of controlled signal makes the situation clearer and understandable. The initial overshoot shows that the output signal takes time to follow the input signal and after that the controlled signal reaches the steady state. The steady state error can be reduced further at the cost of initial lock-in time. There is trade-off between fast lock-in time and steady state error. This can be achieved by changing the values of P and I of the PI controller. The phase angle variation is shown in Figure 5.3 on page 28, it is the output of counter which is counting from 0 to 2π with each time it is incrementing by $2\pi/128$.

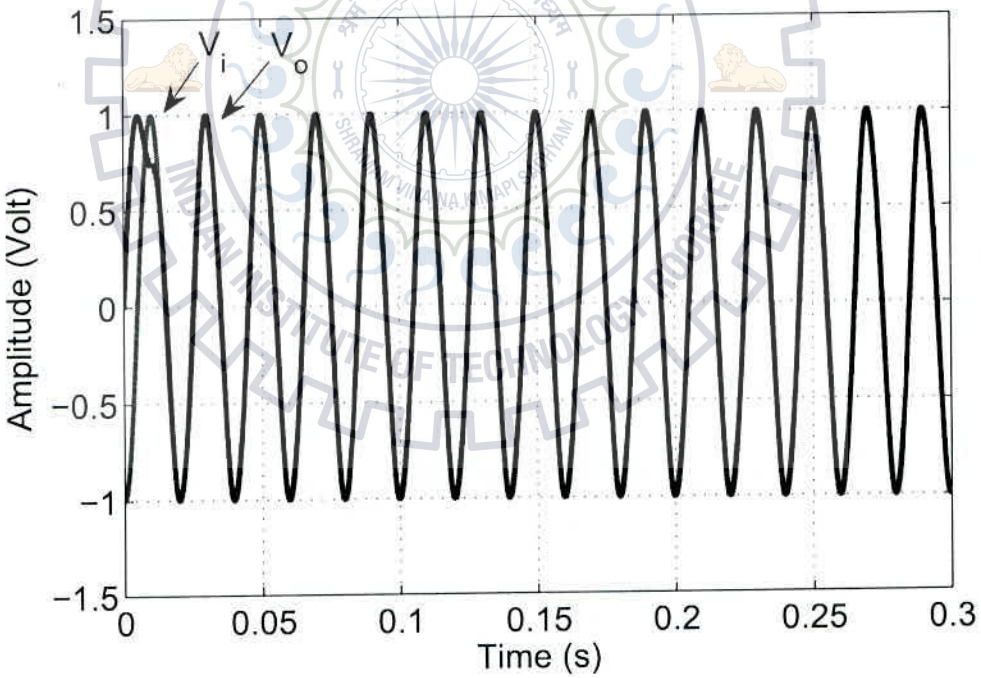

FIGURE 5.1: Input signal tracking of PLL for pure sinusoidal signal of 50 Hz frequency.

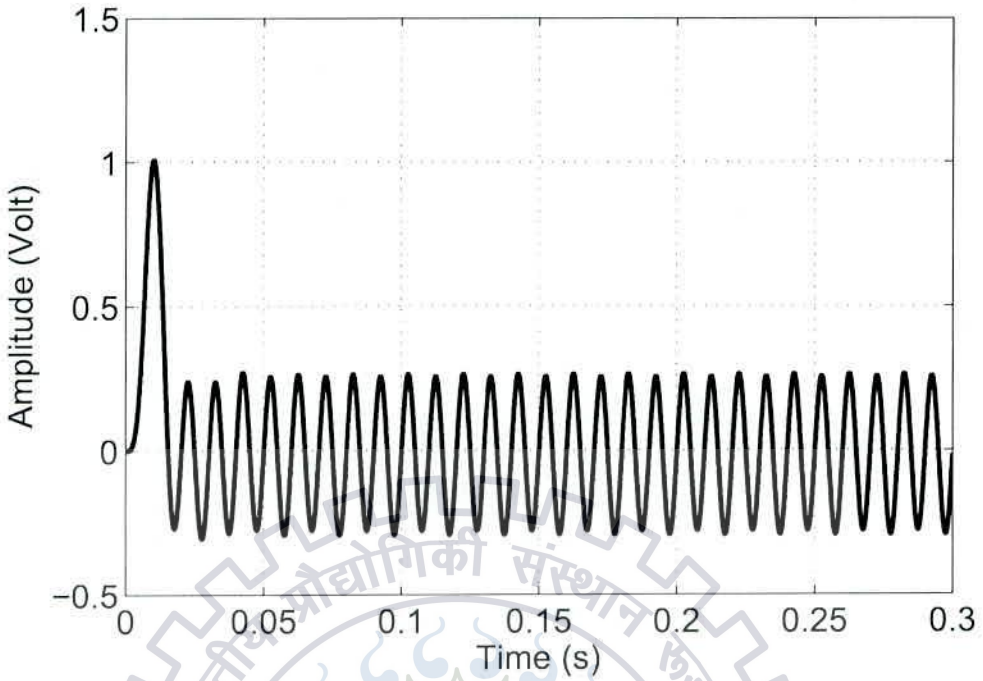

FIGURE 5.2: Control signal variation for pure sinusoidal input signal of 50 Hz frequency.

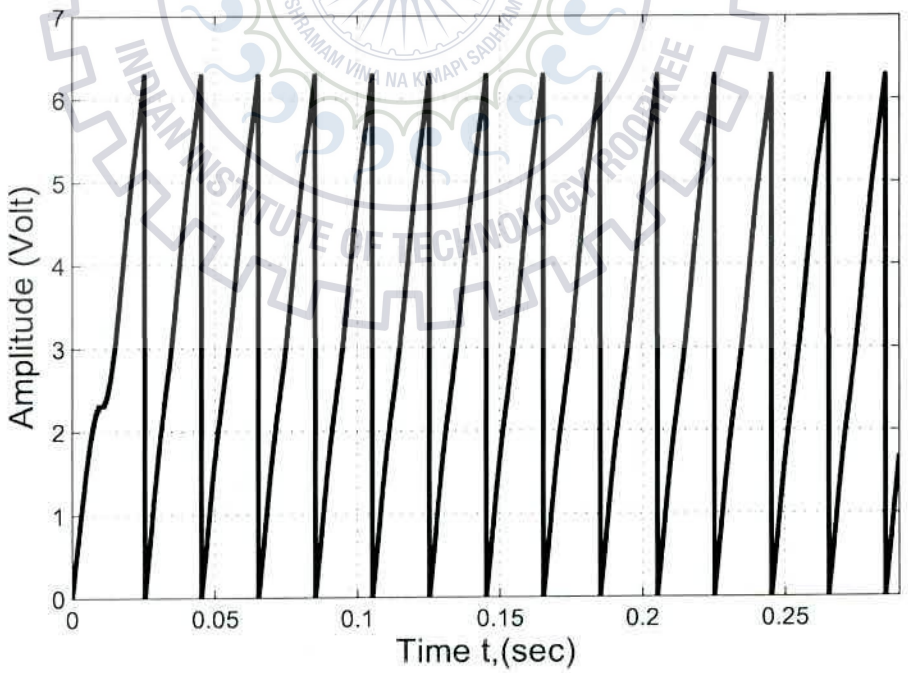

FIGURE 5.3: Phase angle variation for pure sinusoidal input signal of 50 Hz frequency.

5.1.2 Analysis in the Presence of Harmonics

The ideal environment is not easy to get and the presence of harmonics affect the system performance very badly. The Figure 5.4 on page 29 shows tracking of input signal by output signal in presence of harmonics. The input signal is contaminated with one third of third harmonics, one fifth of fifth harmonics and one seventh of seventh harmonics. In this case the tracking takes 0.02 s. The steady state error increases by some amount. The proposed scheme has the advantage that it can be able to track in presence harmonics very easily and the order of harmonics does not impose any limitation. The control signal variation in presence of harmonics is shown in Figure 5.5 on page 30, the value of steady state error increases in presence of harmonics. This PLL is able to track the fundamental signal in presence of any harmonic.

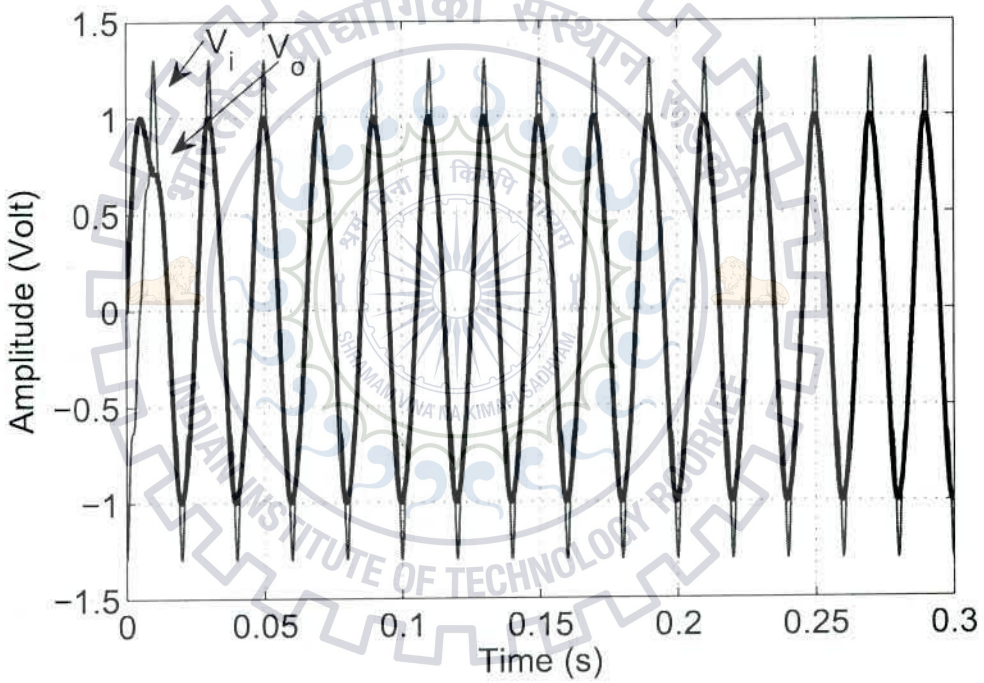

FIGURE 5.4: PLL tracking of fundamental signal 50 Hz in the presence of harmonics.

5.1.3 Linearly Varying Frequency

The proposed scheme is very well able to track the input signal when it is changing linearly. The frequency is either continuously increasing or decreasing at a constant slope. This analysis is used to find the hold-in and pull-in range of a PLL. The Figure 5.6 on page 31 shows the controlled signal variation when the frequency decreases and

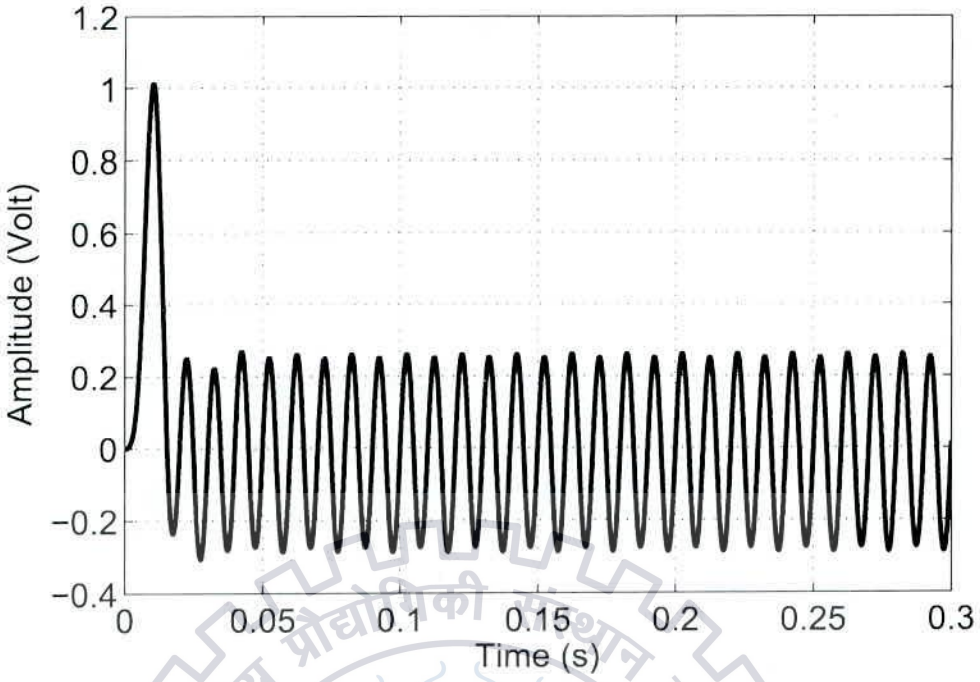

FIGURE 5.5: Control signal variation while tracking the fundamental 50 Hz signal in the presence of harmonics.

increases at a rate of 1 Hz/s. If frequency keep on increasing then the PLL starts locking at 19 Hz called as f_{p1} and it goes up to 95.7 Hz called as f_{h2}. Similarly if frequency keep on decreasing at a rate of 1 Hz/s then it starts locking at 94.5 Hz called as f_{p2} and it goes up 19.6 Hz called as f_{h1}. The hold-in range can be calculated as $f_{h2} - f_{h1} = 95.7 - 19.6 = 76.1$ Hz. The pull-in range can also be calculated as $f_{p2} - f_{p1} = 94.5 - 19 = 75.5$ Hz. In presence of harmonics the hold-in and pull-in range decreases. The lock-in range of the proposed scheme is found to be 38-68 Hz. The slope can be varied and it is found that the ranges always remain approximate same.

5.1.4 Step Change in Frequency

The one main situation in variable frequency environment is the step change in frequency. The Figure 5.7 on page 32 shows the controlled signal variation during step change in frequency. The initial frequency is at 50 Hz and then at t =1 s a sudden step change in frequency from 50 Hz to 55 Hz occurs. The output signal takes one and half cycle i.e. 0.03 s to completely adapt for the change. One more step having more value as compared to previous one in negative direction is also applied at t=1.2 s. It takes 0.035

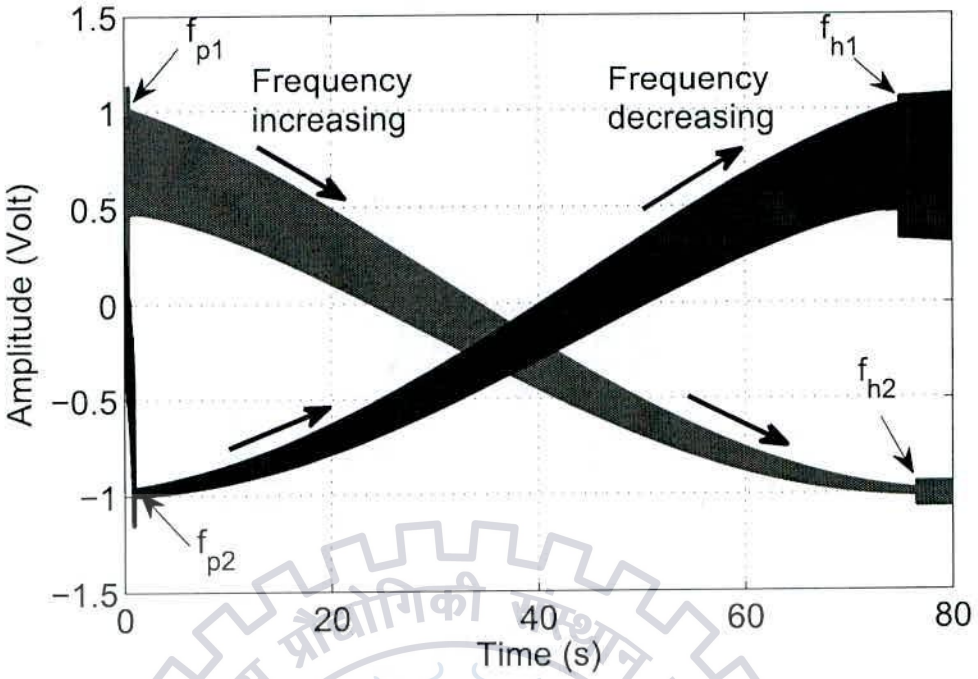

FIGURE 5.6: Control signal variation with a linear change in input frequency for a positive and negative slope of 1Hz/s.

s to adapt for the step change in frequency from 55 Hz to 45 Hz. The P and I values are set at 0.4545 and 18 respectively.

The system is also able to adapt for very wide step change in frequency. The center frequency is chosen at 50 Hz and then step change in both positive and negative directions are applied. The maximum step that the proposed scheme is able to lock in positive direction is from 50 Hz to 91.6 Hz and in negative direction is from 50 Hz to 21 Hz. The Figure 5.8 on page 32 shows that at t=1 s a positive step from 50 Hz to 91.6 Hz is applied and it takes around 0.82 s to completely track the step change. It also shows that when a step change in negative direction from 50 Hz to 21 Hz is applied and it takes 0.35 s to completely lock the signal.

Under this step change in frequency there is one more condition when system is not working at fixed center frequency of 50 Hz. As this scheme is able to track signal from DC to second harmonics. The maximum frequency value that it can starts working with maintaining its tracking behavior is 85 Hz. The system is running at 85 Hz and then a step change is applied at t=2 s from 85 Hz to 22 Hz and it takes 1.8 s to adapt for this wide change in step. This is all clear by seeing Figure 5.9 on page 33. Similarly the minimum frequency from which it can starts while maintaining tracking is 21 Hz. It can

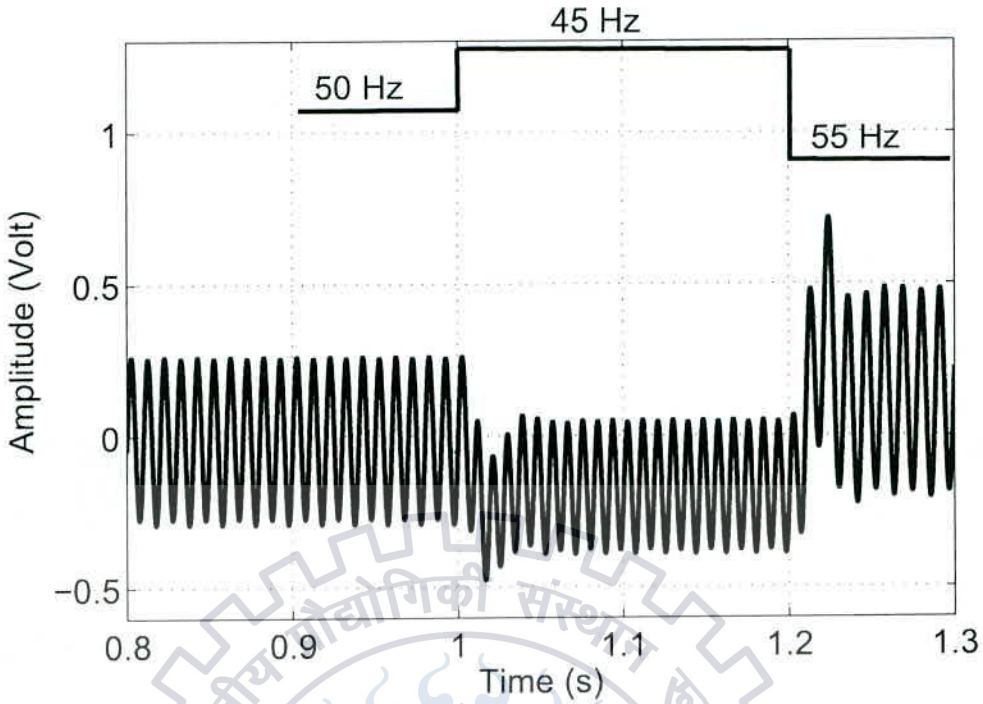

FIGURE 5.7: Control signal variation for a step change in input frequency for positive and negative directions.

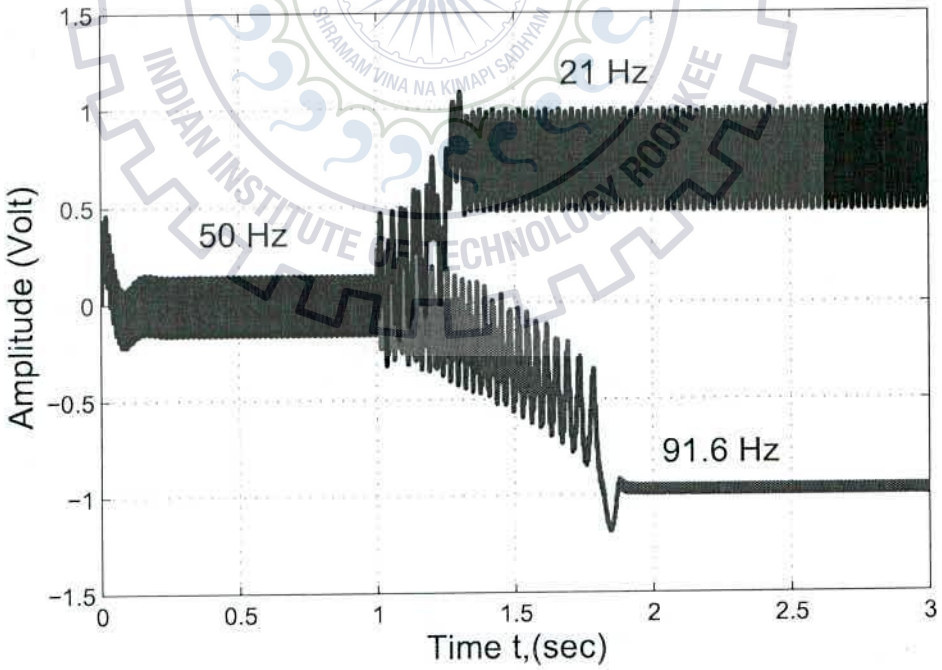

FIGURE 5.8: Control signal variation for a step change in input frequency for positive and negative directions.

go up to 89.5 Hz from 21 Hz with maintaining frequency tracking. The time taken to adapt for this step change is 2.75 s. The initial locking time in case of input signal of 21 Hz and 85 Hz is 0.35 s and 0.61 s respectively.

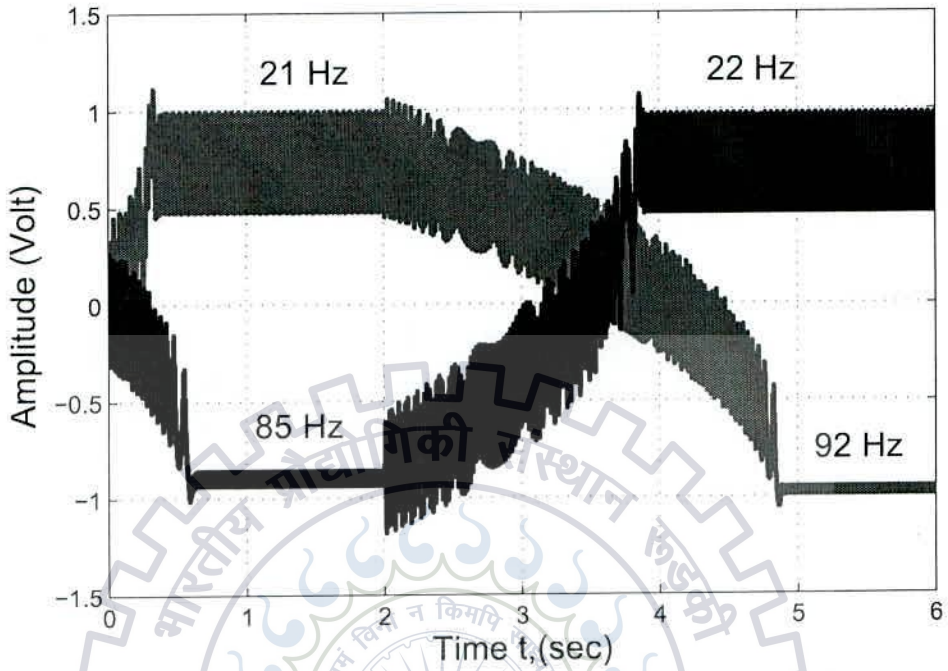

FIGURE 5.9: Control signal variation for a maximum step change in input frequency for positive and negative directions.

5.1.5 Step Change in Phase

The system is also able to lock the input after any sudden step change in phase occurs. The Figure 5.10 on page 34 shows that initially when there was no phase change then output keeps on tracking the input continuously. At t = 1 s a step change in phase from 0 rad to $2\pi/3$ rad is applied. One more step change in phase at t = 1.3 s from $2\pi/3$ to $-3\pi/2$ is applied. In both the cases the output takes 0.1 s to completely track the input.

5.1.6 Steady state count

Steady state count means the number of pulses that the signal is taking with in 1 s of time. Since the input is sampled at two times higher the Nyquist rate and then the synchronized signal is N times the sampled signal ($N = 128$). Suppose the input signal

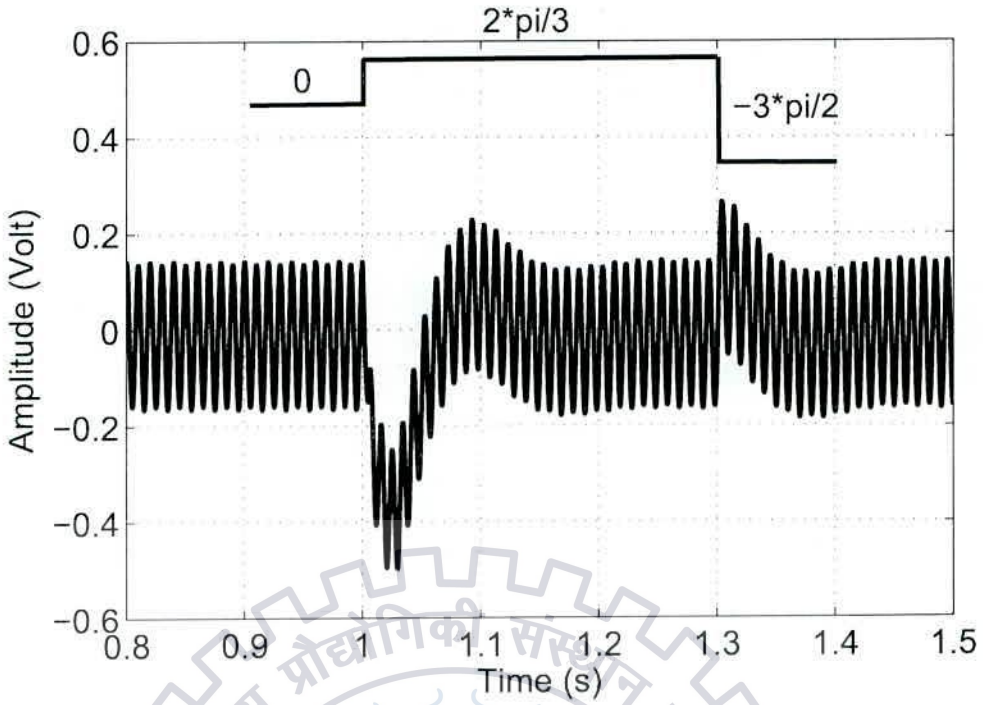

FIGURE 5.10: Control signal variation for a step change in input phase for positive and negative directions.

is of 50 Hz then sampling it two times higher the Nyquist rate i.e. $2 \times (2 \times 50) = 200$ Hz and then the grid synchronized signal is $N \times f_S$ i.e. 128 * 200 = 25600 Hz where f_S is sampling frequency. The number of pulses under ideal situation is $(N \times f_S)/4 = (128 \times 200)/4 = 6400$. The table 5.1 shows the number of pulses ideally applied, the count that achieved and finds the difference between them.

5.2 Simulation Results using DSP Builder

The above simulation are done at 50 Hz input. The proposed scheme has the advantage that it can be used for any frequency range. All the simulations are carried out considering the center frequency at 10 kHz. Some important simulation results are also shown in this regard. The proposed system is implemented in Simulink using DSP Builder from Altera®. The DSP Builder acts as an interface between MathWorks® Simulink and Altera® Quartus II software. The DSP Builder generates VHDL or Verilog file as this software integrates the simulation capabilities of MATLAB, system design power of Simulink, Verilog HDL and VHDL design flows with Altera Quartus II software. The DSP Builder helps in implementing the scheme on hardware and in doing synthesis. The Simulink model files are stored with .mdl extension and once the design is simulated in

TABLE 5.1: No. of pulses applied and achieved

S. No.	Frequency Applied (Hz)	Ideal Count	Frequency Achieved (Hz)	Count Achieved	Count Difference
1	40	5120	40.3125	5160	40
2	41	5248	41.3203	5289	41
3	42	5376	42.3281	5418	42
4	43	5504	43.3359	5547	43
5	44	5632	44.3437	5676	44
6	45	5760	45.3515	5805	45
7	46	5888	46.3593	5934	46
8	47	6016	47.3671	6063	47
9	48	6144	48.3750	6192	48
10	49	6272	49.3828	6321	49
11	50	6400	50.3906	6450	50
12	51	6528	51.3984	6579	51
13	52	6656	52.4062	6708	52
14	53	6784	53.4140	6837	53
15	54	6912	54.4218	6966	54
16	55	7040	55.4296	7095	55
17	56	7168	56.4375	7224	56
18	57	7296	57.4453	7353	57
19	58	7424	58.4531	7482	58
20	59	7552	59.4609	7611	59
21	60	7680	60.4687	7740	60

MATLAB it can be shifted to DSP Builder environment where all files are designed by using DSP Builder library. One thing to be sure that all the blocks are used from DSP Builder library or MegaCore® blocks. The .mdl file is read by DSP Builder Signal Compiler block. The Signal Compiler generates the Tool Command Language (.Tcl) files and VHDL files which are further used for design synthesis, hardware implementation, and simulation [30].

The system level design flow in DSP Builder is as follows

1. The first step is to create the design in MATLAB Simulink environment and simulate it.

2. To develop the design in DSP Builder environment by using Altera DSP Builder Block-set.

3. Include Clock and Signal Compiler block in the design. The clock that I have used in my design is 20 ns.

4. Once the design is ready in DSP Builder, run the Signal Compiler block.

5. Select the desired board on which the design is going to be implemented. The board used in this dissertation is from Stratix III series and is Stratix III 3SL150 FPGA Development Board.

6. Next step is to compile the design.

7. The compilation step is divided into three parts i.e. Quartus II Synthesis, ATOM Netlist, and Quartus II Fitter.

8. After the completion of compilation, click on Scan Jtag (Joint test action group). It will detect USB Blaster II and device EP3SL150

9. Finally click the Program icon to download the generated VHDL codes from the computer to the FPGA.

The Figure 5.11 on page 37 shows the basic building block of the proposed scheme using DSP Builder software. The Figure 5.12 on page 38 shows the simulation of 10 kHz input and its tracking by the output. It is clear from the fig. that it needs almost three cycles to track the input signal. Similarly other results can also be verified using DSP Builder on 10 kHz. The phase angle variation is shown in Figure 5.13 on page 38

5.3 Hardware Implementation

The design simulated using DSP Builder is shown in Figure 5.12 on page 38 where the input signal frequency is set to 10 kHz. For implementing it on hardware and observing it on digital storage oscilloscope (DSO), before feeding the output to scope, first it is converted into unsigned integer by masking. Masking is done by multiplying the output with a gain value and then EX-ORing it with the defined or preset binary

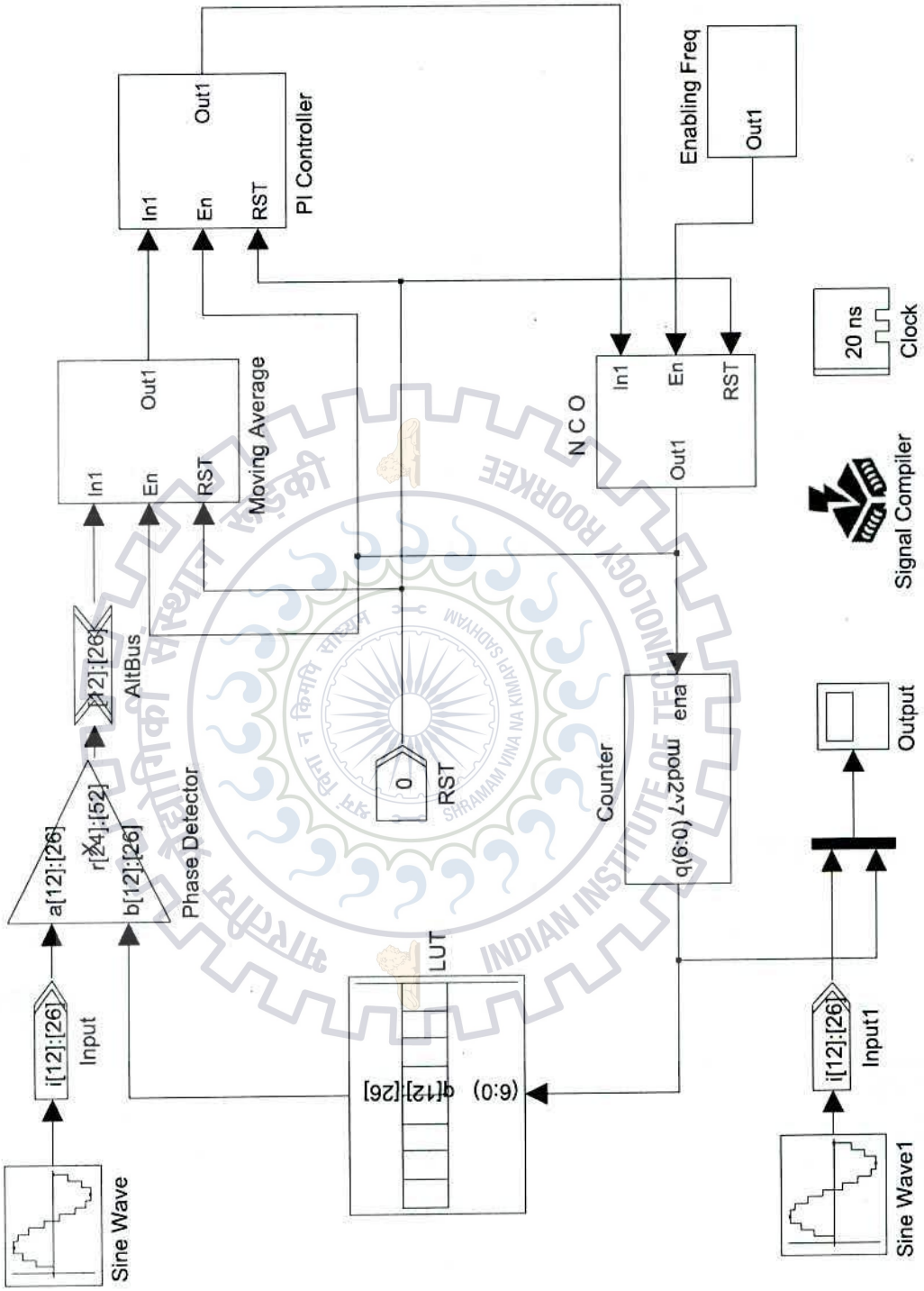

FIGURE 5.11: Block diagram of proposed scheme using DSP builder.

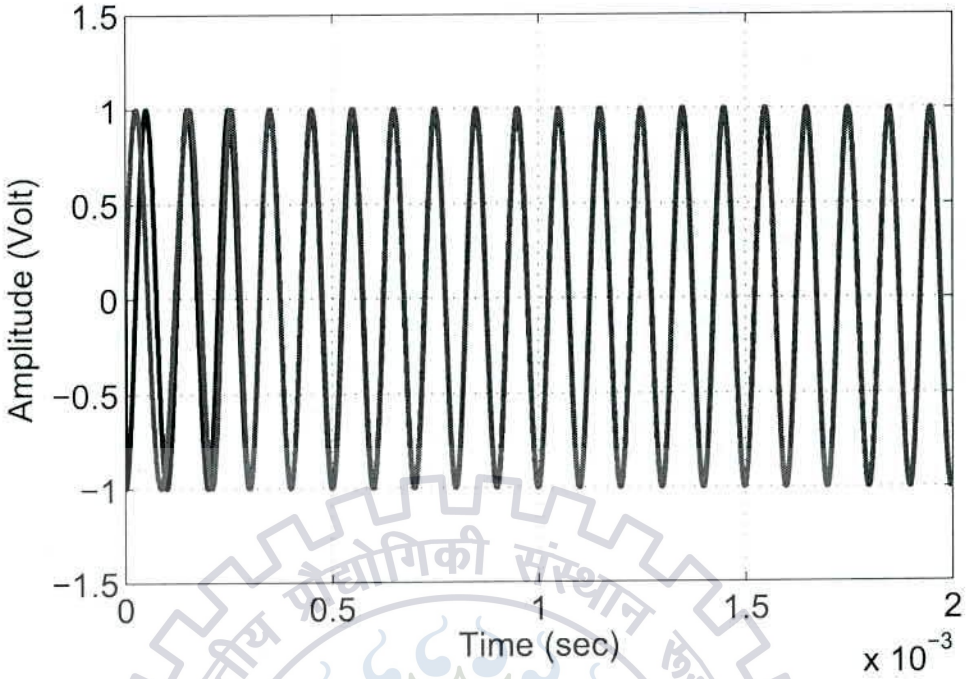

FIGURE 5.12: Tracking of 10 kHz signal by PLL output using DSP builder.

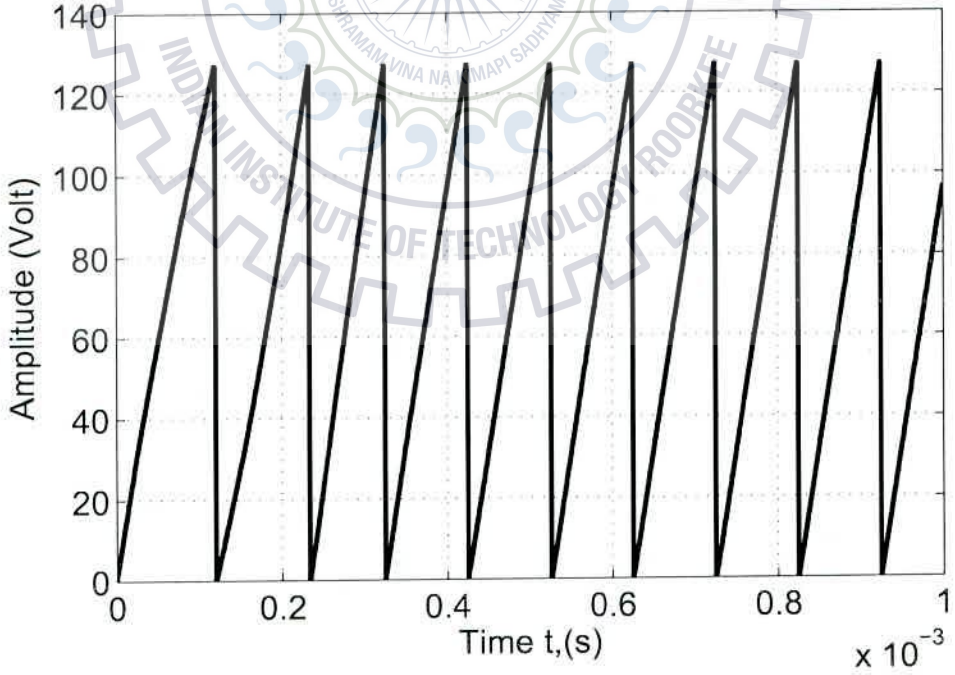

FIGURE 5.13: Phase angle variation of PLLL output signal using DSP Builder.

value. The gain value is selected to 8191 and the mask value for EX-ORing is fixed to decimal 8192 (or in binary 10000000000000). Then the masked output is applied to a digital to analog converter. The FPGA board that is used in this dissertation is Stratix III 3SL150 and DSP Builder supports it. The High Speed Mezzanine Card (HSMC) integrated with Stratix III 3SL150 FPGA Development Board has two digital to analog converter. The results are recorded on DSOX2014A oscilloscope. The input signal after masking is connected to first D/A convertor (D2A1 HSMC A 14 Bit Unsigned) while the output of PLL after masking is connected to second D/A convector (D2A2 HSMC A 14 Bit Unsigned). These D/A converter connects to the digital-to-analog 14-bit unsigned output bus on DSP Daughter Card.

5.3.1 Tracking of Fundamental 10 kHz Signal

The input is applied on channel 1 and PLL output on channel 2 and as it is clear from Figure 5.14 on page 40. The above one with yellow color is the applied input and the below one with green color is the output of PLL. The amplitude of input and output signal is 200 mv and it can be seen on the top left hand of the Figure. The frequency can be seen and verified from the Figure 5.14. The Figure 5.15 on page 40 shows the completely locking state. The perfect locking can be seen there and it is so fine that we are not able to distinguish between input and output signal very easily. The phase angle variation with the output signal is shown in Figure 5.16 on page 41, the PLL output is applied on channel 1 on DSO and phase angle variation are measured on channel 2. The phase angle is the output of a counter so it is counting up to maximum value and then it comes to zero.

5.3.2 Step Change in Frequency

The input is applied with positive step change as well as with negative step change in frequency and these steps are of 2 kHz step sizes. The Figure 5.17 on page 41 shows when a step from 8 kHz to 10 kHz is applied. The system is locked with 8 kHz and then a sudden step change is applied. These Figures are divided into two parts, the above part is showing the input and and output for whole range while the down part is showing the zoomed portion. The input and output amplitude in this case is set to 100 mv. The Figure 5.18 on page 42 shows when a step from 10 kHz to 12 kHz is applied.

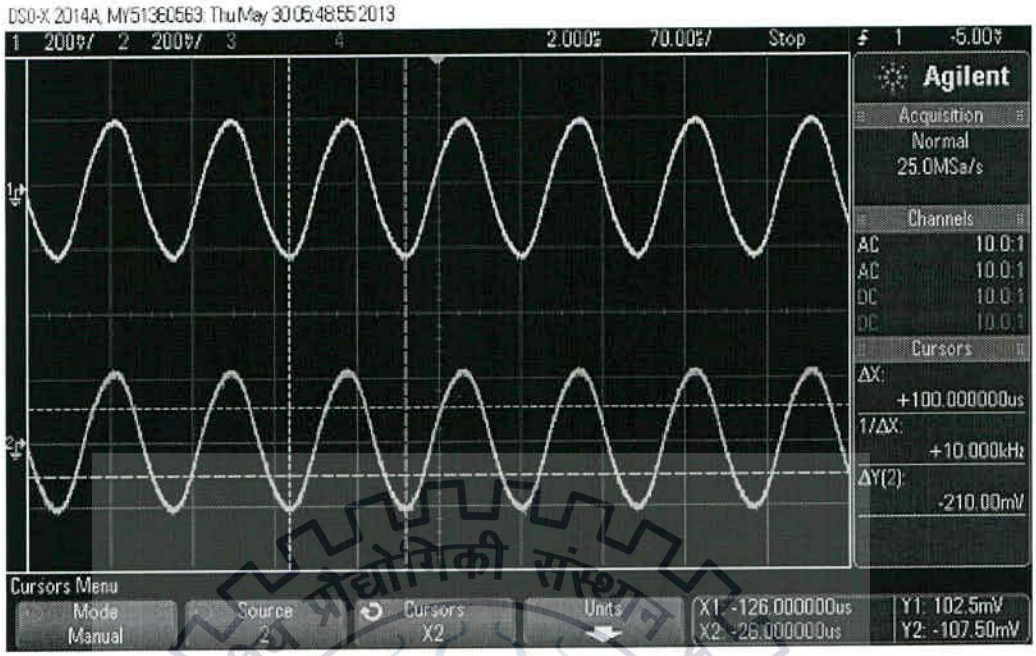

FIGURE 5.14: Tracking of 10 kHz input signal by PLL output signal.

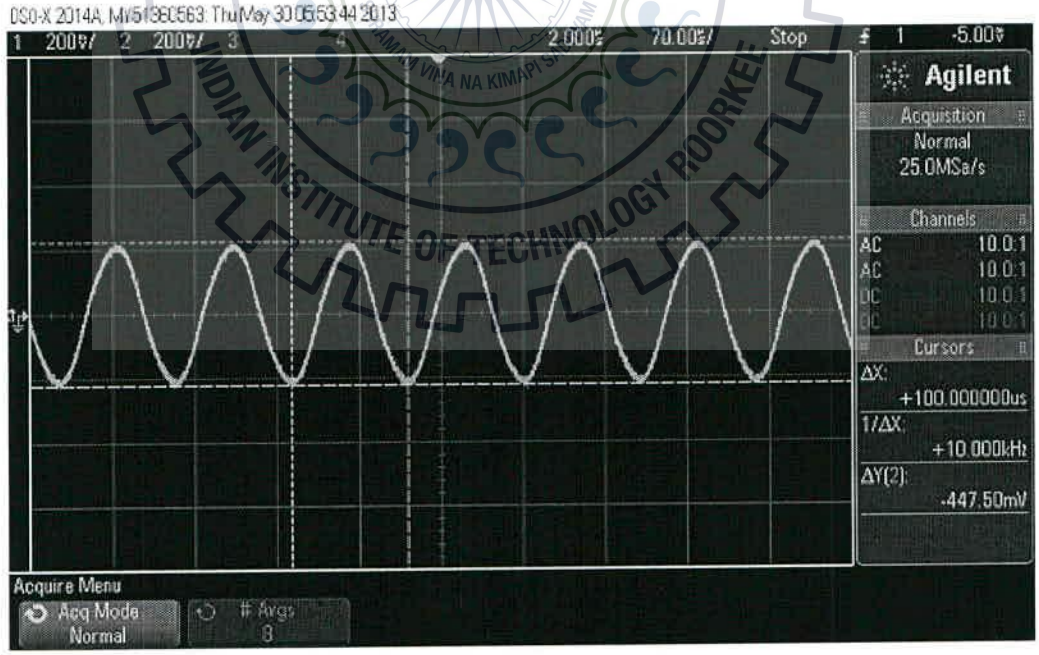

FIGURE 5.15: Completely locked input signal and PLL output signal.

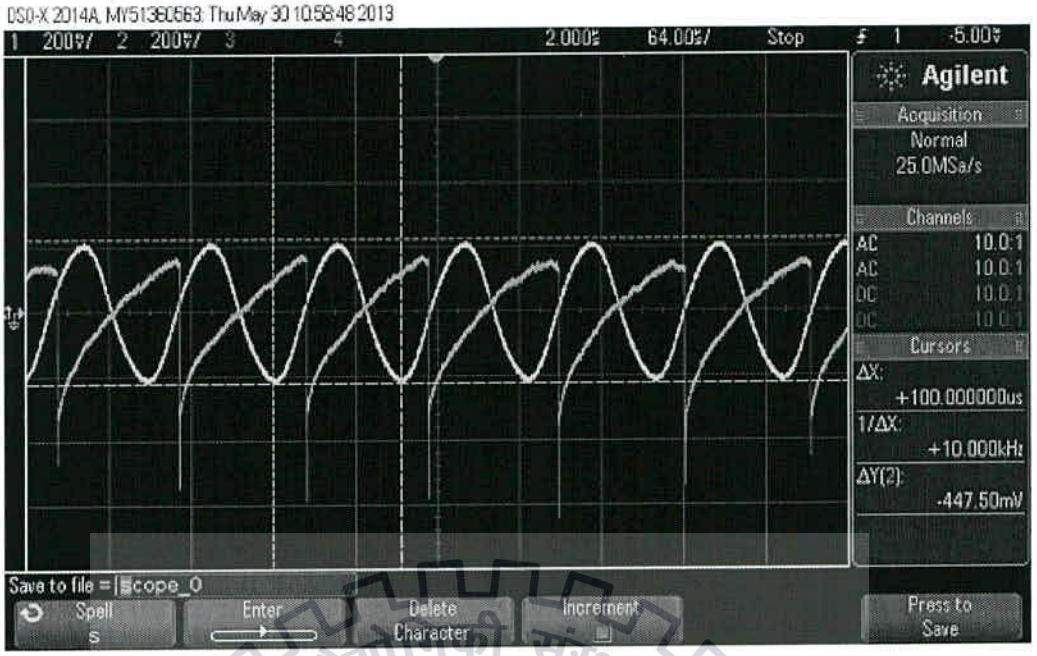

FIGURE 5.16: PLL output signal and phase angle variation.

The Figure 5.19 on page 42 shows a step change in frequency by 2 kHz from 12 kHz to 10 kHz and then Figure 5.20 on page 43 shows step change from 10 kHz to 8 kHz.

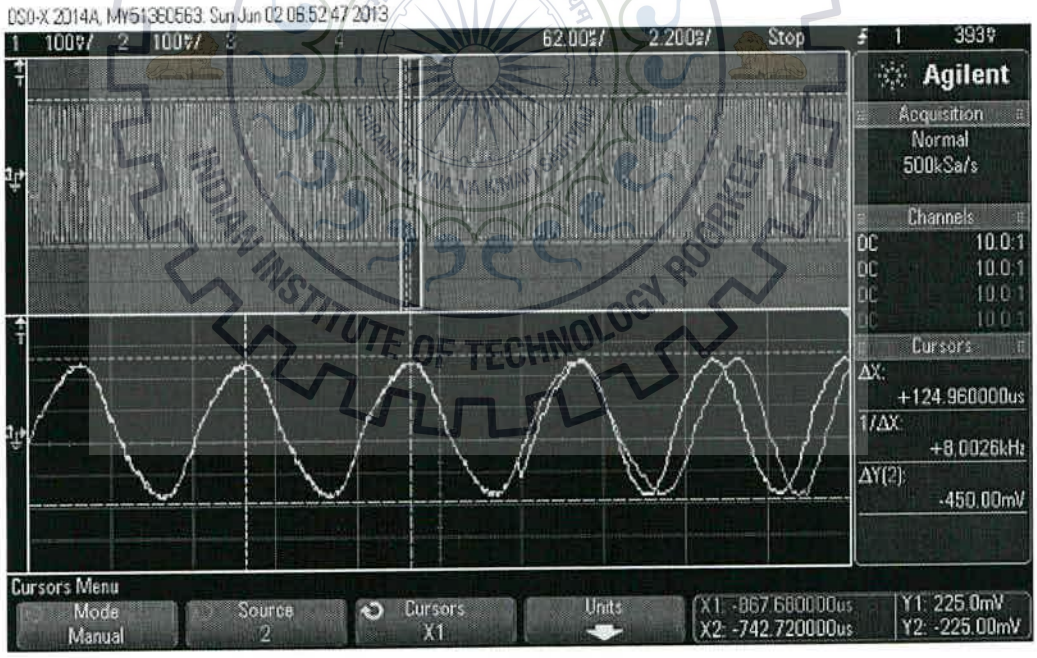

FIGURE 5.17: Tracking of sudden step change in frequency from 8 kHz to 10 kHz.

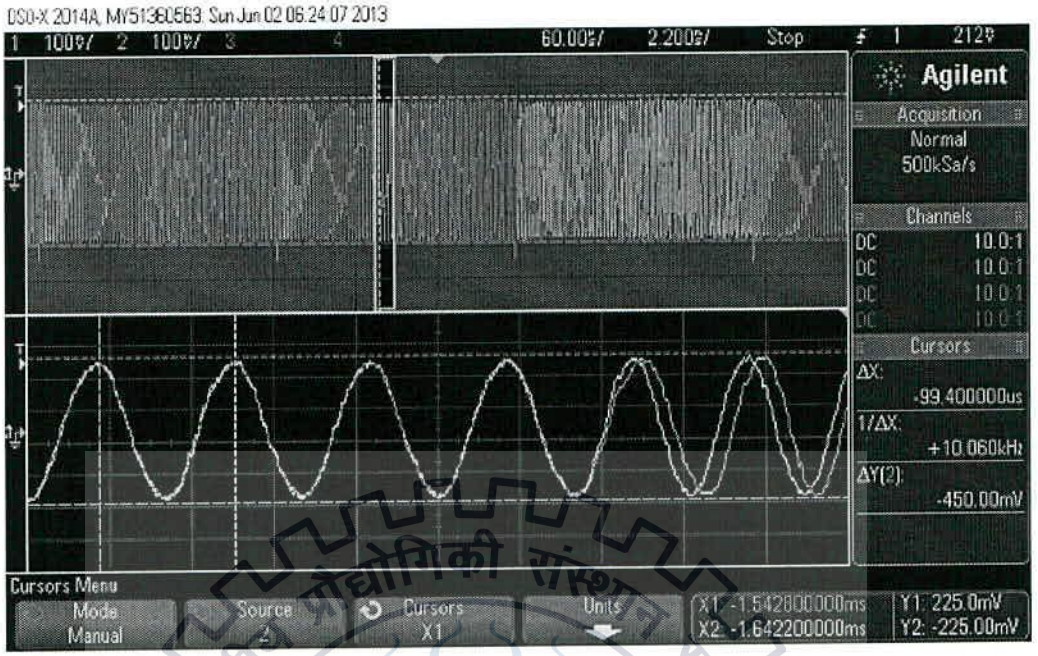

FIGURE 5.18: Tracking of sudden step change in frequency from 10 kHz to 12 kHz.

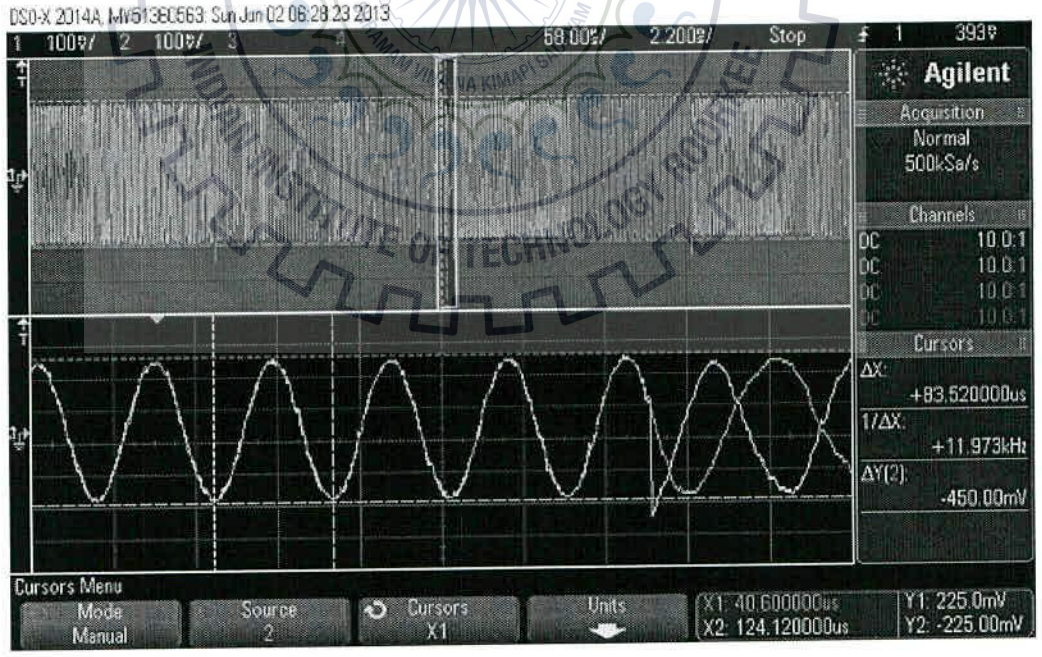

FIGURE 5.19: Tracking of sudden step change in frequency from 12 kHz to 10 kHz.

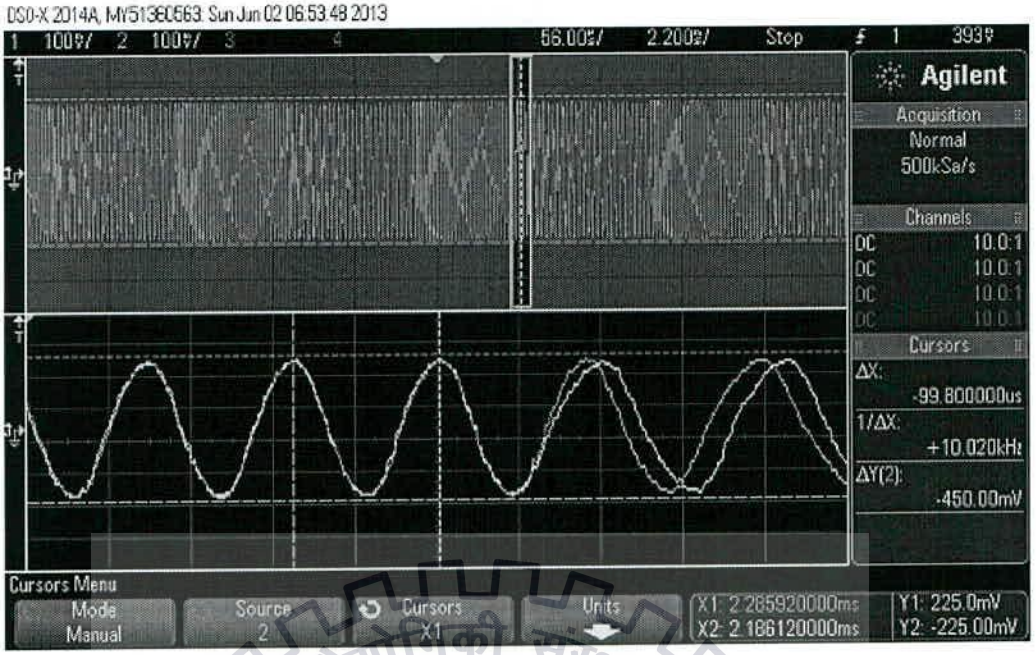

FIGURE 5.20: Tracking of sudden step change in frequency from 10 kHz to 8 kHz.

5.3.3 Step Change in Phase

The input is also tested for step change in phase. Initially the 10 kHz input is in locked condition and the a sudden step change in phase from 0 rad to $2\pi/3$ rad is applied. The Figure 5.21 on page 44 shows this condition. It takes four and half cycle for the output signal to completely track the input signal. The time taken by the output in tracking can be calculated from the zoomed figure itself, one box in zoomed figure corresponds to 99.16 μ sec and the output signal takes four and half boxes i.e. $4.5 * 78 * 10^{-6} = 0.35$ ms. The control signal variation with the output signal is shown in Figure 5.22 on page 44. The sudden peak in control signal shows the instant when step change in phase is applied. The frequency measured after four cycle clears that the steady state frequency is 10 kHz.

5.3.4 Linearly Varying Frequency

The proposed scheme is also very well able to track the input signal when its frequency keep on changing linearly. If either the frequency keeps on increasing of decreasing, in both the situation the PLL maintains tracking of varying input signal. The Figure 5.23 on page 45 shows the tracking of input signal when its frequency is increasing at a rate of 400 kHz per s. The rate of change of frequency is very fast and the proposed scheme

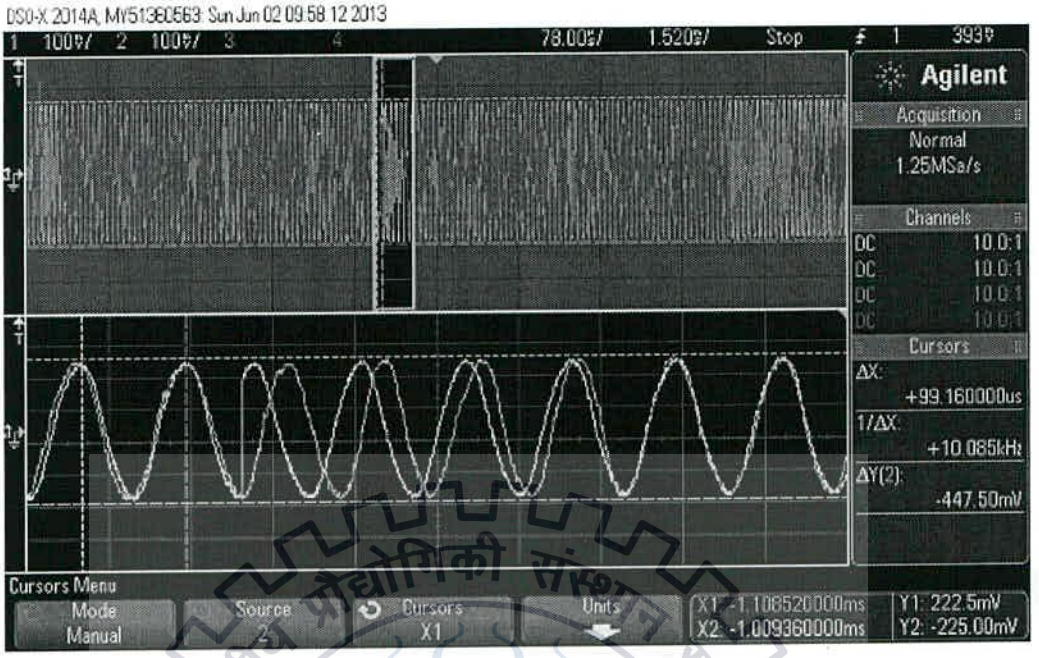

FIGURE 5.21: Tracking of sudden step change in phase from 0 rad to $2\pi/3$ rad.

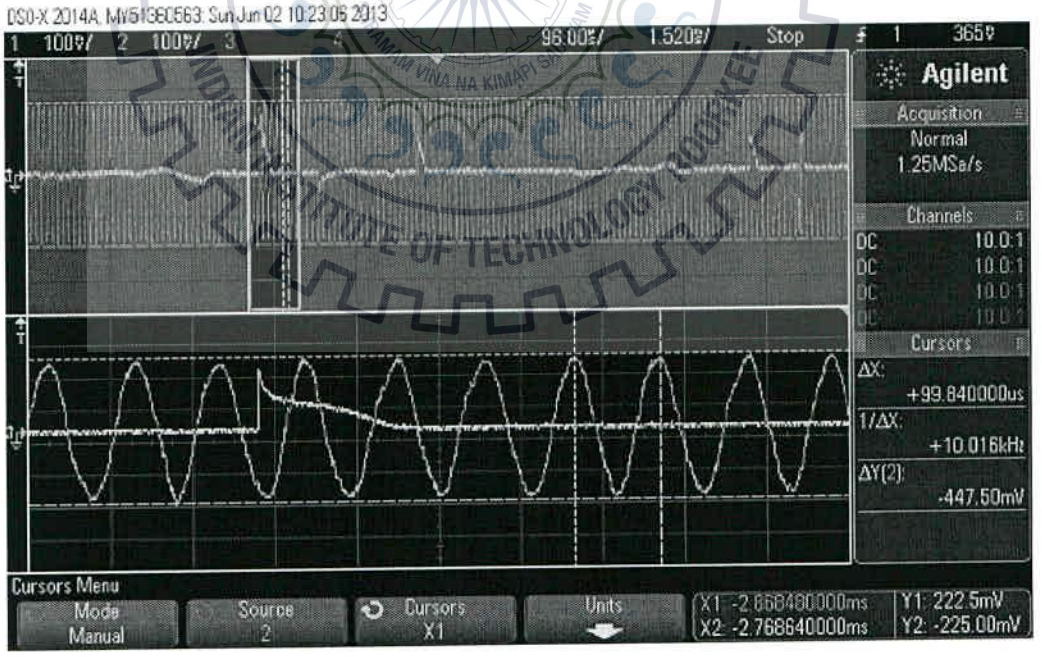

FIGURE 5.22: Control signal variation with output signal for step change in phase.

works very well under this situation also. The input signal shown is varied from 10 kHz to 12.56 kHz with in 0.64 s at a rate of 400 kHz per sec. The frequency s.The Figure 5.24 on page 46 shows the tracking of output signal by input signal when its frequency decreases at the same rate of 400 kHz. The input frequency is taken at 10 kHz and it varied at a rate of -400 kHz (minus sign shows decrease in frequency) from 10 kHz to 7.44 kHz with in 0.64 s.

FIGURE 5.23: PLL output signal tracking the input signal for linear increase in frequency from 10 Hz to 12.56 kHz.

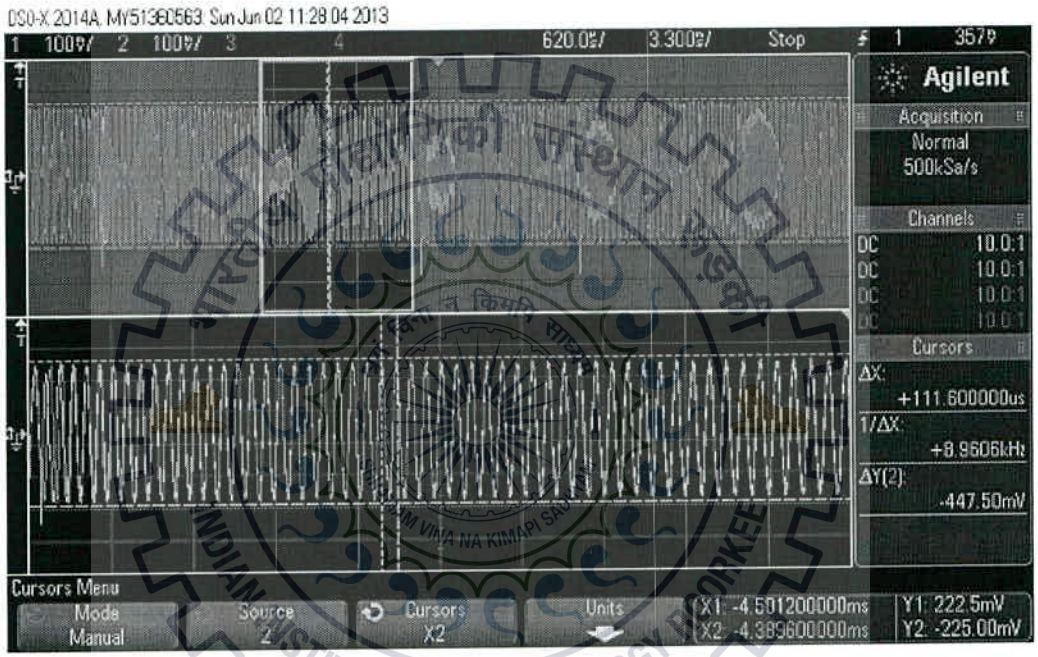

FIGURE 5.24: PLL output signal tracking the input signal for linear decrease in frequency from 10 kHz to 7.44 kHz

Chapter 6

Application and Future Scope

"The mind is everything. What you think you become."

-Buddha

This section describes an particular application of the proposed scheme. The proposed scheme can be used for the purpose of FM demodulation. A frequency modulated signal is applied as an input to the system and the system keeps on tracking the carrier signal. The controlled signal will be the desired message signal. The value of α shows the modulating signal.
The applied frequency modulated input to the system is

$$V_{FM}(t) = Sin(2 * \pi * 49000 * t + 5 * Sin(2 * \pi * 1000 * t)) \tag{6.1}$$

where the carrier frequency is 49 kHz and peak amplitude is 1 V. The frequency of modulating signal is 1 kHz and modulation index β is 5. Then the value of α will be recovered modulating signal and is given by

$$\alpha = V_m Cos(2 * \pi * 1000 * t) \tag{6.2}$$

The Figure 6.1 shows simulation reults where the modulating signal is extracted.

The proposed scheme can be used under the situation when the modulating signal is varying in terms of phase and frequency. The estimation of the modulating signal

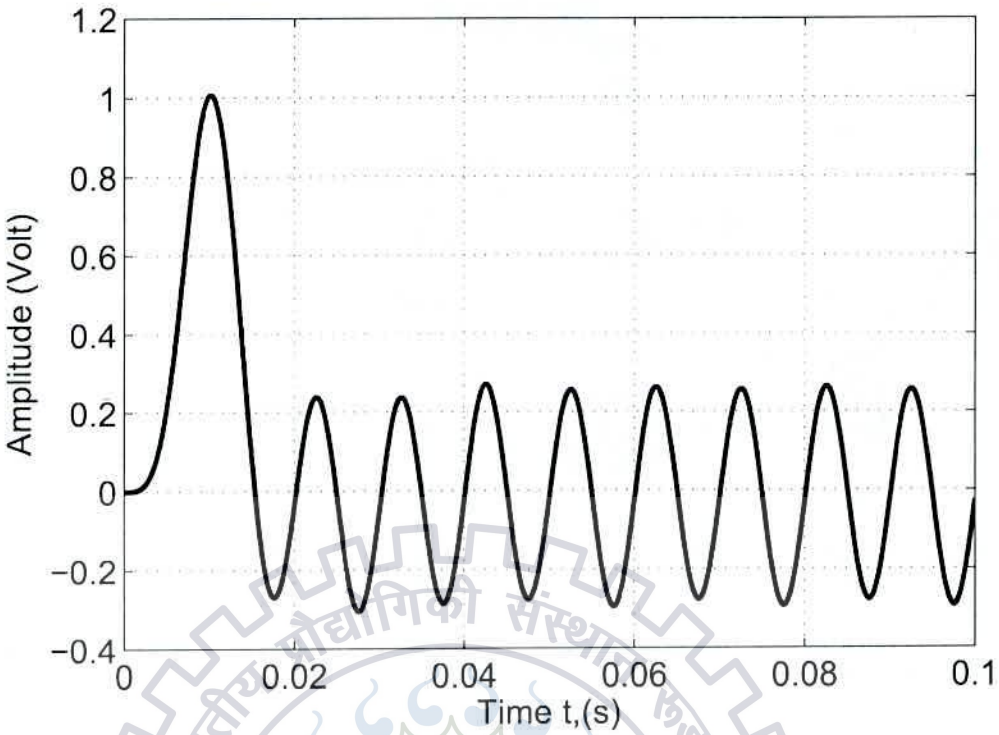

FIGURE 6.1: Extarction of modulating signal from FM modulated signal

under these situation can also be done. All the conditions that have tested can also be applied on modulating signal as well as on modulated signal [31]. The estimation of the amplitude can also be done and it will give important information. Further the time domain anlysis can be very helpful in extracting and estimating the behaviour of PLL.

The scheme can also be implemented for three phase signals and different grid fault situation can also be tested.

Chapter 7

Conclusion

"In the end, it's not the years in your life that count. It's the life in your years."

-Abraham Lincoln

The novel adaptive sampling frequency based phase locking loop is proposed. The scheme is simulated using MATLAB Simulink and then designed using DSP Builder. The locally generated reference signal updates the phase information by N times per period of the input signal. The adaptive PLL is able to track the input signal within one cycle of the fundamental frequency. The simplicity of the phase detector, rejection of higher terms in the error signal by moving average filter and the concept of adaptive sampling frequency, all together make it structurally simple and robust.

The scheme is hardware implemented in Stratix III 3SL150 FPGA Development Board and tested for different fault conditions. The scheme works very well under noisy and polluted environment. There are various situation in noisy environment like frequency step, phase step, and frequency varying continuously, the proposed scheme is tested and implemented in every case. The proposed scheme has wide lock-in and operating range from DC to second harmonics. The acquisition time in ideal as well as in variable frequency environment is very less. Finally the application in FM demodulation and grid synchronization are presented and simulated.

Publications

1. Vineet Vajpayee and P Sumathi ,"Adaptive Sampling Frequency based Phase Locking Scheme for Single-phase Grid Converters", in *IEEE International Conference on Communication and Signal Processing, ICCSP'13*,(to be published)

Bibliography

[1] J. Lyons and V. Vlatkovic, "Power electronics and alternative energy generation," in *Power Electronics Specialists Conference, 2004. PESC 04. 2004 IEEE 35th Annual*, vol. 1, 2004.

[2] A. Timbus, M. Liserre, R. Teodorescu, and F. Blaabjerg, "Synchronization methods for three phase distributed power generation systems - an overview and evaluation," in *Power Electronics Specialists Conference, 2005. PESC '05. IEEE 36th*, 2005, pp. 2474–2481.

[3] V. Kaura and V. Blasko, "Operation of a phase locked loop system under distorted utility conditions," in *Applied Power Electronics Conference and Exposition, 1996. APEC '96. Conference Proceedings 1996., Eleventh Annual*, vol. 2, 1996, pp. 703–708.

[4] S. Haykin, *Communication Systems*, 3rd ed. New York: Wiley, 1994.

[5] P. Hasan, "Intermodulation distortion of the pll fm demodulator in the presence of fm interference," in *Electronics, Circuits, and Systems, 1996. ICECS '96., Proceedings of the Third IEEE International Conference on*, vol. 2, 1996, pp. 635–638 vol.2.

[6] G. Kolumban, "Phase-locked loops," in *The Encyclopedia of RF and Microwave Engineering*, K. Chang, Ed. Wiley New York, 2005, vol. 4, pp. 3735–3767.

[7] M.-F. Lai and M. Nakano, "Special section on phase-locked loop techniques," *Industrial Electronics, IEEE Transactions on*, vol. 43, no. 6, pp. 607–608, 1996.

[8] W. Lindsey and C. M. Chie, "A survey of digital phase-locked loops," *Proceedings of the IEEE*, vol. 69, no. 4, pp. 410–431, 1981.

[9] G.-C. Hsieh and J. Hung, "Phase-locked loop techniques. a survey," *Industrial Electronics, IEEE Transactions on*, vol. 43, no. 6, pp. 609–615, 1996.

[10] D. Abramovitch, "Phase-locked loops: A control centric tutorial," *Proceedings of the American Control Conference*, vol. 1, pp. 1–15, May 2002.

[11] R. Santos Filho, P. Seixas, P. Cortizo, L. A. B. Torres, and A. Souza, "Comparison of three single-phase pll algorithms for ups applications," *Industrial Electronics, IEEE Transactions on*, vol. 55, no. 8, pp. 2923–2932, 2008.

[12] M. Karimi-Ghartemani and M. Iravani, "A new phase-locked loop (pll) system," in *Circuits and Systems, 2001. MWSCAS 2001. Proceedings of the 44th IEEE 2001 Midwest Symposium on*, vol. 1, 2001, pp. 421–424.

[13] M. Karimi-Ghartemani, H. Karimi, and M. Iravani, "A magnitude/phase-locked loop system based on estimation of frequency and in-phase/quadrature-phase amplitudes," *Industrial Electronics, IEEE Transactions on*, vol. 51, no. 2, pp. 511–517, 2004.

[14] S. Silva, B. Lopes, B. J. C. Filho, R. Campana, and W. Bosventura, "Performance evaluation of pll algorithms for single-phase grid-connected systems," in *Industry Applications Conference, 2004. 39th IAS Annual Meeting. Conference Record of the 2004 IEEE*, vol. 4, 2004, pp. 2259–2263.

[15] A. Nicastri and A. Nagliero, "Comparison and evaluation of the pll techniques for the design of the grid-connected inverter systems," in *Industrial Electronics (ISIE), 2010 IEEE International Symposium on*, 2010, pp. 3865–3870.

[16] M. Ciobotaru, V. Agelidis, and R. Teodorescu, "Accurate and less-disturbing active anti-islanding method based on pll for grid-connected pv inverters," in *Power Electronics Specialists Conference, 2008. PESC 2008. IEEE*, 2008, pp. 4569–4576.

[17] T. Thacker, R. Wang, D. Dong, R. Burgos, F. Wang, and D. Boroyevich, "Phase-locked loops using state variable feedback for single-phase converter systems," in *Applied Power Electronics Conference and Exposition, 2009. APEC 2009. Twenty-Fourth Annual IEEE*, 2009, pp. 864–870.

[18] F. Xiong, W. Yue, L. Ming, and L. Jinjun, "A novel frequency-adaptive pll for single-phase grid-connected converters," in *Energy Conversion Congress and Exposition (ECCE), 2010 IEEE*, 2010, pp. 414–419.

[19] B. McGrath, D. Holmes, and J. Galloway, "Power converter line synchronization using a discrete fourier transform (dft) based on a variable sample rate," *Power Electronics, IEEE Transactions on*, vol. 20, no. 4, pp. 877–884, 2005.

[20] M. Perez, J. Espinoza, L. Moran, M. Torres, and E. Araya, "A robust phase-locked loop algorithm to synchronize static-power converters with polluted ac systems," *Industrial Electronics, IEEE Transactions on*, vol. 55, no. 5, pp. 2185–2192, 2008.

[21] I. Carugati, P. Donato, S. Maestri, D. Carrica, and M. Benedetti, "Frequency adaptive pll for polluted single-phase grids," *Power Electronics, IEEE Transactions on*, vol. 27, no. 5, pp. 2396–2404, 2012.

[22] I. Carugati, S. Maestri, P. Donato, D. Carrica, and M. Benedetti, "Variable sampling period filter pll for distorted three-phase systems," *Power Electronics, IEEE Transactions on*, vol. 27, no. 1, pp. 321–330, 2012.

[23] R. E. Best, *Phase-Locked Loops: Design, Simulation, and Applications*, 3rd ed., 1997.

[24] S. Kadam, D. Sasidaran, A. Awawdeh, L. Johnson, and M. Soderstrand, "Comparison of various numerically controlled oscillators," in *Circuits and Systems, 2002. MWSCAS-2002. The 2002 45th Midwest Symposium on*, vol. 3, 2002, pp. III–200–III–202 vol.3.

[25] R. Lyons and A. Bell, "The swiss army knife of digital networks," *Signal Processing Magazine, IEEE*, vol. 21, no. 3, pp. 90–100, 2004.

[26] S. W. Smith, *The scientist and engineer's guide to digital signal processing*, 2nd ed., 1997.

[27] C. Turner, "Recursive discrete-time sinusoidal oscillators," *Signal Processing Magazine, IEEE*, vol. 20, no. 3, pp. 103–111, 2003.

[28] P. Sumathi and P. Janakiraman, "Integrated phase-locking scheme for sdft-based harmonic analysis of periodic signals," *Circuits and Systems II: Express Briefs, IEEE Transactions on*, vol. 55, no. 1, pp. 51–55, 2008.

[29] P. Sumathi and P. A. Janakiraman, "Fpga implementation of an amplitude-modulated continuous-wave ultrasonic ranger using restructured phase-locking scheme," *VLSI Des.*, vol. 2010, pp. 9:1–9:11, Jan. 2010.

[30] Altera, "Dsp builder standard blockset user guide," in *DSP Builder Handbook*, 2011, vol. 2.

[31] P. Sumathi, "A frequency demodulation technique based on sliding dft phase locking scheme for large variation fm signals," *Communications Letters, IEEE*, vol. 16, no. 11, pp. 1864–1867, 2012.