A Simple Design to Mitigate Problems of Conventional Digital Phase Locked Loop

M. Saber *mohsaber@kairo.csce.kyushu-u.ac.jp*
Department of Informatics
Kyushu University
744 Motooka, Nishi-ku, Fukuoka-shi, 89-0395, Japan

Y. Jitsumatsu *jitsumatsu@inf.kyushu-u.ac.jp*
Department of Informatics
Kyushu University
744 Motooka, Nishi-ku, Fukuoka-shi, 89-0395, Japan

M. T. A. Khan *tahir@apu.ac.jp*
Ritsumeikan Asia Pacific University, College of Asia pacific Studies
1-1 Jumonjibaru, Beppu, Oita, 874-8577, Japan

Abstract

This paper presents a method which can estimate frequency, phase and power of received signal corrupted with additive white Gaussian noise (AWGN) in large frequency offset environment. Proposed method consists of two loops, each loop is similar to a phase–locked loop (PLL) structure. The proposed structure solves the problems of conventional PLL such as limited estimation range, long settling time, overshoot, high frequency ripples and instability. Traditional inability of PLL to synchronize signals with large frequency offset is also removed in this method. Furthermore, proposed architecture along with providing stability, ensures fast tracking of any changes in input frequency. Proposed method is also implemented using field programmable gate array (FPGA), it consumes 201 mW and works at 197 MHz.

Keywords: Digital Phase-Locked Loop (DPLL), Field Programmable Gate Array (FPGA).

1. INTRODUCTION

In communication, the received signal has to be synchronized first before being demodulated. Synchronization includes carrier recovery and timing recovery. Carrier recovery is the estimation and compensation of carrier frequency and phase. Carrier frequency offset occurs in communication systems due to different reasons such as frequency mismatch between oscillators in transmitter and receiver, Doppler shift as a result of motion between transmitter and receiver and the channel noise. The estimation of frequency and phase in the presence of noise is a critical issue in signal processing applications [1: 3]. PLL is used in synchronization of frequency and phase of received signal. A PLL is a feedback control circuit; it operates by trying to lock the phase and frequency of input signal through the use of its negative feedback. Conventional PLL consists of three fundamental blocks namely a phase detector (usually multiplier), loop filter which is a low pass filter (LPF) and a voltage controlled oscillator (VCO) as shown in Fig. 1 [4, 5].

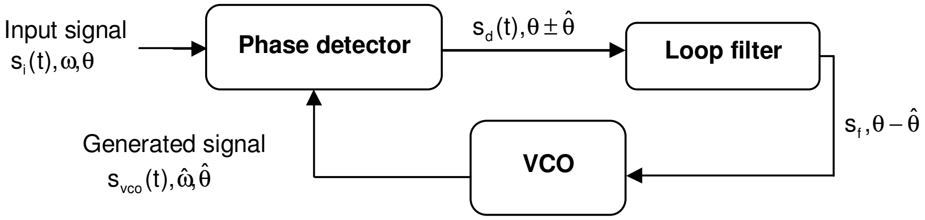
Input signal
$s_i(t), \omega, \theta$
Phase detector
$s_d(t), \theta \pm \hat{\theta}$
Loop filter
$s_f, \theta - \hat{\theta}$
VCO
Generated signal
$s_{vco}(t), \hat{\omega}, \hat{\theta}$

FIGURE 1: Phase Locked Loop.

Phase detector compares the phase of output signal to the phase of input signal. If there is a phase difference between the two signals, it generates an output voltage, which is proportional to the phase error of two signals. This output voltage passes through the loop filter and then as an input to the VCO. Due to this self-correcting technique, the output signal will be in phase with the input signal. When both signals are synchronized the PLL is said to be in lock condition. The bandwidth of a PLL depends on the characteristics of the phase detector, LPF and VCO [6:8]. The performance of PLL in estimating and tracking frequency has some limitations such as:

1) Higher order of LPF is needed to remove high frequency ripples inside the loop. Unfortunately this leads to unstable PLL [9: 11].

2) All PLLs will have ringing or settling time [12, 13].

3) The tracking range of PLL is very narrow and must be near the center frequency of VCO. As the order of LPF increases tracking range of PLL decreases [14, 15].

We propose a design which can not only estimate frequency and phase of the input signal but can also provide estimate of power of the received signal in the presence of background noise. Proposed structure consists of two loops, the first loop has the ability to estimate frequencies up to half the sampling frequency. The second loop is a low power digital phase-locked loop (DPLL). Using the frequency estimator's information, the second loop estimates phase of the received signal. The estimator is modeled with VHDL and implemented using FPGA. Proposed structure's hardware implementation consumes 201 mW and works at a frequency of 197 MHz.

2. PROPOSED ESTIMATOR

The main idea of proposed structure is to first estimate the large frequency offset using frequency estimator which has the ability to estimate frequencies with large range from zero to half sampling frequency with high accuracy. The second step is to use the output of frequency estimator to control a phase estimator which is a low noise PLL. Proposed architecture is shown in Fig. 2 and in Fig. 3 in which the frequency estimator block receives quadrature signal; the output of this block is the estimated power and the estimated frequency. Estimated frequency is then used in phase estimator block as the center frequency of VCO. In this manner PLL, will compensate only the difference in phase between the input signal and generated signal of VCO not in both phase and frequency. In the next subsections, we will provide a detailed explanation of the operation of each block in the proposed architecture.

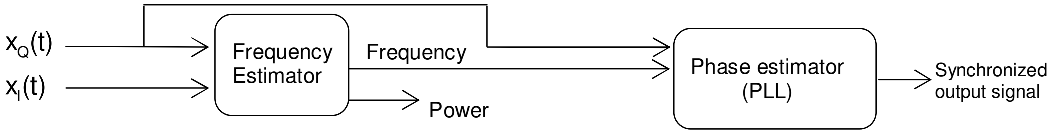

FIGURE 2: Proposed architecture.

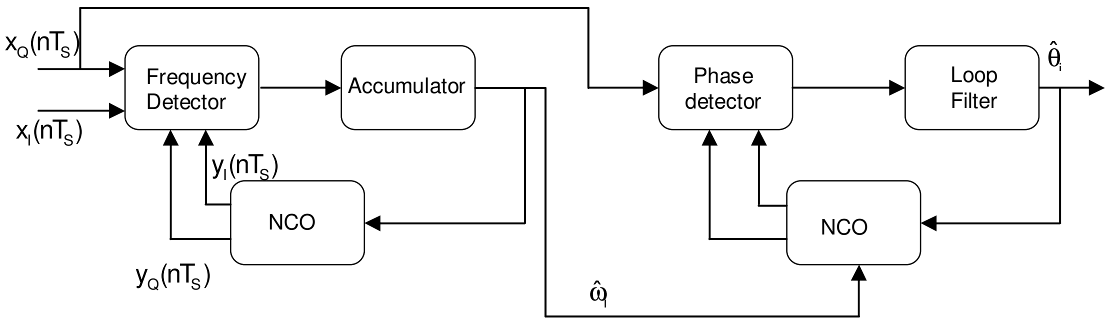

FIGURE 3: Operation of proposed architecture.

A. Frequency Estimator

Block diagram of the suggested frequency estimator is given in Fig. 4. It is like a PLL structure except that the phase detector is changed to frequency detector, LPF is changed to an accumulator and VCO is changed to quadrature numerically controlled oscillator (NCO) [16]. The quadrature input signal enters the frequency detector component which consists of multipliers, adders, subtractors, differentiator and squaring circuit. Fig. 4 explains discrete time implementation of the proposed estimator structure. We would also provide here the mathematical details of frequency and power estimation and stability of proposed architecture. Suppose the input quadrature signal and the quadrature signal generated by the oscillator are

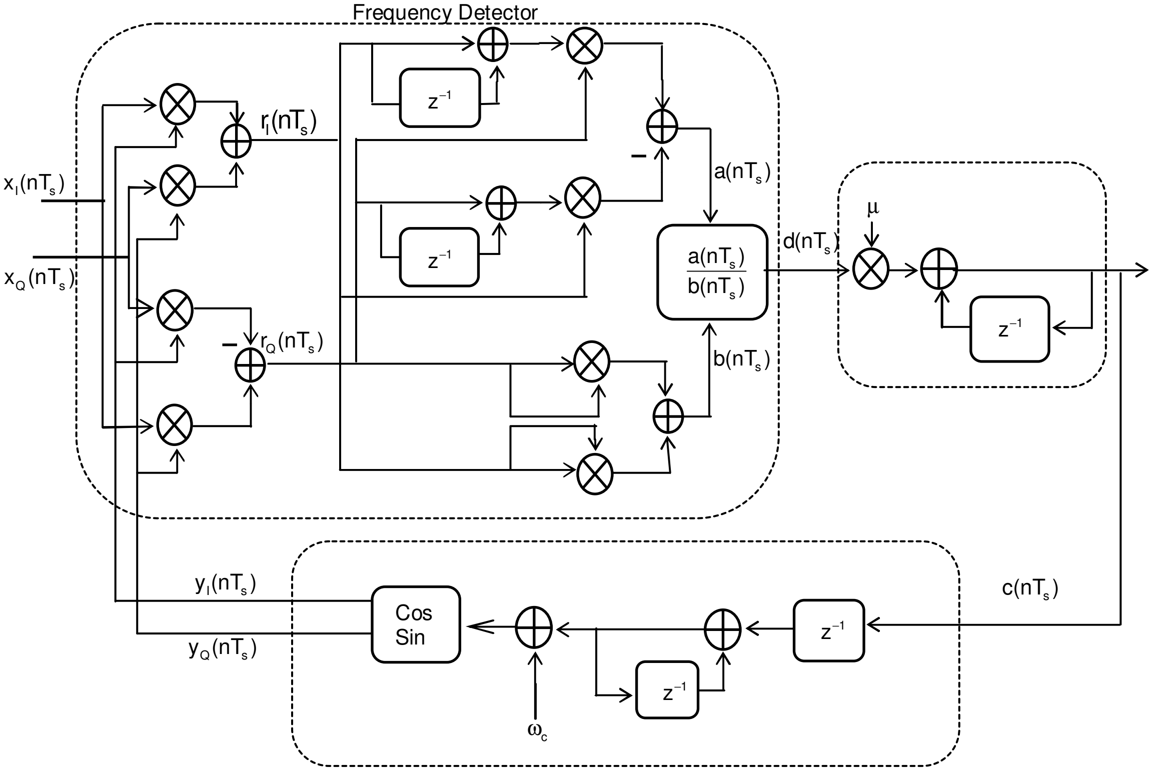

FIGURE 4: Discrete time model of proposed estimator.

1) Frequency and power estimation

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

$$x_I(nT_s) = \sqrt{2p\ (nT_s)}\cos(\omega_i\ nT_s + \theta_i). \tag{1}$$

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

$$y_I(nT_s) = \cos(\omega_o\ nT_s + \theta_o). \tag{3}$$

$$y_Q(nT_s) = \sin(\omega_o nT_s + \theta_o). \tag{4}$$

Where $p(nT_s)$ is the power of input signal. ω_i , ω_o , θ_i and θ_o are the input and output radian frequencies and phases respectively, T_s is the sampling time. Two input signals are multiplied with the two signals generated by oscillator. This results in two quadrature signals $(r_I(n),\ r_Q(n))$ given as:

$$r_I(nT_s) = x_I(nT_s)y_I(nT_s) + x_Q(nT_s)y_Q(nT_s) = \sqrt{2p(nT_s)}\cos(\Delta\omega nT_s + \Delta\theta), \tag{5}$$

$$r_Q(nT_s) = x_I(nT_s)y_Q(nT_s) - x_Q(nT_s)y_I(nT_s) = -\sqrt{2p(nT_s)}\sin(\Delta\omega nT_s + \Delta\theta). \tag{6}$$

Where $\Delta\omega = \omega_i - \omega_o$, $\Delta\theta = \theta_i - \theta_o$. The quadrature signals $(r_I(n),\ r_Q(n))$ pass through a differentiator circuit.

$$\begin{aligned}\frac{d}{dt}r_I(t)\,|_{t=nT_s} &\approx \{r_I(nT_s) - r_I(n-1)T_s\}\times\frac{1}{T_s}\\ &= -\sqrt{2p(nT_s)}\Delta\omega\sin(\Delta\omega nT_s + \Delta\theta) + (\frac{d}{dt}\sqrt{2p(t)})\,|_{t=nT_s} \times\cos(\Delta\omega nT_s + \Delta\theta),\end{aligned} \tag{7}$$

$$\begin{aligned}\frac{d}{dt}r_Q(t)\,|_{t=nT_s} &\approx \{r_Q(nT_s) - r_Q(n-1)T_s\}\times\frac{1}{T_s}\\ &= -\sqrt{2p(nT_s)}\Delta\omega\cos(\Delta\omega nT_s + \Delta\theta)\text{- }(\frac{d}{dt}\sqrt{2p(t)})\,|_{t=nT_s} \times\sin(\Delta\omega nT_s + \Delta\theta).\end{aligned} \tag{8}$$

The output of the differentiator is

$$\begin{aligned}a(nT_s) &= r_Q(nT_s)\times\frac{d}{dt}r_I(t)\,|_{t=nT_s} - r_I(nT_s)\times\frac{d}{dt}r_Q(t)\,|_{t=nT_s}\\ &= 2p(nT_s)\Delta\omega[\sin^2(\Delta\omega nT_s + \Delta\theta) + \cos^2(\Delta\omega nT_s)]\\ &= 2p(nT_s)\Delta\omega.\end{aligned} \tag{9}$$

The signals $(r_I(n),\ r_Q(n))$ also at the same time passes through squaring circuit. The output of squaring circuit can be considered as power of input signal.

$$\begin{aligned}b(nT_s) &= r_I^2(nT_S) + r_Q{}^2(nT_S)\\ &= 2p(nT_s)[\sin^2(\Delta\omega nT_s + \Delta\theta) + \cos^2(\Delta\omega nT_s)]\\ &= 2p(nT_s).\end{aligned} \tag{10}$$

The output of divider is

$$d(nT_s) = \frac{a(nT_s)}{b(nT_s)} = \Delta\omega. \tag{11}$$

The accumulator starts to accumulate frequency difference until the generated frequency of oscillator is the same as the input frequency. The accumulator equation is

$$\mu\, d\,(nT_S) = c(nT_s) - c((n-1)T_S). \tag{12}$$

Where "μ" is a variable which determines the speed of locking and stability of the system. NCO output is adjusted according to the output of accumulator. Feedback loop of the estimator continues until there is no difference between the input frequency and NCO generated frequency. NCO equation is

$$\omega_o(nT_s) = \omega_c + \sum_{k=0}^{n-1} c(kT_s). \tag{13}$$

Where ω_c is the center frequency. When the generated frequency from NCO is equal to the input frequency, the accumulator saturates. Thus at $\omega_i(nT_s) = \omega_0(nT_S)$

$$c(nT_s) = \omega_i T_s = 2\pi f_i T_s \tag{14}$$

2) Stability

Linear model (z-model) of the system is shown in Fig. 5. Following equations are obtained from Fig. 5

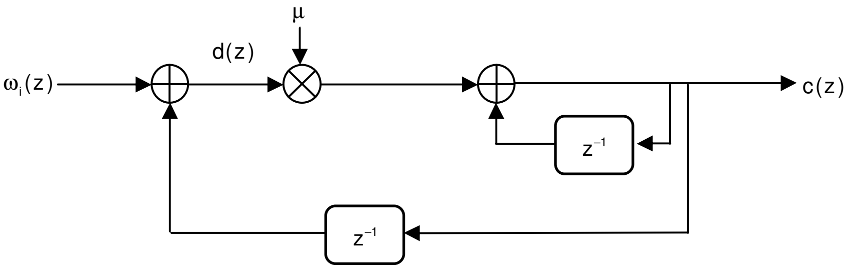

FIGURE 5: Z-model of proposed estimator.

$$c(z) = \omega_i(z) - z^{-1}c(z), \tag{15}$$

$$\mu d(z) = c(z)(1 - z^{-1}), \tag{16}$$

$$\omega_i(z) = c(z)\left(z^{-1} + \frac{1 - z^{-1}}{\mu}\right). \tag{17}$$

Therefore, transfer function of the proposed estimator

$$T(z) = \frac{c(z)}{\omega_i(z)} = \frac{\mu z}{z - (1-\mu)}. \tag{18}$$

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

B. Phase Estimator

The phase estimator used in this paper is a low noise digital phase-locked loop (DPLL) [17]. Conventional DPLL cannot completely remove the high frequency ripples present in the signal. These ripples come from the phase detector (multiplier) which multiplies the input signal with the signal generated by NCO. The result of multiplication is a signal with two terms; high frequency term and low frequency term. In conventional PLL, it assumed that the LPF will remove the high frequency term completely, but first order LPF is not able to remove all ripples. One solution to this problem is to use higher order LPF but this leads to instability of PLL. The other solution could be lowering the cut-off frequency of LPF, but also this will lower the bandwidth of PLL. Our proposed structure of DPLL aims to use a first order LPF to provide stability for the system besides the ability to remove the ripples. Fig. 6 shows the proposed design of DPLL.

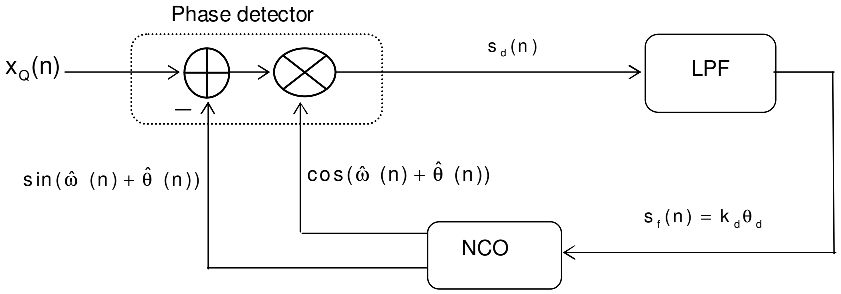
Phase detector
$x_Q(n)$
−
$s_d(n)$
LPF
$\sin(\hat{\omega}\ (n)+\hat{\theta}\ (n))$
$\cos(\hat{\omega}\ (n)+\hat{\theta}\ (n))$
$s_f(n) = k_d\theta_d$
NCO

FIGURE 6: Block diagram of proposed PLL.

$$s_d(nT_s) = k_d\{A_i \sin(\omega_i nT_s + \theta_i) - A_o \sin(\hat{\omega}_i nT_S + \hat{\theta})\} \times A_o \cos(\hat{\omega}_i nT_S + \hat{\theta}). \tag{19}$$

Where $A_i = \sqrt{2p(nT_s)}$, A_o, $\hat{\omega}_i$, $\hat{\theta}$ are amplitude, radian frequency and phase of NCO generated signal respectively, k_d is the gain of multiplier. The first stage of propose architecture estimates the frequency so at $\omega_i = \hat{\omega}_i$

$$s_d = \frac{A_i A_o k_d}{2}\{\sin(2\omega_i nT_s + \theta_i + \hat{\theta}) + \sin(\theta_i - \hat{\theta}) - \sin(2\omega_i nT_s + 2\hat{\theta})\}. \tag{20}$$

From (20) a new term is subtracted from the high frequency term before passing to LPF. Therefore, in this case LPF just have to remove the residual of subtraction instead of removing the entire high frequency term in conventional PLL. As such, a first order LPF is able to remove the residual and stability of DPLL is enhanced. After LPF the resulting signal will be

$$s_f(nT_S) = \frac{A_i A_o k_d}{2}\sin(\theta_i - \hat{\theta}). \tag{21}$$

At $(\theta_i - \hat{\theta}) << 1$, then

$$s_f(n) = \frac{A_i A_o k_d}{2}(\theta_i - \hat{\theta}). \tag{22}$$

This signal is used to control the phase of generated signal from NCO, the DPLL continues to vary the phase until locking occurs and $\theta_i = \hat{\theta}$.

3. SIMULATION RESULTS

We have also carried out computer simulations to compare working of our proposed method with conventional DPLL having both 1st and 2nd order LPF. We have investigated the frequency and phase estimation and tracking performances in both with and without AWGN.

A. Frequency Estimation

First of all we provide frequency estimation comparison between proposed and conventional DPLL. An input signal with frequency of 10 kHz is applied. In these simulations we used a conventional DPLL with both 1st order LPF and 2nd order LPF. Each LPF has cut-off frequency of 1 kHz. The parameters of DPLL are: NCO gain=1024 rad/ v. s, f_i =10.5 kHz, f_{nco} =10 kHz, f_s =100 kHz, while the parameters of proposed estimator are f_{nco} =100 Hz, f_s =100 kHz.

Results of simulations are shown in Fig. 7. Performance of a conventional DPLL with a first order LPF is shown in Fig. 7(a). It is evident that the LPF cannot eliminate the double frequency component completely and overshoot occurs. The settling time, which is time taken by the PLL to reach the frequency estimate is a very important measure of PLL performance. In this case settling time is around 1 ms. In case of DPLL, to remove the higher frequency ripple, a higher order LPF is employed. We have also performed simulations of PLL using 2nd order LPF and results are shown in Fig. 7(b). It can be seen that the high frequency ripple has been suppressed better than a first order LPF. However, the overshoot and ringing still exist and settling time has increased. Furthermore, estimation range is reduced and stability becomes critical in this case. In Fig. 7(c) the proposed method estimates the input frequency in 0.5 ms with no ripples (although no loop filter is used in the system), the estimation range is wide from 0 up to half the sampling frequency. The system is also more stable than conventional PLL.

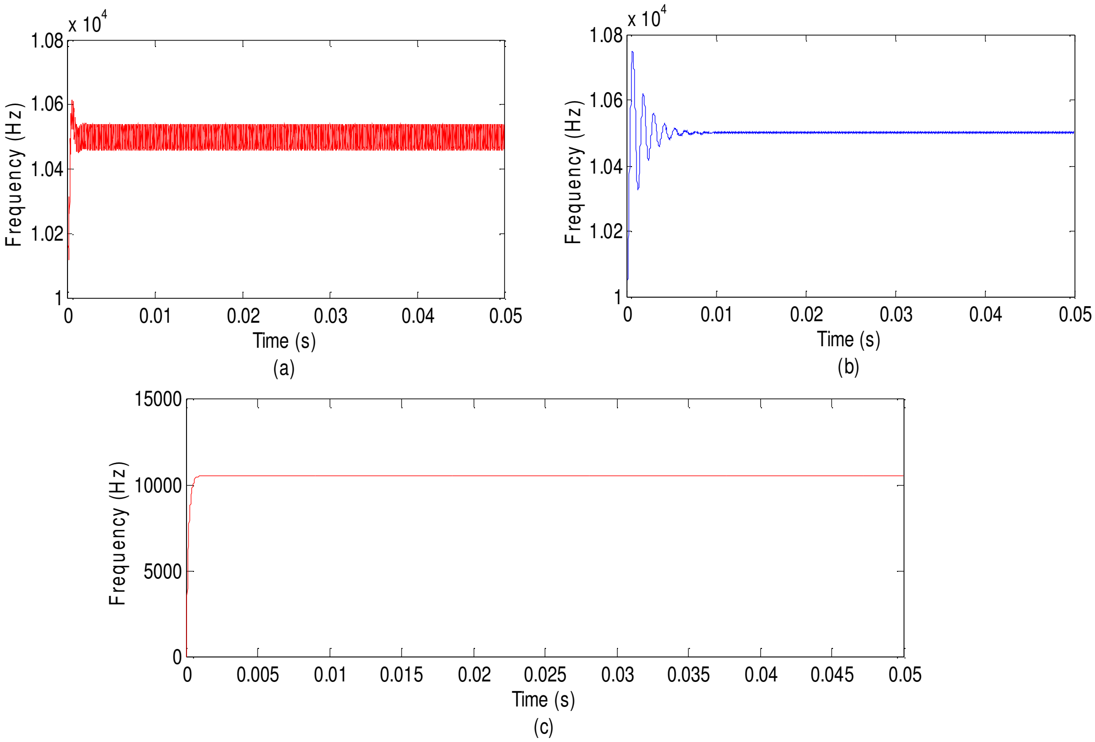

FIGURE 7: Frequency estimation curves of (a) Conventional DPLL with 1st order LPF, (b) Conventional DPLL with 2nd order LPF, (c) Proposed Estimator.

B. Frequency and Phase Estimation

The aim of this simulation is to determine the locking time for both frequency and phase. The estimator receives input sinusoidal signal $x_Q(n)=\sin(10000n+pi/4)$, the center frequency of the NCO is $f_{nco} = 100$ Hz. The proposed estimator estimates the input frequency as shown in Fig. 8 while phase estimation is shown in Fig. 9. It's noticed that the frequency estimator locked the frequency after a very short time ($3\mu s$), phase estimator locked the phase after ($0.1\,ms$) as shown in Fig. 8.

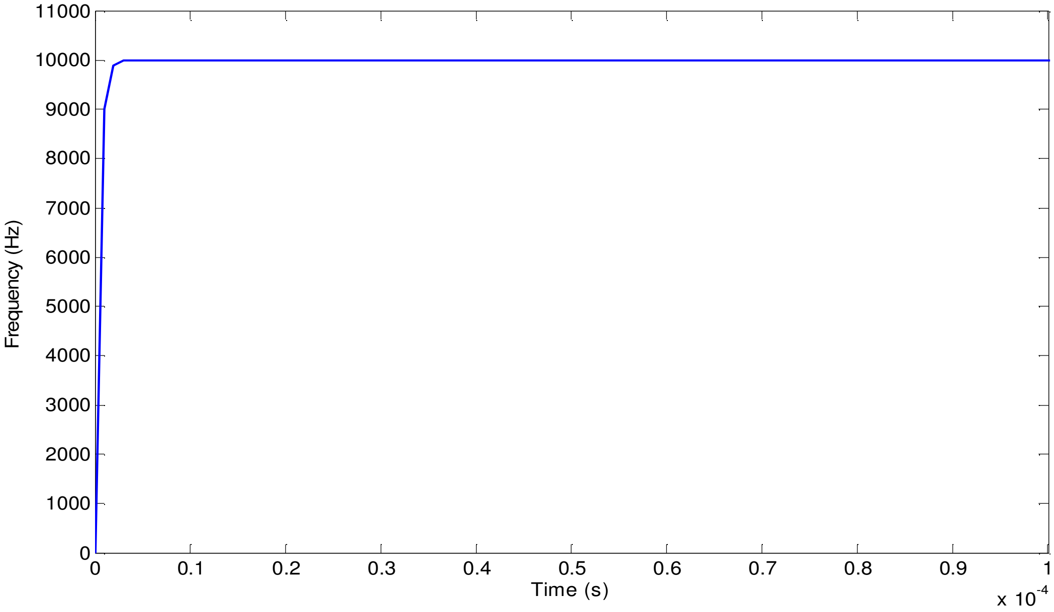

FIGURE 8: Estimated Frequency.

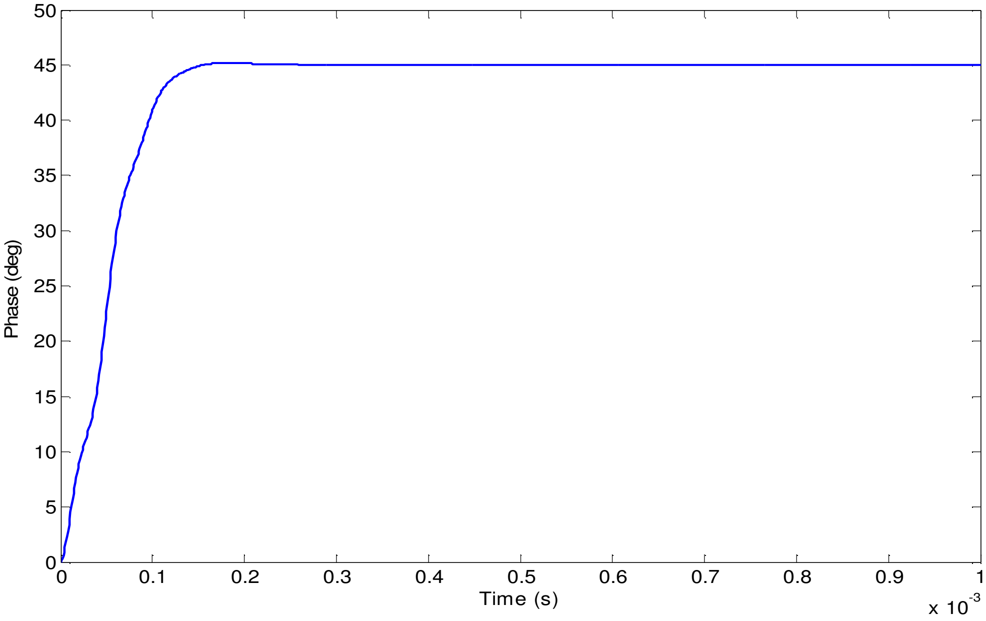

FIGURE 9: Estimated Phase.

C. Frequency and Phase Estimation with AWGN

When the input signal is corrupted with AWGN, two moving average filters (MAF) with length of 10 stages are used for I and Q phase input signals. The MAF is a low pass finite impulse response (FIR) filter commonly used for smoothing an array of sampled data/signal. The following simulations are performed to investigate the effect of AWGN on both conventional DPLL with 1st order LPF and 2nd order LPF and proposed estimator. MAF with the same lengths are added to both architectures. An input signal is applied to three architectures with three different frequencies 50.5, 51.5, 52.5 kHz, all input signals have input phase $\pi/8$ and AWGN with SNR= 10 dB at sampling frequency of 10 MHz. Fig. 10 shows estimations of the three frequencies with conventional DPLL with 1st order LPF, 2nd order LPF and proposed estimator respectively. DPLL with 1st order DPLL estimates the first frequency with high noise level and noise variance is 3269.7. While DPLL with DPLL with 2nd order LPF estimates the second frequency with a reduction in noise level causes the variance to be 1618. Proposed estimator estimates the third frequency with a higher reduction in noise level causes the variance to further reduce to 999. It can be seen that, the proposed architecture produces the smallest values for each estimated frequency although loop filter is not used.

Fig. 11 shows the estimates of the input phase for the three architectures. DPLL with 1st order LPF, estimates the input phase with variance 128.5932. DPLL with 2nd order DPLL estimates the phase with variance 118.7396. The proposed estimator estimates the input phase with variance 99.2743. As the proposed estimator estimates the input frequency with further reduction in variance, it also estimates the input phase with reduction in variance compared to conventional DPLL.

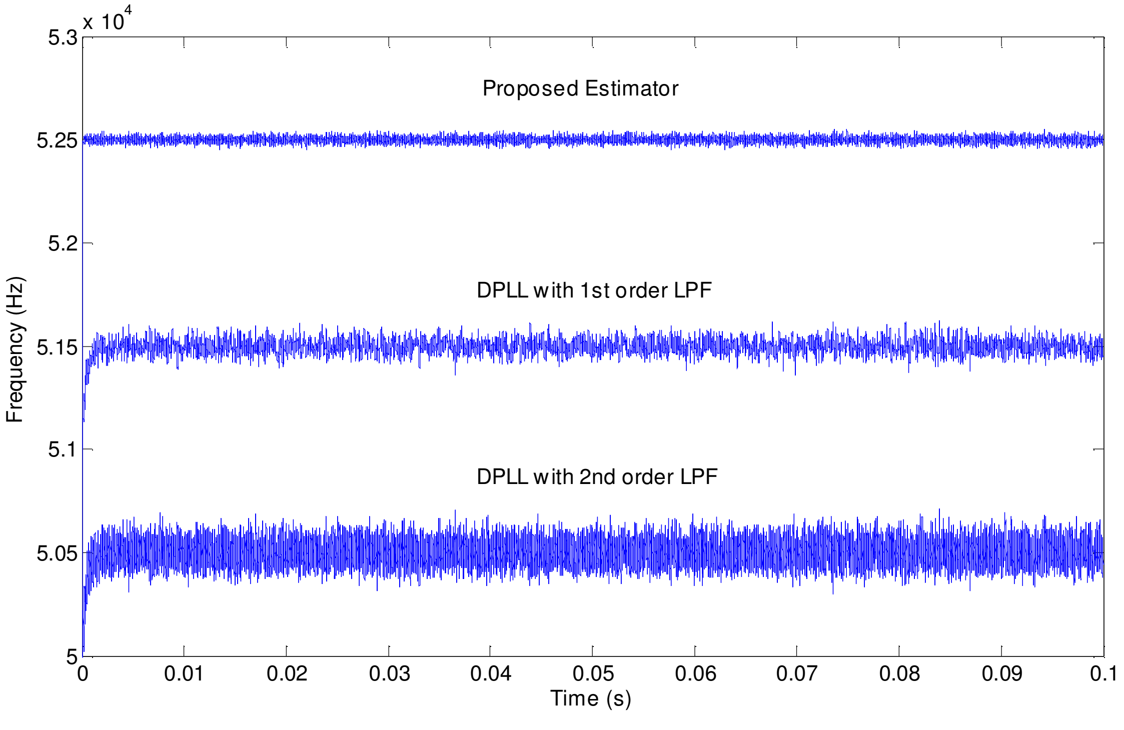

FIGURE 10: Frequency Estimation Comparison.

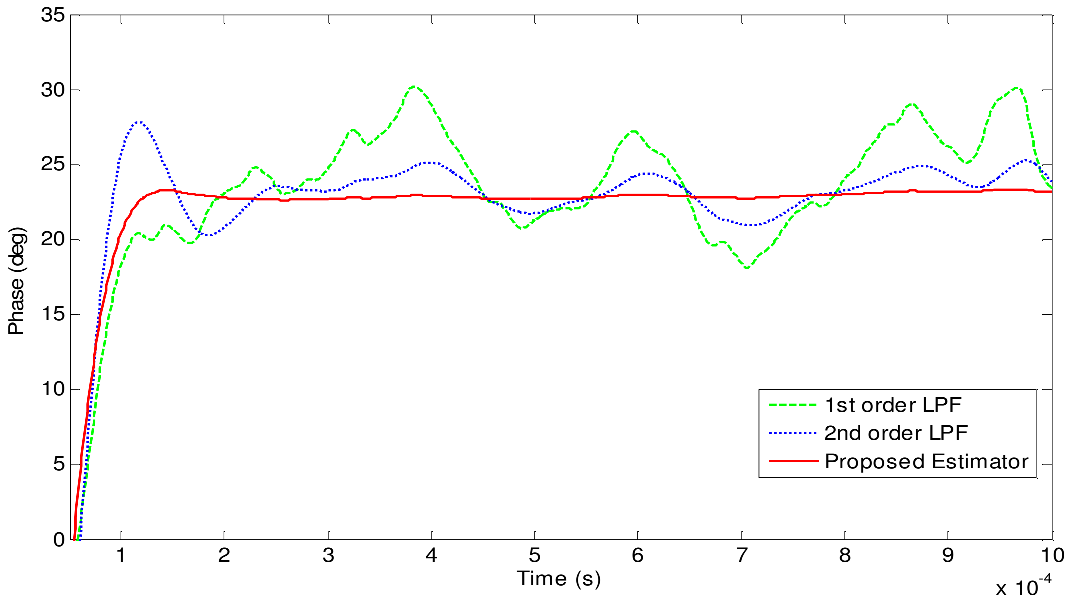

FIGURE 11: Phase Estimation Comparison.

D. Frequency, Power and Phase tracking

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 and phases within the range $\pi/8:\pi/2$ rad. 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. Fig. 14 shows phase estimation, it is seen from the figure that the locking time for each phase is greater than the frequency locking time because phase estimation begins after frequency estimation is finished this also explains the gaps between phases.

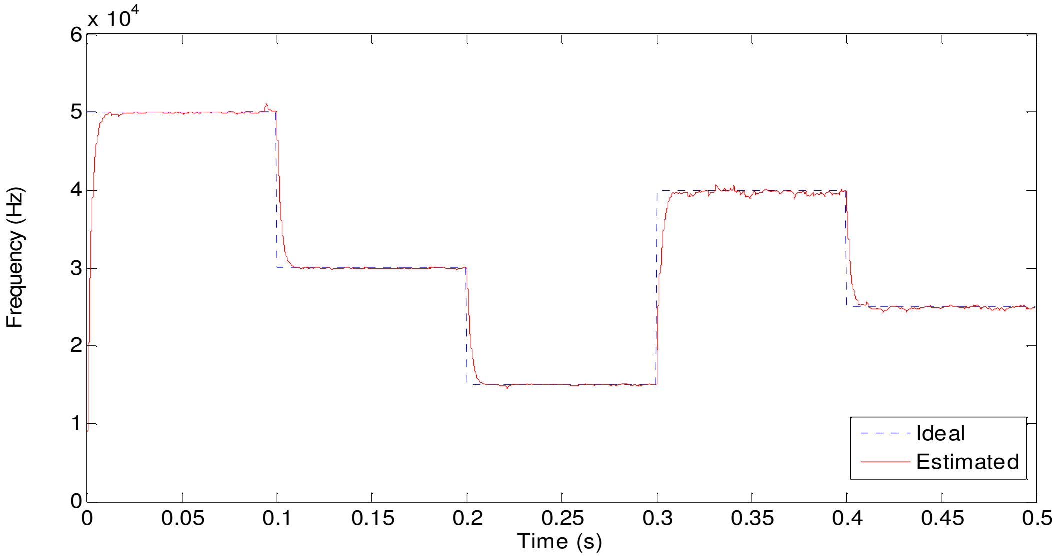

FIGURE 12: Frequency tracking.

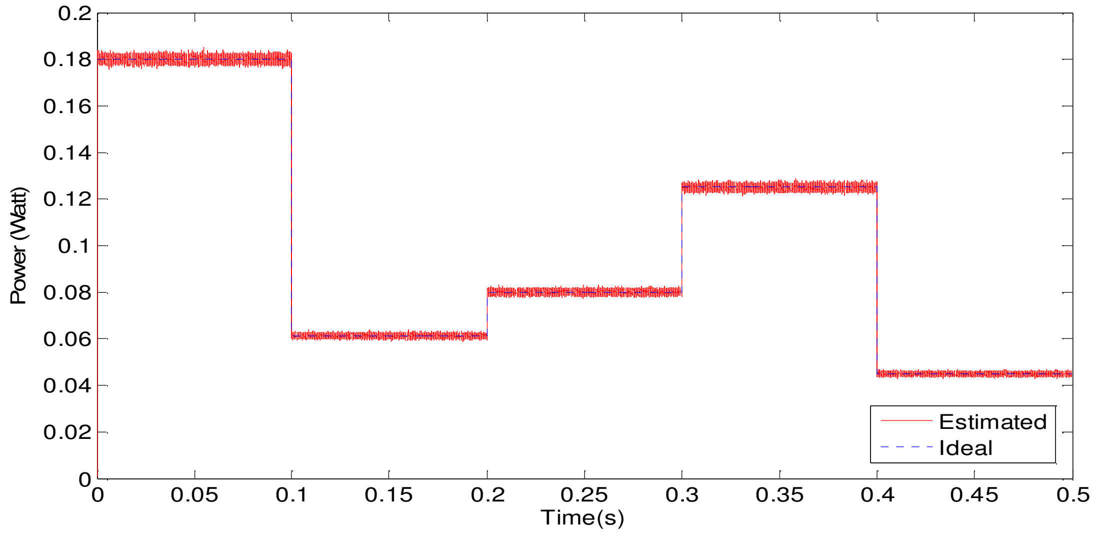

FIGURE 13: Power estimation.

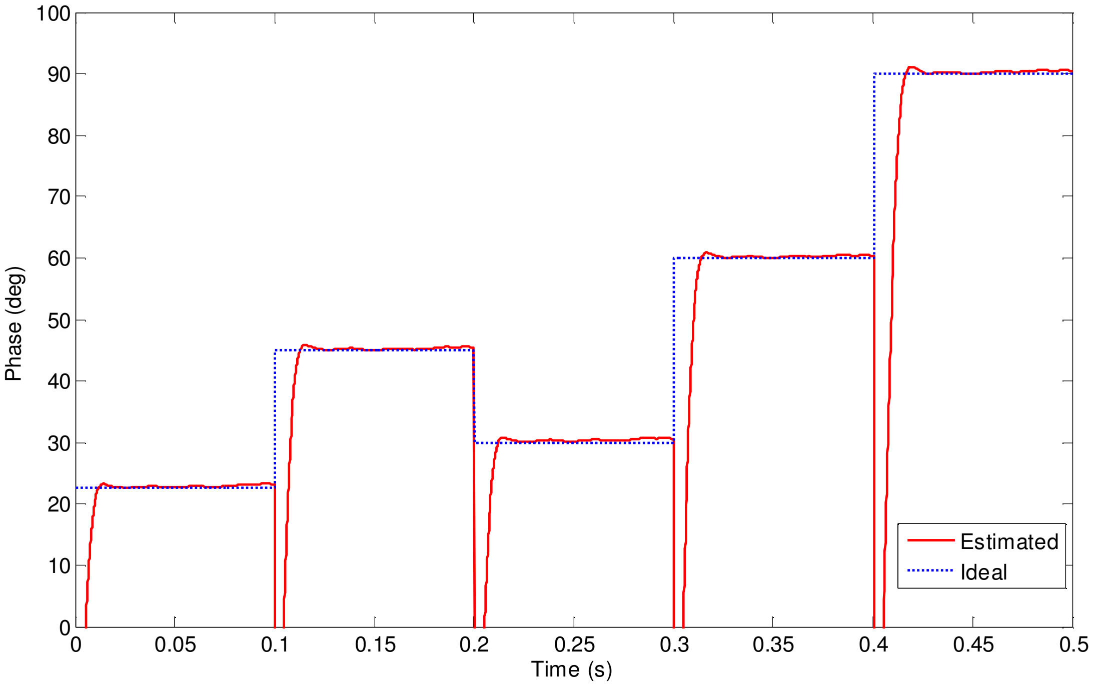

FIGURE 14: Phase tracking.

4. FPGA IMPLEMENTATION

The estimator is modeled using XILINX system generator tool. VHDL code is used to describe the proposed estimator [18:21]. All signals of the estimator model are fixed point signals with 16 bits. The estimator was implemented using XCSD1800A-4FG676C Spartan-3A DSP board. The hardware model simulation 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 1. 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.

TABLE 1: FPGA resource utilization

Component	Used	Available	Utilization
Number of slice flip flops	180	33280	1%
Number of 4 input LUTs	1100	33280	3%
Number of occupied slices	594	16640	3%
Number of DSP48As	13	84	7%
Number of bonded IOBs	97	519	18%
Maximum frequency	201 MHz		
Power consumption	197 mW		

5. CONCLUSION

Conventional DPLL used for carrier synchronization is unable to synchronize signals with large frequency shift. DPLL has other drawbacks such as overshot, ringing, limited tracking range and instability. This paper presented a new structure which can solve these problems. The proposed architecture divided the synchronization task to two steps. The first step is the estimation of the large frequency offset through a frequency estimator. The second step is to use the estimated frequency to estimate the phase through a low noise DPLL, which has the ability to estimate phase accurately by removing the internal noise of DPLL. Computer simulations performed showed significant improvements over conventional DPLL. The paper also discussed hardware implementation of the system 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.

6. 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.

7. REFERENCES

[1] J. G. Proakis, M. Salehi, Digital Communications, Mcgraw-Hill, 2008.

[2] Y. Wang, Y. Zhang, Z. Yang, “Parallel Carrier Recovery in All-Digital Receiver,” in Proc. WICOM, pp. 1-4, 2009.

[3] G. Karam, J. Kervarec, H. Sari, P. Vandamme, ”All-digital implementation of the carrier recovery loop in digital radio systems,” International Conference on Communications, pp. 175-179, 1991.

[4] M. Padua, S. Deckmann, G. Sperandio, F. Marafao, D. Colon “Comparative analysis of Synchronization Algorithms based on PLL, RDFT and Kalman Filter,” IEEE International Symposium on Industrial Electronics, pp. 964-970, 2007.

[5] Y. Linn,” Robust M-PSK phase detectors for carrier synchronization PLLs in coherent receivers: theory and simulations ,” IEEE Transactions on Communications, Vol. 57, pp. 1794-1805, 2009.

[6] K. Lim, S. Choi, B. Kim , ” Optimal loop bandwidth design for low noise PLL applications,” Proceedings of the ASP-DAC '97, pp. 425-428,1997.

[7] 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. of Electronics, circuits and systems IEEE conference, pp. 802-805, 2008.

[8] W. Ping-Ying, Fu. Chia-Huang, ”All digital modulation bandwidth extension technique for narrow bandwidth analog fractional-N PLL,” in Proc. of ESSCIR, pp. 270-273, 2010.

[9] 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.

[10] S. Changhong, C. Zhongze, Z. Lijun, L. Yong, “Design and Implementation of Bandwidth Adaptable Third-order All digital Phase-Locked Loops,” in Proc. WICOM, pp.1-4, 2010.

[11] 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.

[12] 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.

[13] 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.

[14] 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.

[15] 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.

[16] M. Saber, M.T.A. Khan, Y. Jitsumatsu, “Frequency and Power Estimator for Digital Receivers in Doppler Shift environment." Signal Processing: An international journal (SPIJ), vol. 5, pp. 192-208, Dec. 2011.

[17] 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.

[18] P. Pong Chu. RTL Hardware Design Using VHDL: Coding for Efficiency, Portability and Scalability. Wiley-IEEE Press, 2006.

[19] Xilinx Inc. system generator for DSP user guide. Xilinx, 2009.

[20] W.Y. Yang. Matlab/Simulink for digital communication. A-Jin, 2009.

[21] Pong p. chu. FPGA Prototyping by VHDL Examples: Xilinx Spartan-3 Version. Wiley-Interscience, 2008.