

# Frequency and Power Estimator for Digital Receivers in Doppler Shift Environments

**M. Saber**

*Department of Informatics  
Kyushu University  
744 Motooka, Nishi-ku, Fukuoka-shi, 89-0395, Japan*

*mohsaber@tsubaki.csce.kyushu-u.ac.jp*

**M. T. A. Khan**

*Ritsumeikan Asia Pacific University, College of Asia Pacific Studies  
1-1 Jumonjibaru, Beppu, Oita, 874-8577, Japan*

*tahir@apu.ac.jp*

**Y. Jitsumatsu**

*Department of Informatics  
Kyushu University  
744 Motooka, Nishi-ku, Fukuoka-shi, 89-0395, Japan*

*jitsumatsu@inf.kyushu-u.ac.jp*

---

## Abstract

A frequency estimator well suited for digital receivers is proposed. Accurate estimates of unknown frequency and power of input sinusoidal signal, in the presence of additive white Gaussian noise (AWGN), are provided. The proposed structure solve the problems of traditional phase locked loop (PLL) such as, narrow tracking range, overshoot, long settle time, double frequency ripples in the loop and stability. Proposed method can estimate frequencies up to half the sampling frequency irrespective of the input signal power. Furthermore, it provides stability and allows fast tracking for any changes in input frequency. The estimator is also implemented using field programmable gate array (FPGA), consumes 127 mW and works at a frequency of 225 MHz. Proposed method can estimate the fluctuation in frequency of transmitter's oscillator, can be used as a frequency shift keying receiver and can also be applied as a digital receiver in Doppler shift environment.

**Keywords:** Digital Phase Locked Loop (DPLL), Frequency Estimator, FPGA.

---

## 1. INTRODUCTION

The main objective of a receiver is to generate an accurate replica of the transmitted symbol sequence. Synchronization is very important for proper detection. It involves recovery of reference parameters from the received signal and using these parameters to demodulate and detect data [1]. Synchronization includes carrier recovery and symbol timing recovery. Carrier recovery is the estimation and compensation of carrier frequency and phase. In case of coherent detection, knowledge of both frequency and phase of the carrier is necessary. The frequency offset is generated at the receiver as a result of factors such as variations between the oscillators at transmitter and receiver, Doppler shift caused by relative motion between transmitter and receiver and the phase noise generated by other channel impairments [2]. The estimation of frequency of periodic signals in the presence of noise is also important in many other practical applications of signal processing.

PLLs are a very important building block in the modern communications. They are used in synthesizing carrier signals, demodulation and synchronization of frequency and phase of the received signal. Conventional PLL consists of three components, a phase detector (usually multiplier), a loop filter which is low pass filter (LPF) and voltage controlled oscillator (VCO) arranged in feedback manner. Conventional PLL suffers from problems such as high frequency ripples in the loop, overshoots and narrow tracking range. Higher order loop filters avoid these

problems; However use of higher order loop filters raise the issues such as stability, longer settling time, and more narrow tracking range [3:8].

We propose a simple design for frequency estimator which can estimate both frequency and power of received sinusoidal signal in the presence of background noise. Proposed algorithm can estimate wide range of frequencies up to half the sampling frequency, irrespective of the received signal power. The structure works in a feedback manner like a PLL and allows fast tracking for any changes in the received frequency; however, no loop filter is needed. The basic design is simple and comprises of only a first order system which provides stability. Computer simulations performed confirm the validity of analytical results. Furthermore, the estimator is modeled with VHDL and implemented using FPGA. Proposed structure's hardware implementation consumes 127 mW and works at a frequency of 225 MHz.

This paper is organized as follows: next section explains the working of conventional PLL and their characteristics. Section 3 presents the proposed method; explain the structure, operation and mathematical equations of proposed estimator. Computer simulations are presented in section 4. FPGA implementation is mentioned in section 5 and finally conclusions are given in section 6.

## 2. CONVENTIONAL PLL

A conventional PLL as shown in Fig. 1 consists of three main components i.e. a phase detector (multiplier), loop filter and VCO. PLL receives the input sinusoidal signal with input frequency and phase. The input signal is multiplied with the generated signal from VCO. Multiplication results in two signals the first with higher frequency term and the other with lower frequency term. Loop filter is assumed to remove the high frequency signal. The result output signal from loop filter is the low frequency signal which represents the frequency and phase difference between input and generated signals. This difference is used to control the generated phase and frequency from VCO. When there is no phase difference, VCO generates center frequency only.

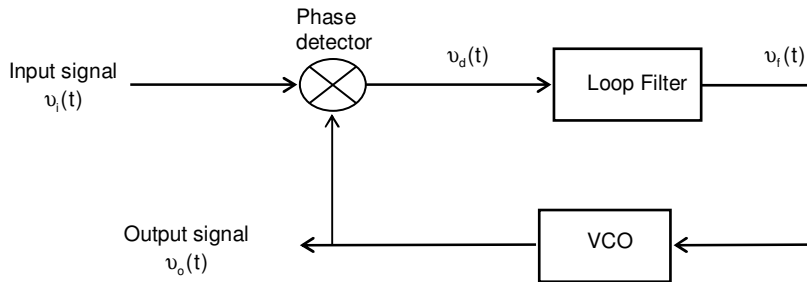


FIGURE 1: PLL Block diagram.

Assume an input sinusoidal signal

$$v_i(t) = A_i \sin(\omega_i t + \theta_i(t)) = A_i \sin \psi_i(t), \quad (1)$$

where  $\omega_i$  is the angular frequency and  $\theta_i(t)$  is the unknown phase of input signal. The signal generated by VCO is

$$v_o(t) = A_o \cos(\omega_o t + \theta_o(t)) = A_o \cos \psi_o(t), \quad (2)$$

where  $\omega_o$  is the estimation of angular frequency of VCO and  $\theta_o(t)$  is the estimated phase of VCO. Input signal is multiplied by the VCO output, then

$$\begin{aligned} v_d(t) &= k_m v_i(t) \times v_{vco}(t) \\ &= \frac{A_i A_o k_m}{2} \sin((\omega_i - \omega_o)t + \theta_i(t) - \theta_o(t)) + \frac{A_i A_o k_m}{2} \sin((\omega_i + \omega_o)t + \theta_i(t) + \theta_o(t)) \\ &= k_d [\sin(\psi_i(t) - \psi_o(t)) + \sin(\psi_i(t) + \psi_o(t))], \end{aligned} \quad (3)$$

where  $k_m$  is the gain of phase detector with dimension  $[1/V]$ ,  $k_d = \frac{A_i A_o k_m}{2}$ . In the simplest case we assume that low-pass filter removes the upper sideband with frequency  $\omega_i + \omega_o$  but passes the lower sideband  $\omega_i - \omega_o$ . VCO's tuning voltage will be

$$v_f(t) = k_d \sin(\psi_i(t) - \psi_o(t)) = k_d \sin \psi_e(t), \quad (4)$$

where  $\psi_e(t)$  is the phase difference between input and output VCO signals

$$\psi_e(t) = \psi_i(t) - \psi_o(t). \quad (5)$$

This difference will be used to control the frequency and phase generated by VCO. If the error signal is zero, VCO produces just its free running frequency ( $\omega_c$ , center frequency). If the error signal is other than zero, then VCO responds by changing its operating frequency.

$$\omega_o(t) = \omega_c + k_o v_f(t), \quad (6)$$

where the constant  $k_o$  is the gain of VCO. After integration of the above equation and substituting into (5), the phase difference is

$$\psi_e(t) = \psi_i(t) - \omega_c t - \int_{-\infty}^t k_o v_f(\tau) d\tau. \quad (7)$$

This can be rearranged as follows:

$$\psi_e(t) = \omega_i t - \omega_c t - \int_{-\infty}^t k_o k_d \sin \psi_e(\tau) d\tau. \quad (8)$$

Differentiating (8) w.r.t. 't' gives

$$\frac{d}{dt} \psi_e(t) = \Delta\omega - K \sin \psi_e(t) \quad (9)$$

where  $\Delta\omega = \omega_i - \omega_c$  and  $K = k_o k_d$  is the gain of PLL. The PLL continues to vary the phase of VCO  $\omega_o$  until locked, i.e. frequency and phase of the input signal are the same as those generated by VCO. After getting locked, PLL follows the changes in frequency and phase of input signal.

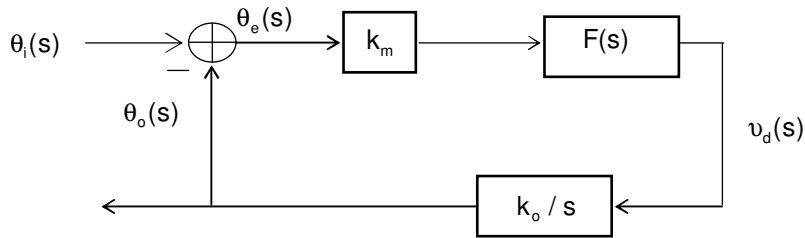
It can be concluded from the above analysis that the phase lock arrangement is described by the non-linear equation (9). Solution of this equation is not known for arbitrary values of  $\Delta\omega$ , and  $K$ . Without an aperiodic solution, the feedback system (PLL) cannot achieve phase stability, i.e.,

output frequency of VCO  $\omega_i$  will never be equal to input frequency  $\omega_o$ . Simplifications are needed to solve the equation [9]. One solution is the linear solution, in which we assume that, for small values of  $\psi_e(t)$

$$\sin \psi_e(t) \approx \psi_e(t) \tag{10}$$

The linear model of PLL in the continuous time-domain (S-domain) is shown in Fig. 2 [10]. Since phase is integral of the angular frequency, the VCO transfer function

$$\frac{\theta_o(s)}{v_i(s)} = \frac{k_o}{s} \tag{11}$$



**FIGURE 2:** Linear model of the PLL.

The transfer function of loop filter is  $F(s)$ . The closed loop transfer function of a PLL is

$$H(s) = \frac{\theta_o(s)}{\theta_i(s)} = \frac{k_m k_o F(s)}{s + k_m k_o F(s)} \tag{12}$$

$F(s)$  will have at least one pole (first order LPF) so the order of PLL always exceeds the order of loop filter by one. If we assume that the loop filter is first order LPF then the PLL will be a second order PLL [11].

A PLL has two main operation modes; Acquisition mode in which PLL is either out of lock or just starts to lock to a signal, the frequency range in which the PLL can acquire the input signal is called capture range. Tracking mode in which PLL is already locked to the input signal and begins to follow the changes in frequency and phase of input signal, the frequency range of PLL in which it can track the input signal is called tracking range. The tracking range is larger than capture range. The loop filter bandwidth is chosen according to the capture range and tracking range of PLL. There is a trade-off between the loop filter bandwidth and the noise in PLL. In case the bandwidth is narrow, the noise level is low but locking time will be long. While larger bandwidth implies large noise level but provides faster locking time [12:14].

**2.1 Drawbacks of PLL**

The performance of PLL in estimating and tracking frequency has some limitations such as

- 1- Double frequency ripples cannot be suppressed easily using first-order LPF, so higher order LPF is needed. The problem is that stability is not guaranteed for higher order PLLs [15].
- 2- All PLLs will exhibit some ringing or settling time. PLLs settling and lock times are primarily a function of the loop filter bandwidth. There is, in general, a trade off relationship between stability/noise immunity and settling/lock time [16].
- 3- The maximum allowable bandwidth of PLL is usually about 10% of the input frequency i.e. the tracking range for PLL is very limited and must be near the center frequency of

VCO, in case of first-order loop filter this range is  $f_{vco} + 0.1 \times f_{vco}$ . As the order of LPF increases tracking range of PLL decreases [17].

### 3. PROPOSED FREQUENCY ESTIMATOR

The proposed estimator consists of three components, i.e. frequency detector, accumulator and quadrature numerically controlled oscillator (NCO). The frequency detector component receives the two input sinusoidal quadrature signals and the two quadrature signals from NCO. Multiplication between these signals is done to obtain two quadrature signals which represent frequency difference between the input frequency and NCO signal frequency. These two sinusoidal signals are passed through a differentiator and squaring circuit to extract the frequency difference from the argument of sinusoidal signal. The final result from the frequency detector is the frequency difference term. This difference is directed to the accumulator which accumulates the differences until no difference is founded between input and generated signal frequencies. At this point the output of the accumulator represents input radian frequency and the generated signal from NCO runs at the same input frequency.

We first discuss the design of a digital receiver which can help in understanding our proposed method. A typical digital receiver structure implemented in software defined radio (SDR), except the radio frequency (RF) front-end, is shown in Fig. 3 [18]. The analog RF signal is received and fed through a low noise amplifier (LNA) to a mixer to convert RF to intermediate frequency. Signal is then passed to an analog to digital converter (ADC). A digital down-converter (DDC) converts ADC output to digitized real signal centered at zero frequency. DDC consists of direct digital synthesizer (DDS), low pass filter (LPF) and decimator. In addition to lowering frequency, DDC decimates the signal to a lower sampling rate allowing application of lower speed processors.

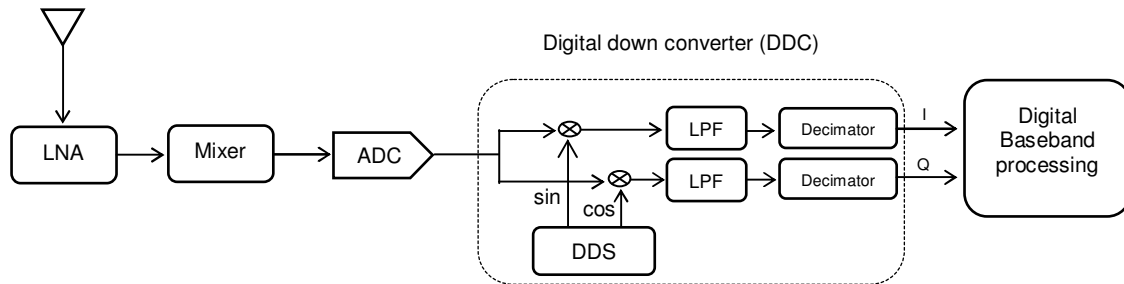
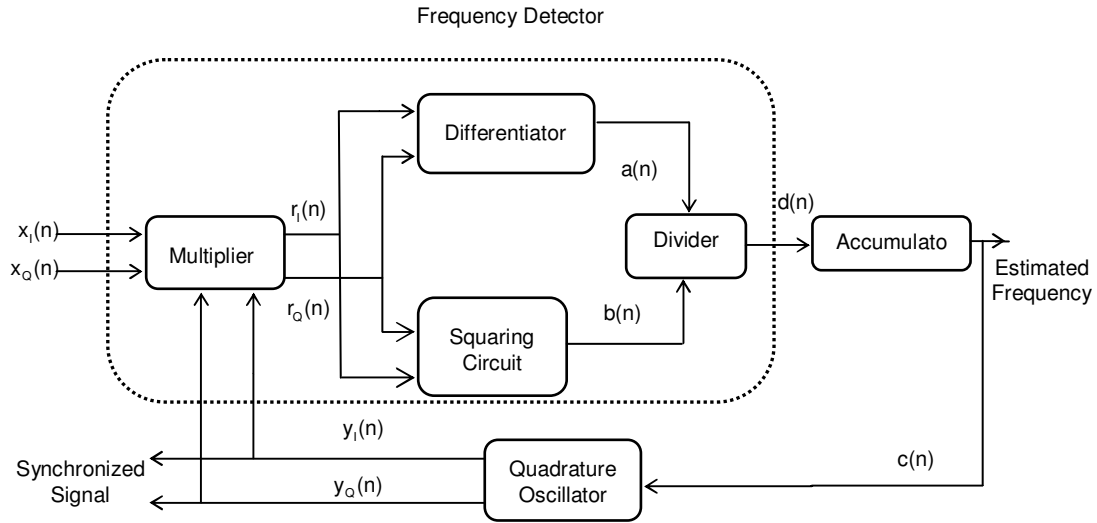


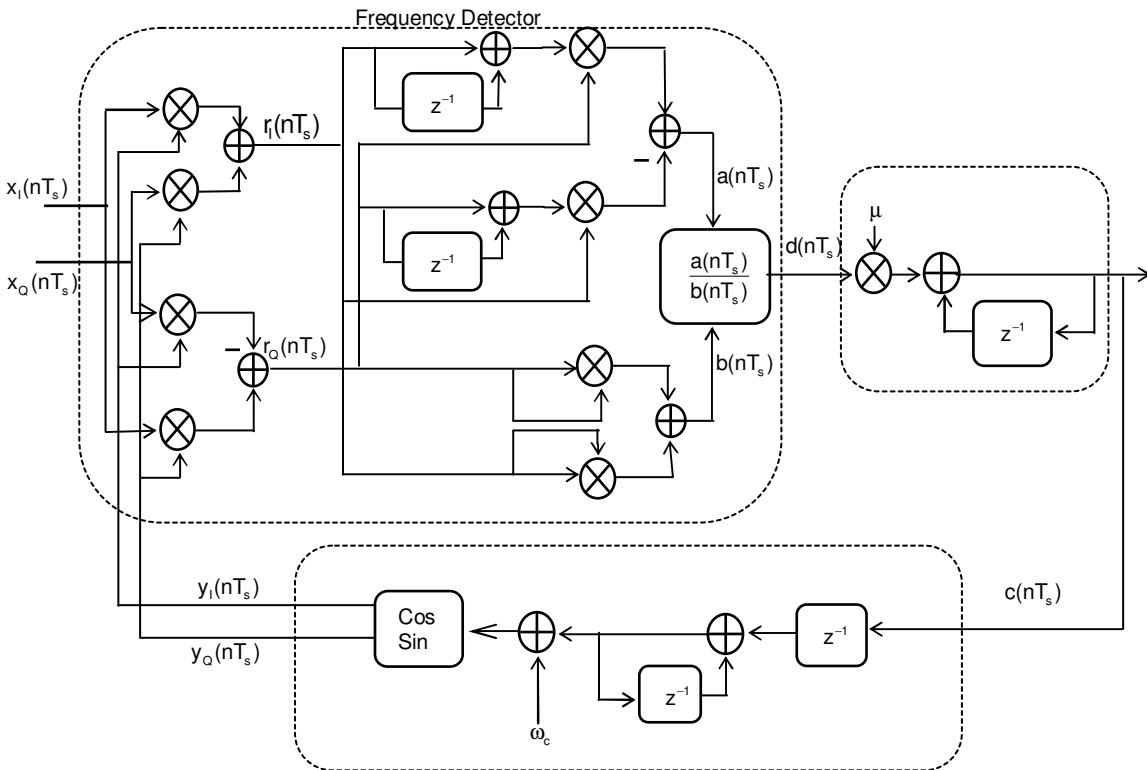
FIGURE 3: Digital receiver structure.

A block diagram of the proposed frequency estimator is given in Fig. 4. It is very similar to a PLL structure except that the phase detector and loop filter are replaced with a frequency detector and an accumulator respectively. The estimator receives the quadrature input signal from DDC, which is then sent to the frequency detector. Frequency detector consists of three components i.e., multiplier, differentiator and squaring circuit.



**FIGURE 4:** Block diagram of proposed estimator.

The block diagram of Fig. 4 can be implemented in analog or digital domain. Since the proposed estimator is designed mainly for digital receivers, description in the discrete time domain will be more accurate. Fig. 5 explains the detailed discrete time implementation of the proposed estimator structure. Mathematical and working details are explained below:



**FIGURE. 5** Discrete time model of proposed estimator.

### 3.1 Frequency and Power Estimation

We assume the input quadrature signal and the quadrature signal generated by the oscillator are

$$x_I(nT_s) = \sqrt{2p(nT_s)} \cos(\omega_i nT_s + \theta_i). \quad (13)$$

$$x_Q(nT_s) = \sqrt{2p(nT_s)} \sin(\omega_i nT_s + \theta_i). \quad (14)$$

$$y_I(nT_s) = \cos(\omega_o nT_s + \theta_o). \quad (15)$$

$$y_Q(nT_s) = \sin(\omega_o nT_s + \theta_o). \quad (16)$$

Where  $p(nT_s)$  is the power of input signal.  $\omega_i, \omega_o, \theta_i$  and  $\theta_o$  are the input and output radian frequencies and phases respectively,  $T_s$  is the sampling time.

#### A. Frequency detector

##### A.1. Multiplier

It multiplies input quadrature signals with quadrature signals of oscillator. This results in two quadrature signals ( $r_I(n), r_Q(n)$ ) which represent the difference between input frequency and generated frequency of oscillator.

$$r_I(nT_s) = x_I(nT_s)y_I(nT_s) + x_Q(nT_s)y_Q(nT_s) = \sqrt{2p(nT_s)} \cos(\Delta\omega nT_s + \Delta\theta), \quad (17)$$

$$r_Q(nT_s) = x_I(nT_s)y_Q(nT_s) - x_Q(nT_s)y_I(nT_s) = -\sqrt{2p(nT_s)} \sin(\Delta\omega nT_s + \Delta\theta). \quad (18)$$

Where  $\Delta\omega = \omega_i - \omega_o, \Delta\theta = \theta_i - \theta_o$ .

##### A.2. Differentiator

The quadrature signals ( $r_I(n), r_Q(n)$ ) pass through a differentiator circuit. This circuit is used to extract frequency difference from the argument of ( $r_I(n), r_Q(n)$ ). The output will be the quadrature signals multiplied by the frequency difference and amplitude of both signals.

$$\begin{aligned} \frac{d}{dt}r_I(t) \Big|_{t=nT_s} &\approx \{r_I(nT_s) - r_I(n-1)T_s\} \times \frac{1}{T_s} \\ &= -\sqrt{2p(nT_s)}\Delta\omega \sin(\Delta\omega nT_s + \Delta\theta) + \left(\frac{d}{dt}\sqrt{2p(t)}\right) \Big|_{t=nT_s} \times \cos(\Delta\omega nT_s + \Delta\theta), \end{aligned} \quad (19)$$

$$\begin{aligned} \frac{d}{dt}r_Q(t) \Big|_{t=nT_s} &\approx \{r_Q(nT_s) - r_Q(n-1)T_s\} \times \frac{1}{T_s} \\ &= -\sqrt{2p(nT_s)}\Delta\omega \cos(\Delta\omega nT_s + \Delta\theta) - \left(\frac{d}{dt}\sqrt{2p(t)}\right) \Big|_{t=nT_s} \times \sin(\Delta\omega nT_s + \Delta\theta). \end{aligned} \quad (20)$$

$$\begin{aligned} a(nT_s) &= r_Q(nT_s) \times \frac{d}{dt}r_I(t) \Big|_{t=nT_s} - r_I(nT_s) \times \frac{d}{dt}r_Q(t) \Big|_{t=nT_s} \\ &= 2p(nT_s)\Delta\omega [\sin^2(\Delta\omega nT_s + \Delta\theta) + \cos^2(\Delta\omega nT_s)] \\ &= 2p(nT_s)\Delta\omega \end{aligned} \quad (21)$$

### A.3. Squaring circuit

The quadrature signal  $(r_i(nT_s), r_Q(nT_s))$  also passes through squaring circuit. Squaring circuit and divider eliminate the amplitude term from the result of differentiator. The output of squaring circuit can be considered as power of input signal.

$$\begin{aligned} b(nT_s) &= r_i^2(nT_s) + r_Q^2(nT_s) \\ &= 2p(nT_s)[\sin^2(\Delta\omega nT_s + \Delta\theta) + \cos^2(\Delta\omega nT_s)] \\ &= 2p(nT_s). \end{aligned} \tag{22}$$

### A.4. Divider

The divider is used to eliminate the amplitude part from (5). Its output is

$$d(nT_s) = \frac{a(nT_s)}{b(nT_s)} = \Delta\omega. \tag{23}$$

### B. Accumulator

Frequency difference is passed to the accumulator which accumulates this difference until saturation which means the input frequency is same as the generated frequency of oscillator. The accumulator action is given by the following equation

$$\mu d(nT_s) = c(nT_s) - c((n-1)T_s). \tag{24}$$

Where  $\mu$  is a variable whose value determines the speed of locking and stability of the system.

### C. Quadrature oscillator

The oscillator considered in this paper is numerically controlled oscillator (NCO). Output of the accumulator controls the value of generated frequency of NCO. The feedback loop of the estimator continues until there is no difference between the input and generated frequency. At this point, output of the accumulator represents the estimated input radian frequency. NCO equation is

$$\omega_o(nT_s) = \omega_c + \sum_{k=0}^{n-1} c(kT_s). \tag{25}$$

Where  $\omega_c$  is the center frequency. When the generated frequency from NCO is equal to the input frequency, the accumulator saturates. Thus at  $\omega_i(nT_s) = \omega_o(nT_s)$

$$c(nT_s) = \omega_i T_s = 2\pi f_i T_s \tag{26}$$

## 3.2 Stability

Stability of the estimator is analyzed in this sub-section. Linear model (z-model) of the system is built to get system's transfer function, which is the ratio between output and input signals. Fig. 6 shows z-model of the system. It may be noted that NCO equivalent model is only a delay circuit. The input is frequency and the output is also frequency so no integration is needed unlike PLL. Following equations are obtained from Fig. 6.



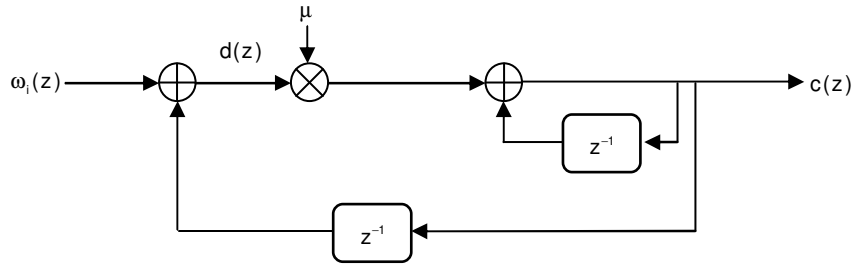


FIGURE. 6 Z-model of proposed estimator.

$$c(z) = \omega_i(z) - z^{-1}c(z), \quad (27)$$

$$\mu d(z) = c(z)(1 - z^{-1}), \quad (28)$$

$$\omega_i(z) = c(z)\left(z^{-1} + \frac{1 - z^{-1}}{\mu}\right). \quad (29)$$

Therefore, transfer function of the proposed estimator

$$T(z) = \frac{c(z)}{\omega_i(z)} = \frac{\mu z}{z - (1 - \mu)}. \quad (30)$$

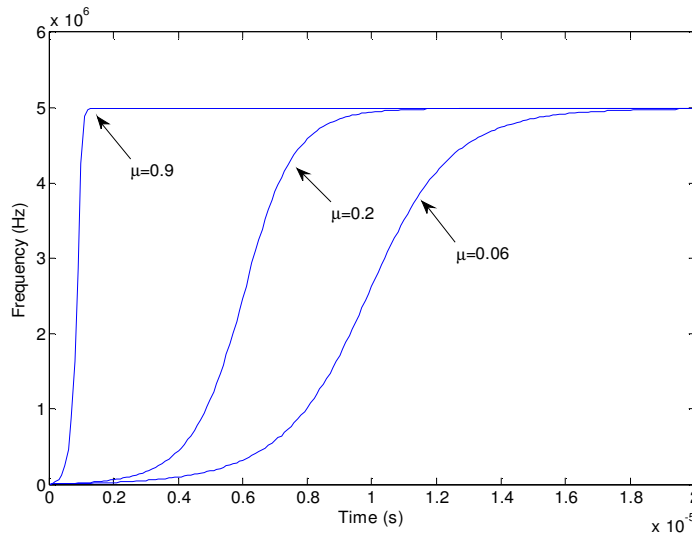
The variable  $\mu$  controls stability of the system and its range must be  $0 \geq \mu < 1$ . For small values of  $\mu$  the system is more stable but takes long settle time, for greater values of  $\mu$  settling time decreases.

#### 4. COMPUTER SIMULATIONS

Validity of the above mentioned mathematical analysis is checked by performing computer simulations. In this section we also present a comparison between conventional PLL and proposed estimator. The results of these simulations are discussed in the sub-sections below.

##### 4.1 Frequency Estimation

We first perform simulations for frequency estimation. The proposed system input frequency is set at  $f_i = 500$  kHz, sampling frequency  $f_s = 1/T_s = 10$  MHz with different values of  $\mu$ . Fig. 7 shows the output of the estimator for each value of  $\mu$ . It is clear that for small values of  $\mu$  frequency locking takes more time compared to that when  $\mu$  is large. Stability is guaranteed as  $\mu$  is in the specified range.

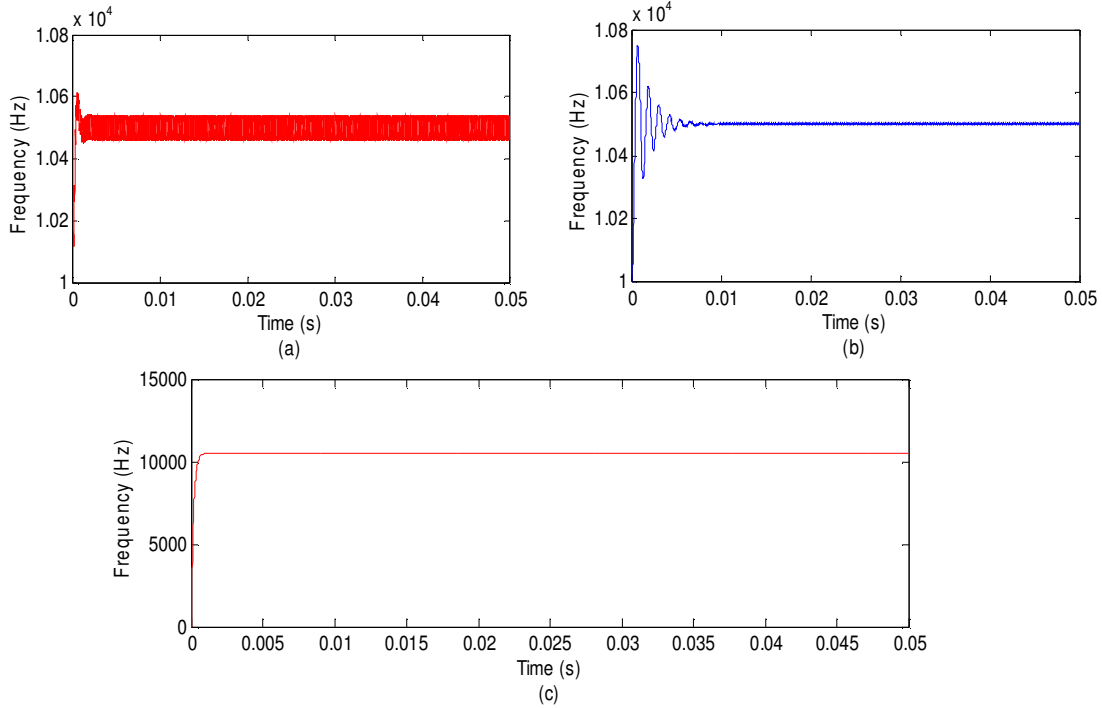


**FIGURE. 7** Frequency estimation with different  $\mu$  .

Following simulations provide a frequency estimation comparison of proposed and conventional PLL. An input signal with frequency of  $f_i = 10$  kHz is applied. In these simulations we used two different loop filters with the PLL; the first loop filter is first order LPF with cut-off frequency 1000 Hz, the other is second order LPF with the same cut-off frequency. The parameters of PLL are:  $k_m = 1/2$ ,  $k_o = 1024$ ,  $f_{in} = 10.5$  kHz,  $f_{nco} = 10$  kHz,  $f_{cut} = 1000$  Hz,  $f_s = 100$  kHz . While the parameters of proposed estimator are  $f_{nco} = 100$  Hz,  $f_s = 100$  kHz .

In Fig. 8 (a) conventional PLL with first-order LPF (second order PLL) did not suppress the double frequency ripple completely (variance is 665.3610), and overshoot (maximum overshoot reaches 1.06 KHz) occurs, settling time (the time required to reach correct estimated frequency is about 1 ms) [19]. In Fig. 8 (b) conventional PLL with second-order PLL (third order PLL) the ripple is suppressed better than the first-order LPF (variance is 1.2759), overshoots (maximum overshoot reaches 1.075 KHz) and ringing occur, settling time (increased to 7.5 ms seconds), estimation range is reduced and stability becomes critical.

In Fig. 8 (c) the proposed estimator estimates the input frequency at 0.5 ms with no ripples (Although no loop filter is used system variance is  $1.0839e-18$ ), the estimation range is wide from 0 to half the sampling frequency and more stable than PLL. The previous comparison clearly shows that the proposed estimator is faster, more stable, has a wide range than conventional PLL. The variance of estimations is also given in Table 1. It can be seen that estimation's variance is extremely low in case of the proposed method showing its superiority.



**FIGURE. 8** Frequency estimation curves of (a) Conventional PLL with first-order loop filter, (b) Conventional PLL with second-order loop filter, (c) Proposed Estimator.

Architecture	Variance	Settling time	Max. overshoots	Tracking range
Second order PLL	665.3610	1 ms	1.06 kHz	8950:11050 HZ
Third order PLL	1.2759	7.5 ms	1.075 KHz	9095:10904 Hz
Proposed estimator	1.0839E-18	0.5 ms	-	0:50 kHz

**TABLE 1:** Comparison between PLL and proposed estimator.

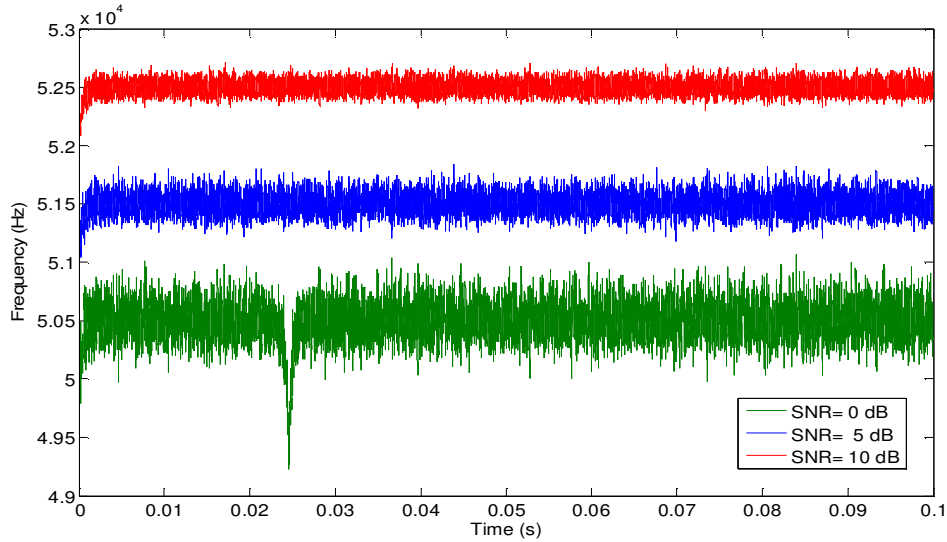
#### 4.2 Frequency Estimation With AWGN

In practical situations the received input signal is corrupted with AWGN and this will affect the results. Some design modifications are performed to take care of this situation. Block diagram of the modified design is shown in Fig. 15. Two moving average filters (MAF) with length of 10 stages are used for I and Q phase input signals. Another filter is used for smoothing the output. The MAF is a low pass finite impulse response (FIR) filter commonly used for smoothing an array of sampled data/signal. It takes M samples of input at a time and computes the average of those samples and produces a single output point. As the filter length (number of delays) increases the smoothness of output increases [20].

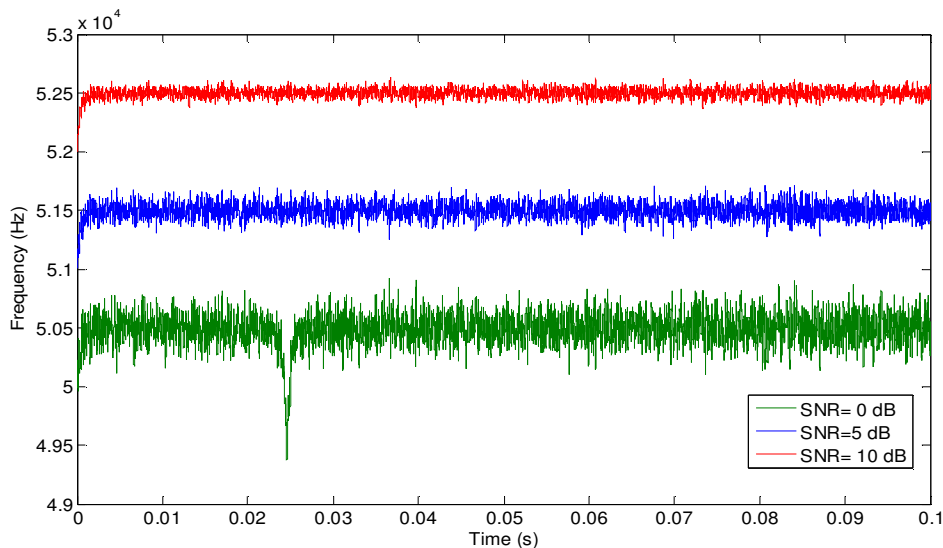
The following simulations are performed to investigate the effect of AWGN on both conventional DPLL (second and third order DPLL) and proposed estimator. MAF with the same lengths are added to both architectures. An input signal is applied to three architectures with different frequencies 5.05E4, 5.15E4, 5.25E4 Hz and SNR 0, 5, 10 dB at sampling frequency of 10 MHz.

Fig. 9 shows estimations of the second order DPLL with different frequencies and SNRs. At frequency of  $5.05E4$  and SNR =  $0$  dB (green line) the noise level affected the estimation and variance is  $2.2728E4$ . At frequency of  $5.15E4$  and SNR=  $5$  dB (blue line) the noise level is reduced and variance is  $6.6063E3$ . At frequency of  $5.25E4$  and SNR= $10$  dB (red line) a higher reduction in noise level causes the variance to further reduce to  $3.1311E3$ .

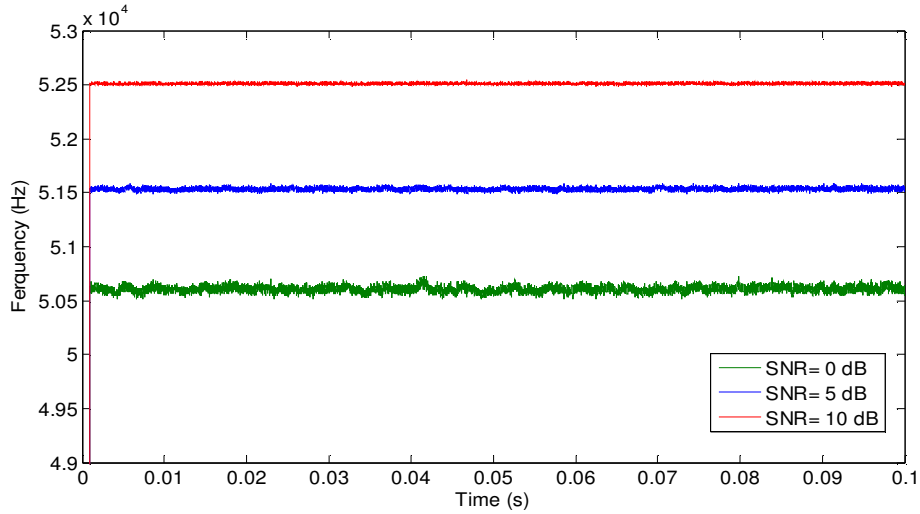
The estimations are improved using third order DPLL as shown in Fig. 10. The proposed estimator results are shown in Fig. 11. Variance of estimates of each architecture is summarized in Table. 2 it can be seen that, the proposed architecture produces the smallest values for each estimated frequency although loop filter is not used.



**FIGURE. 9** Frequency estimation with different SNRs using second order DPLL.



**FIGURE. 10** Frequency estimation with different SNRs using third order DPLL.



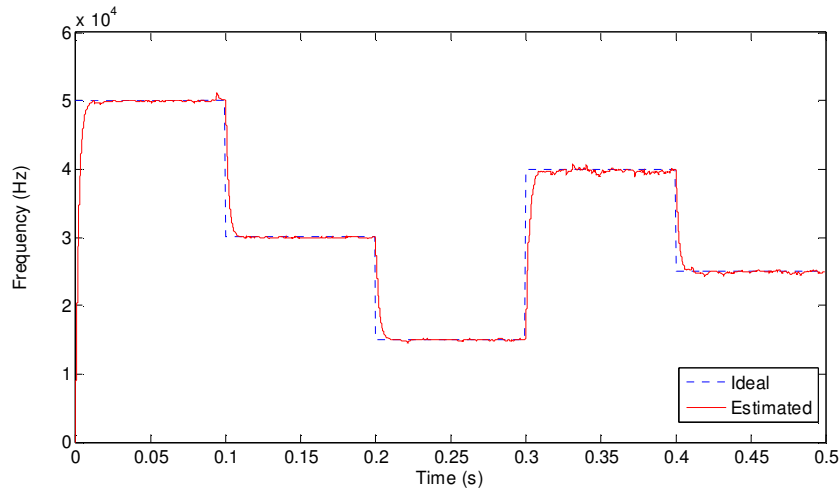
**FIGURE. 11** Frequency estimation with different SNRs using proposed estimator.

Architecture	SNR	Variance
Second order PLL	0 dB	2.2728E4
	5 dB	6.6063E3
	10 dB	3.1311E3
Third order PLL	0 dB	1.7288E4
	5 dB	4.1194E3
	10 dB	1.6340E3
Proposed estimator	0 dB	3.4036E-4
	5 dB	3.5282E-4
	10 dB	3.6620E-4

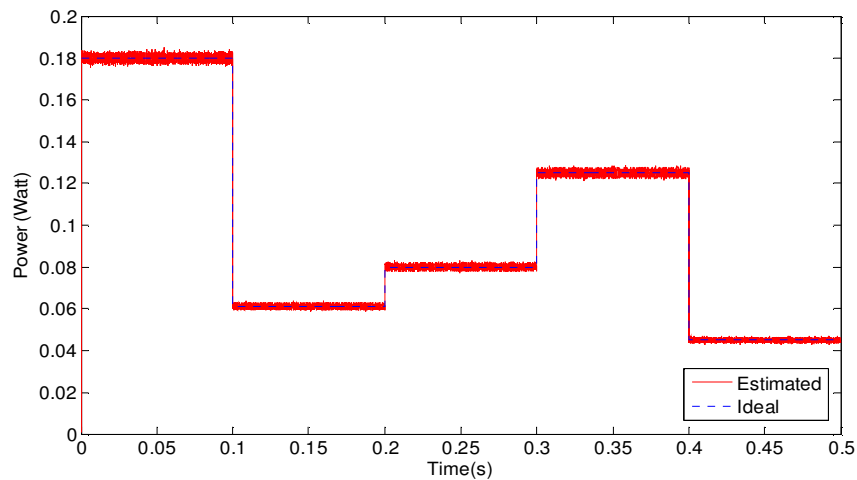
**TABLE 2:** Comparison between PLL and proposed estimator with different SNR.

### 4.3 Tracking Performance

We also checked performance of the proposed method in tracking frequency changes and estimating amplitude/power variations. Frequency of the input sinusoidal signal is varied randomly within the range 10 kHz-60 kHz. The input signal power is also randomly varied within the range 0.06125-0.21125 watt. The simulations are performed with SNR 10 dB and sampling frequency of 1 MHz. Fig. 12 shows the tracking of the input frequency while Fig. 13 gives estimation of power. It can be seen that the estimator tracks changes in input frequency and amplitude very fast.

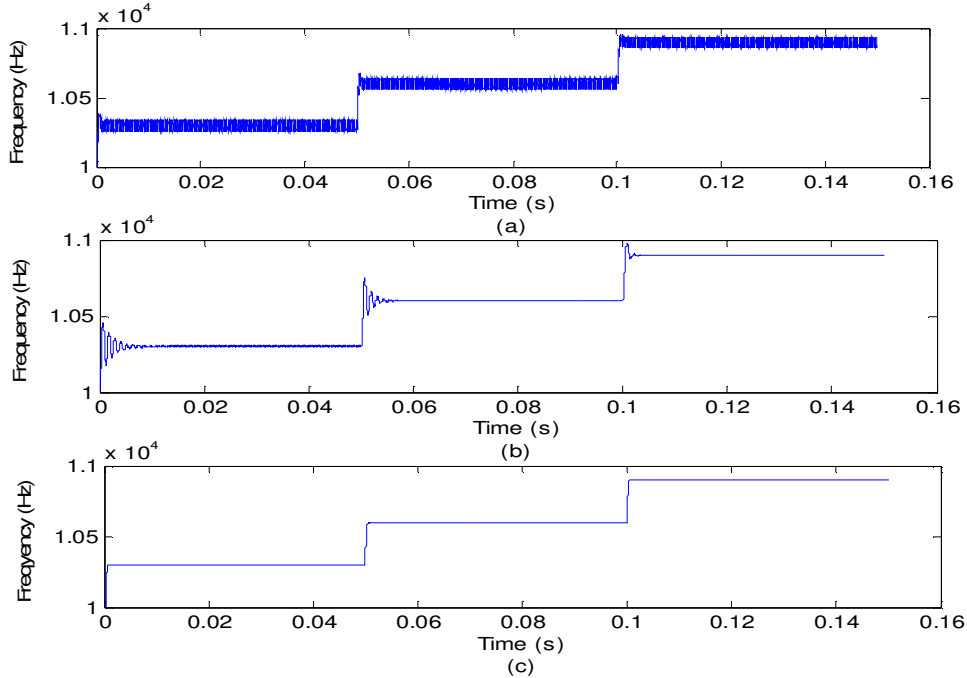


**FIGURE. 12** Frequency tracking.



**FIGURE. 13** Power estimation.

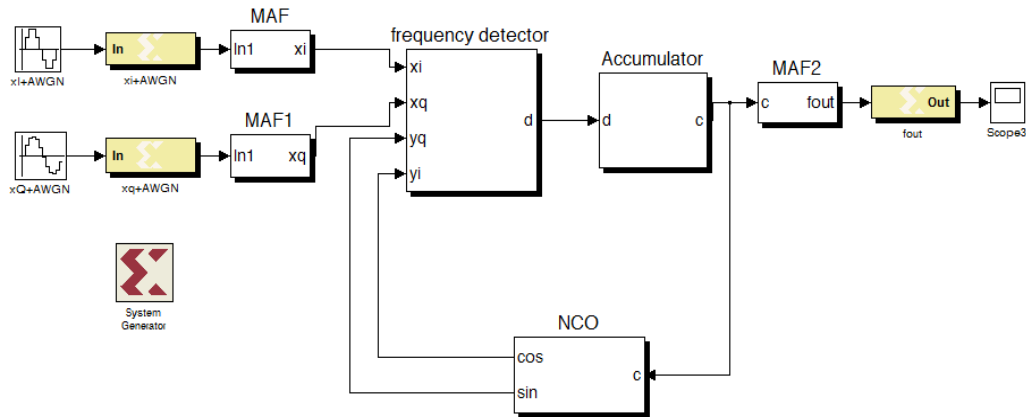
In order to investigate the performance of conventional PLL to track the frequency change, we have to change the previous estimation range because it exceeds PLL capabilities. A signal with input frequency changed from 10.3-10.9 kHz with sampling frequency of 100 kHz without AWGN is applied to conventional PLL with second-order LPF with  $f_{nco} = 10$  kHz. As shown in Fig. 14 for each tracked frequency the overshooting and ringing occurred. The situation becomes worst when the input signal have AWGN. As a result of using second-order LPF the estimation range is reduced to 10:10.9 kHz.



**FIGURE. 14** Frequency tracking using [1] Conventional second order DPLL. b) Conventional third order DPLL. c) Proposed estimator.

## 5. FPGA IMPLEMENTATION

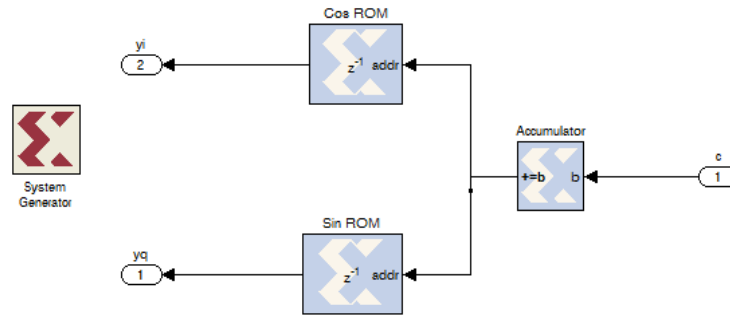
The estimator is modeled using Xilinx system generator tool [21,22]. VHDL code is used to describe the proposed estimator. Fig. 15 shows the system model used to generate the VHDL code [23, 24]. All signals of the estimator model are fixed point signals with 16 bits.



**FIGURE. 15.** System generator model of proposed estimator.

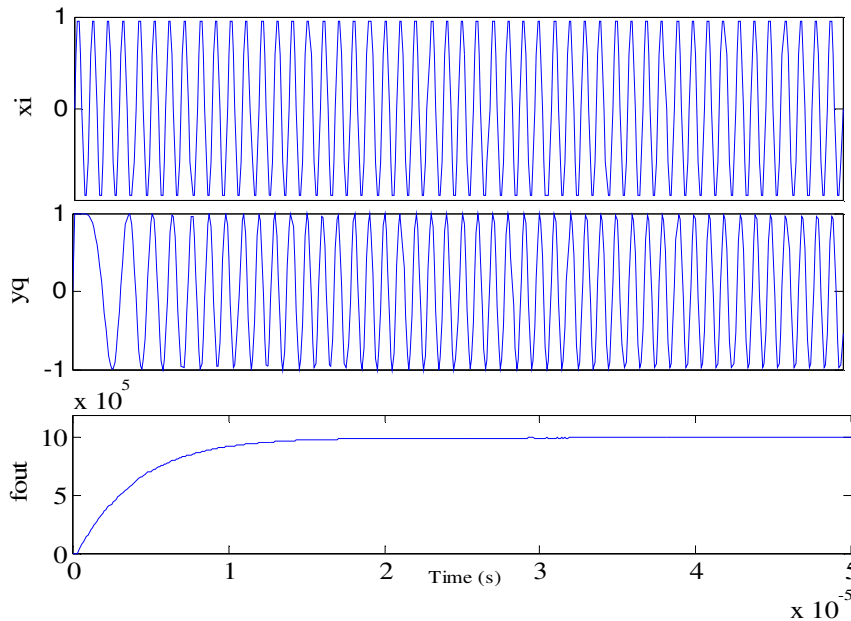
The model receives quadrature input signal with AWGN then the signals are passed to the moving average filter (MAF) to generate signals named 'xi' and 'xq'. The generated quadrature signals from NCO are 'yi' and 'yq'. These two quadrature signals are directed to the frequency detector block whose output is the signal 'd'. This signal enters the accumulator block to generate

the signal 'c'. Another MAF is used to enhance the output signal 'fout'. The model of quadrature NCO as shown in Fig. 16 consists of accumulator and two read only memory (ROM), one ROM stores sine waveform samples and the other stores cosine waveform samples. The accumulator receives the signal 'c' and generates both signals 'yi' and 'yq'. One rotation of the accumulator represents one complete cycle of sine or cosine waveform.



**FIGURE. 16.** System generator model of NCO.

The estimator was implemented using XCSD1800A-4FG676C Spartan-3A DSP board. Fig. 17 shows simulation results of the VHDL model when input and sampling frequencies are 1 MHz and 10 MHz respectively. It can be seen that the hardware model results agree with the simulations in locking time and stability. FPGA resources utilization for proposed architecture, which indicates how many hardware components are used by the model, is provided in Table 3. It can be seen that the proposed method requires fewer hardware resources. Implementation results indicate that the proposed design provides fast operation and low power consumption. Furthermore, it can be observed that the method has a simple design and avoids complexities.



**FIGURE. 17.** VHDL simulation results.



Component	used	Avilable	Utilization
Number of slice flip flops	76	33280	1%
Value Number of 4 input LUTs	583	33280	1%
Number of occupied slices	320	16640	1%
Number of DSP48As	6	84	7%
Number of Bonded IOBs	81	519	15%
Maximum frequency	225 MHz		
Power consumption	127 mW		

**TABLE 3:** FPGA resources utilization.

## 6. CONCLUSIONS

PLL is wide used in communication application such as frequency tracking. Conventional PLL has many drawbacks such as overshoot, ringing, limited tracking range and instability. The aim of this paper is to present a proposed estimator solve limitation in conventional PLL. The proposed estimator accurately estimates unknown frequency and power of input sinusoidal signal simultaneously. Proposed method worked well even in the presence of AWGN. The estimator rapidly tracked any changes in input signal frequency and amplitude. Operation of the proposed structure is similar to that of a PLL. The phase detector component and the low pass filter in PLL have been replaced with frequency detector and accumulator. Computer simulations performed showed promising results. The paper also discussed hardware implementation of the system. The proposed architecture is designed and modeled using VHDL and implemented using FPGA circuit. Implementation results indicate that the estimator has a simple design, faster operation and low power consumption.

## 7. ACKNOWLEDGEMENT

This research is partially supported by Grant-in-Aid for scientific Research (B) no.20360174, and the Aihara Project, the FIRST program from JSPS, initiated by CSTP.

## 8. REFERENCES

- [1] B. Sklar. Digital Communications: Fundamentals and Applications. Prentice Hall Inc.,2001.
- [2] J. G. Proakis, M. Salehi. Digital Communications. Mcgraw-Hill., 2008.
- [3] D. Green. "Lock-In, Tracking, and acquisition of AGC-aided phase-locked loops." IEEE Transaction on Ciruits and Systems, vol. 32, pp. 559-568,1985.
- [4] S. Changhong, C. Zhongze, Z. Lijun, L. Yong. "Design and Implementation of Bandwidth Adaptable Third-order All digital Phase-Locked Loops," in Proc. WICOM, 2010, pp.1-4.
- [5] P. Hanumolu. "W Gu-Yeon, M. un-ku, A Wide- Tracking Range clock and Data recovery Circuit." IEEE Transaction on Solid state circuits, vol. 43, pp. 425:439, 2008.
- [6] X. hongbing, G. Peiyuan, L. Shiyi. "Modeling and simulation of Higher-order PLL rotation tracking system in GNSS receiver," in Proc. ICMA, 2009, pp. 1128-1133.
- [7] A. Carlosena, A. Manuel- Lazaro. "Design of Higher-Order Phase-Lock Loops." IEEE Transaction on Circuits and systems II, vol. 54, pp. 9-13, 2007.
- [8] B. Razavi, Monolithic Phase-Locked Loops and clock recovery circuits: Theory and Design, Wiley-IEEE Press, 1996.

- [9] V.F. Kroupa. Phase Lock Loops and Frequency Synthesis. John Wiley & Sons, 2003.
- [10] F.M Gardner. Phase lock techniques. Wiley-Interscience, 2005.
- [11] R. E. Best. Phase locked loops Design, simulation, and Application. Mcgraw-Hill, 2007.
- [12] L. Kyoohyun, C. Seunghee, K. Bemsup. "Optimal loop bandwidth design for low noise PLL applications," in Proc. ASP-DAC,1997, pp. 425-428.
- [13] W. Ping-Ying, F. Chia-Huang. "All digital modulation bandwidth extension technique for narrow bandwidth analog fractional-N PLL," in Proc. ESSCIRC,2010, pp. 270-273.
- [14] J.Roche, W. Rahadjandrabey, L. Zady, G. Bracmard, D. Fronte. "A PLL with loop bandwidth enhancement for low-noise and fast-settling clock recovery," in Proc. ICECS, 2008, pp. 802-805.
- [15] M. Saber, Y. Jitsumatsu, M.T.A. Khan. "Design and Implementation of Low Ripple Low Power Digital Phase-Locked Loop." Signal Processing: An international journal (SPIJ), vol. 4, pp. 304-317, Feb. 2011.
- [16] M. Min, V. Mannama, T. Paavle."Settling time minimization in PLL frequency synthesizer," in Proc. ICCSC, 2005, pp. 366-369.
- [17] L. Verrazzani . "Pull-In Time and Range of Any Order Generalized PLL. " IEEE Transaction on Aerospace and Electronic systems, vol. AES14, pp.329-333, 2007.
- [18] M. Dillinger, K. Madani, N. Alonistioti. Software Defined radio: Architectures, Systems, and Functions. Wiley, 2003.
- [19] M. J.Roberts, R.A. Riccardo. Student's Guide to Analysis of Variance. Routledge, 1999.
- [20] W.S. Steven. The Scientist & Engineer's Guide to Digital Signal Processing. California Technical Pub,1997.
- [21] P. Pong Chu. RTL Hardware Design Using VHDL: Coding for Efficiency, Portability and Scalability. Wiley-IEEE Press, 2006.
- [22] Xilinx Inc. system generator for DSP user guide. Xilinx, 2009.
- [23] W.Y. Yang. Matlab/Simulink for digital communication. A-Jin, 2009.
- [24] Pong p. chu. FPGA Prototyping by VHDL Examples: Xilinx Spartan-3 Version. Wiley-Interscience, 2008.