Efficient Phase Recovery System

Mohamed Saber
Department of Electronics and Communications, National Research Institute of Astronomy and
Geophysics, Helwan, Egypt
Corresponding author, e-mail: mohsaber@gmail.com

Abstract

An efficient, stable and reliable system to detect the unknown phase of the input sinusoidal signal is presented. The proposed design is a feedback loop consists of subtractor, peak detector and direct digital synthesizer (DDS). DDS generates a sinusoidal signal with the same frequency of input signal and with initial zero phase. An error signal is the output of subtracor after subtracting DDS signal from input signal. The peak of amplitude of error signal has a mathematical relation to the phase difference between the two input signals. Peak detector is used to extract the phase difference from error signal and passes the result to DDS to control the generated phase. The loop continues until the generated phase of DDS is the same as the input unknown phase. The proposed phase estimator has been implemented using a field programmable gate array (FPGA), consumed less power about 330 mW, and worked at 200 MHz clock.

Keywords: phase recovery, direct digital frequency synthesizer, FPGA, PLL

1. Introduction

In digital receivers it is important to synchronize the phase and frequency of the local oscillators in both transmitter and receiver to perform perfect coherent demodulation. The transmitted signal is shifted in frequency and phase due to many causes such as the Doppler effect. Phase shift must be estimated and compensated to perform demodulation and this operation is called phase recovery [1-2]. Synchronization systems can be divided into two main types:

1.1. Feedback (FB) Synchronization

The feedback synchronization systems involve the use of a closed loop form. An example of a feedback scheme includes the well-known Costas loop used for carrier recovery in satellite and point-to-point systems and phase locked loop (PLL) [3-4]. Costas and PLL based carrier synchronization schemes typically have loop bandwidths (BL) on the order of one thousandth of the symbol rate for optimal carrier phase tracking. As a loop-settling times are proportional to (1/BL), such loops require significant time to achieve stable synchronization. The PLL consists from phase detector (multiplier), loop filter (low pass filter), and voltage controlled oscillator (VCO) [5]. The performance of the PLL in phase recovery has some limitations such as [6]:
a) Internal frequency ripples which, results from the phase detector used in a PLL, it needs higher order LPF to be removed, which leads to unstable PLL [7-8].
b) PLL suffers from ringing or long settling time, and overshoot [9-10].

1.2. Feedforwad (FF) Synchronization

In this system, phase synchronization is performed using the incoming matched filter samples without any loop form. One of the attractive characteristics of this method is the absence of hang up and cycle slip problems since there is no feedback (estimates are updated block by block). Cycle slips dodge conventional PLL and Costas loops. The absence of cycle slipping and hang-ups allows rapid acquisition of short burst type signaling [11-12]. Furthermore, digital signal processing (DSP) implementation of FF synchronizers can make the receiver design more flexible and less expensive. Further, in FF systems, there is no feedback path from the synchronizer to the analog front-end of the receiver, which allows modular design and independent testing, resulting in rapid product development. The main drawback in this

type is that, it needs a complex implementation circuits which increase the cost and power consumption. Based on the ways of decoded data, feedforward synchronization algorithms are also further classified into the following types:

a) Data-Aided (DA) systems

Data aided synchronization are implemented by incorporating known preamble symbols that are used by feedforward loop to aid in the estimation [13-14]. The disadvantage of DA systems is that they require an overhead transmission.

b) Decision-Directed (DD) systems

Decision-directed synchronizers use an estimate of the data and not the true data extracted from the transmitted preamble [15]. The performance of these algorithms is only optimal at high SNR where its performance approaches the performance of the DA when the signal-to-noise ratio (SNR) is high.

c) Non-Data Aided (NDA) systems

Neither data nor any decision on data is used in this technique for obtaining the estimates of synchronization parameters. Instead, it averages over the data to obtain reliable estimates [16-17]. At low SNR, NDA is the only available preamble-less rapid technique to estimate phase, frequency, and symbol timing. The main disadvantage of the NDA system is that it degrades heavily once the received signal has been distorted by multipath fading. Still, at low SNR it operates more reliably than its DD/DA counterparts.

It is clear that traditional types of phase estimators has a drawbacks limit its performance. The feedback methods generally suffer from overshoot, long settling time, and instability [18]. While feedforward methods suffers from complexity in design and extra hardware requirement, which leads to high power consumption and cost [19].

The proposed method to recover the input unknown phase is a feedback loop, to provide fast recovery when the input phase is changed rapidly. It provides stability, reliability, and simple structure without complexity.

2. Proposed Phase Estimator

The proposed circuit is shown in Figure 1, it consists of a phase detector, peak detector and DDS only, no loop filter is needed. The phase detector is just a subtractor. The phase detector receives the input sinusoidal signal with unknown phase and another sinusoidal signal generated by DDS, with the same frequency and estimated phase.
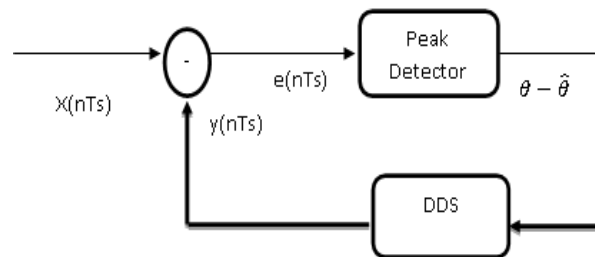

Figure 1. Proposed Phase Estimator

$$x(nT_s) = A \sin(\omega nT_s + \theta) \tag{1}$$

$$y(nT_s) = B \sin(\omega nT_s + \hat{\theta}) \tag{2}$$

Where A, B, ω, θ, $\hat{\theta}$ are the amplitudes, frequency, phases of the input and the generated NCO signals respectively. T_s is the sampling time. The error signal which is the output of the phase detector is $e(nT_s)$

$$e(nT_s) = x(nT_s) - y(nT_s)$$
$$e(nT_s) = A \sin(\omega nT_s + \theta) - B \sin(\omega nT_s + \hat{\theta}) = C \sin(\omega nT_s + \phi) \tag{3}$$

Where c, ϕ are the amplitude and phase of the error signal. Using trigonometric identities we can expand (3) to be

$$A \sin(\omega n T_s) \cos(\theta) + A\cos(\omega n T_s) \sin(\theta) - B \sin(\omega n T_s) \cos(\hat{\theta}) - B\cos(\omega n T_s) \sin(\hat{\theta})$$
$$= C \sin(\omega n T_s) \cos(\phi) + C \cos(\omega n T_s) \sin(\phi) \qquad (4)$$

Eqn (4) can be arranged to be

$$[A \cos(\theta) - B \cos(\hat{\theta}) - C \cos(\phi)]\sin(\omega n T_s)$$
$$= [-A \sin(\theta) - B \sin(\hat{\theta}) + C \sin(\phi)]\cos(\omega n T_s) \qquad (5)$$

The terms inside the parentheses in (5) must be both equal, to be zero, so squaring both sides will be

$$C^2 \cos^2(\phi) = A^2 \cos^2(\theta) + B^2 \cos^2(\hat{\theta}) - 2AB \cos(\theta)\cos(\hat{\theta}) \qquad (6)$$

$$C^2 \sin^2(\phi) = A^2 \sin^2(\theta) + B^2 \sin^2(\hat{\theta}) + 2AB \sin(\theta)\sin(\hat{\theta}) \qquad (7)$$

Adding (6) and (7) will be

$$c^2[\cos^2(\phi) + \sin^2(\phi)]$$
$$= A^2[\cos^2(\theta) + \cos^2(\theta)] + B^2[\cos^2(\hat{\theta}) + \sin^2(\hat{\theta})] - 2AB[\cos(\theta) \cos(\hat{\theta}) - \sin(\theta)\sin(\hat{\theta})]$$

$$C^2 = A^2 + B^2 - 2AB\cos(\theta - \hat{\theta})$$

$$C = \sqrt{A^2 + B^2 - 2AB \cos(\theta - \hat{\theta})} \qquad (8)$$

If A=B

$$C = A\sqrt{2}\sqrt{1 - \cos(\theta - \hat{\theta})} \qquad (9)$$

$$\cos(\theta - \hat{\theta}) = (1 - \frac{c^2}{2})$$

$$(\theta - \hat{\theta}) = \cos^{-1}(1 - \frac{c^2}{2})$$

For small phase difference

$$(\theta - \hat{\theta}) = C^2 \qquad (10)$$

3. Results and Analysis

A different simulation has been done to investigate the performance of the proposed phase recovery system.

3.1. Phase Difference Range

The mathematical relation between the peak of amplitude of error signal (c) and phase difference in Equation (9) and Equation (10) has been investigated. A phase difference range from $(-\pi : \pi)$ is applied to the proposed recovery system and the peak of amplitude of the error signal is plotted in Figure 1. The response of the system agrees with the expected values of (c) and for small phase difference Equation (10) has been applied correctly.
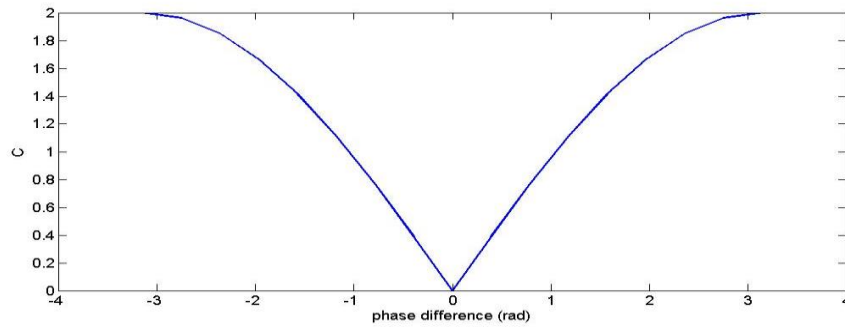
Figure 2. The Response for the Range of Phase Detector

3.2. Different Fixed Value Phase

When the input phase is changed at fixed time the response of the estimator is very important to recover the change in phase as quickly as it is an essential task in the demodulation process such as phase shift keying (PSK). The input phase takes the following values $\left[\frac{\pi}{2}, \frac{3\pi}{4}, \frac{\pi}{9}, \frac{\pi}{6}, \frac{\pi}{2}, \frac{\pi}{4}\right]$ At every milisec and the response of the proposed phase estimator is plotted in Figure 3.
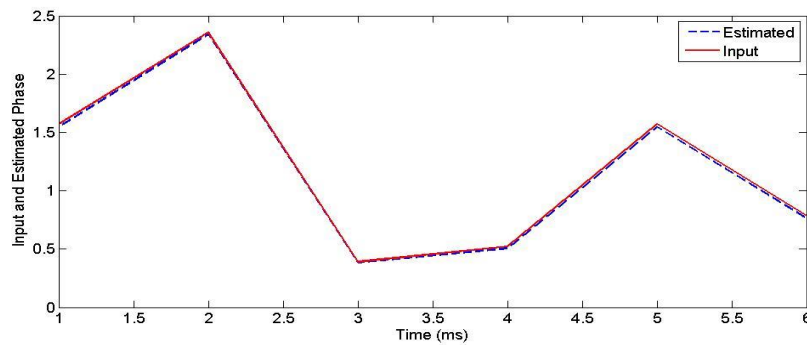

Figure 3. The Estimator Response for Different Input Fixed Time Phases

3.3. Varied Time Phase

Another challenge to the proposed estimator, when the input phase is varied time. An input phase $\theta(t) = \frac{\pi}{4} + \cos(2\pi 2000 t)$ is applied to the estimator and the response is plotted in Figure 4.
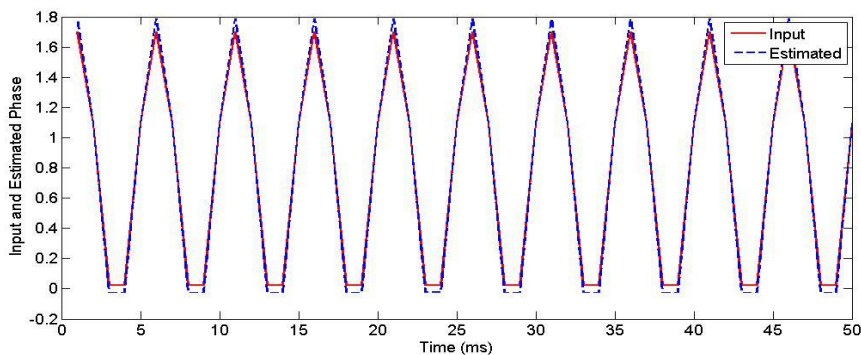

Figure 4. The Estimator Response for Varied Time Phase

3.4. PLL Vs Proposed Estimator

Since PLL has a similar structure with proposed estimator a comparison between PLL response and proposed estimator response to estimate a varying time phase has been done. An input varying time phase $\theta(t) = \frac{\pi}{4} + \frac{\pi}{50}\sin(0.01\pi t)$ is applied to both structures and the both responses are plotted in figure 5. The comparison shows that, the PLL response has a long settling (locking) time and ripples, and the proposed estimator locked the input phase quickly without any ripples. To remove the ripples from PLL it needs to use a higher order LPF, but unfortunately it increases the instability of the system [20-21].
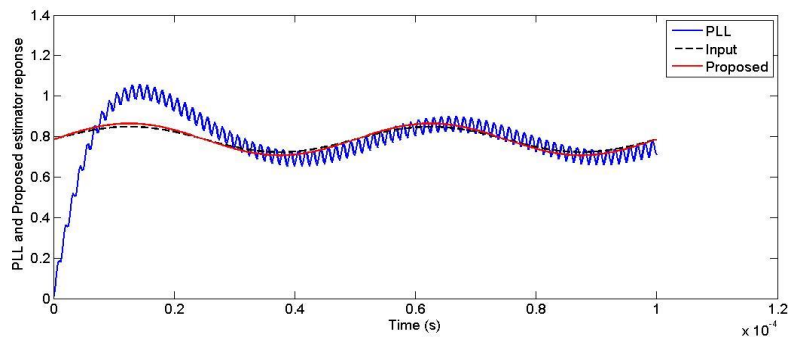

Figure 5. PLL Response Vs Proposed Estimator Response

All previous simulations indicate that the proposed estimator provides an efficient, robust, reliable phase estimator with a low complexity structure suitable for phase recovery operation in digital receivers.

4. Hardware Implementation

The proposed model has been implemented with two methods. The first with mixed circuits, where the subtractor and DDS implemented with FPGA board, and the peak detector is a half wave rectifier (analog component) as shown in figure 4. Equation (9) can be written before phase accumulator using VHDL code [22]. DDS can be considered as two blocks accumulator and read only memory (ROM) or look-up table (LUT) which used to convert the output of the accumulator to phase.
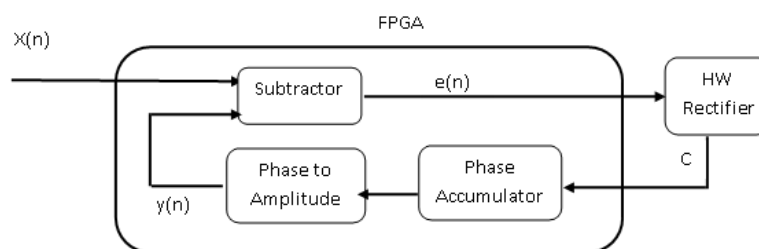

Figure 6. Implementation with Mixed Circuits

The second method is, implementation with FPGA, in this case the peak detector is a delay unit and comparator to compare the input and output of the delay unit $e(n-1) > e(n)$, when $e(n)$ is greater than $e(n-1)$, peak detector stop comparing and the peak has been detected as the current value of $e(n-1)$. Due to the nature of sinusoidal waves, peak detector takes a quarter period of the error signal to reach the peak value. Proposed estimator modeled in VHDL, simulated using Simulink program.
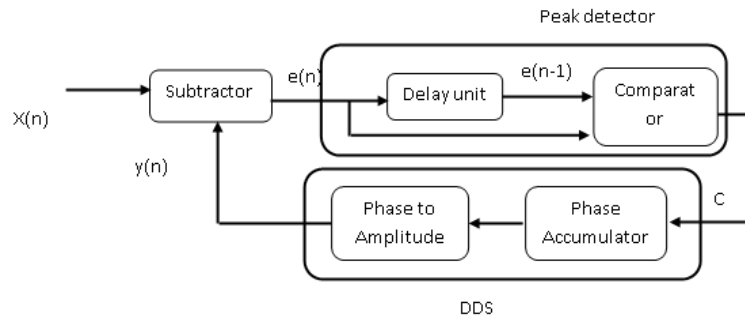
Figure 7. Implementation with Mixed Circuits

The estimator implemented using XCSD1800A-4FG676C Spartan-3A DSP board. FPGA consumed resources are shown in table 1.

Table 1. FPGA consumed resources

Component	Used	Avilable
Number of slice flip flops	100	33280
Number of 4 input LUTs	980	33280
Number of occupied sliced	490	16640
Number of bonded IOBs	12	84
Maximum frequency	196 MHz	
Power consumption	334 mW	

4. Conclusion

Simple, efficient and stable phase estimator method has been presented. Simulations results for different cases, showed the performance of the proposed estimator. Comparison between proposed estimator and PLL (widely used as phase estimator in communication circuits), has been also presented and indicates that the proposed estimator solves many drawbacks in performance of PLL. Proposed estimator has been implemented with FPGA, consumes 334 mW and works at 196 MHz.

References

[1]   Heinrich Meyr, Gerd Ascheid. Synchronization in Digital Communications: Phase, frequency locked loops and amplitude control. Volume-1, New York: John Wiley & Sons. 1990.
[2]   Li Li, WEI Jian-yi, LI Hong-an, ZHANG Xiu-tai. Analysis Modulation Formats in DWDM Transmission System. *Indonesian Journal of Electrical Engineering and Computer Science*. 2013; 11(1): 536-543.
[3]   R Best. Phase-Lock Loops: Design, simulation and Application. McGraw-Hill. 2007
[4]   JP Costas. *Synchronous Communications*. Proceedings of the IRE.1956; 1713- 1718.
[5]   Yuan Gao, Lei Zhang, Jing Yuan. *Performance analysis of Costas-PLL based on four-way signal processing.* IET Radar conference. China. 2013.
[6]   Kuznets N, Kuznets ova, Lenov G, Neittaanmaki P, Yuldashev M, Yuldashev R. *Limitations of the classical phase-locked loop analysis.* IEEE International Symposium on Circuits and Systems (ISCAS). 2015: 533-536.
[7]   J Piqueira J, Monteiro L. Considering second-harmonic terms in the operation of the phase detector for second-order phase-locked loop. *IEEE Transactions On Circuits And Systems*. 2003; 50(6): 805– 809.
[8]   M Saber, Y Jitsumatsu, MTA Khan. Design and Implementation of Low Ripple Low Power Digital Phase-Locked Loop. *Signal Processing: An international journal (SPIJ)*. 2011; 4: 304-317.
[9]   Min M, Mannama V, Paavle T.*Settling time minimization in PLL frequency synthesizer.* In Proc. ICCSC. Russia. 2002: 366-369.
[10] Pallavi, Priyank Laad. *Tradeoffs between settling time and jitter in phase locked loops.* IEEE 56th International Midwest Symposium on Circuit and Systems. Ohio State University. 2013: 746-749.

[11] Geert de Jonghe, marc Moeneclaey. *Cycle slipping behavior of NDA feedforward carrier synchronization for time-varying frequency-nonselective fading channels*. Proceedings of IEEE GLOBECOM. Florida. USA. 1992: 350-354.

[12] G Jonghe, Marc Moeneclaey. *The effect of the averaging filter on the cycle slipping of NDA feedforward carrier synchronizers for MPSK*. Proceedings of the ICC. Chicago. USA. 1992: 365-369.

[13] Umberto Mengali and M Morelli. Data-Aided frequency Estimation for burst digital transmission. *IEEE Trans. Commun*. 1997; 43(1): 23-25.

[14] K Geothals, M Moeneclaey. PSK symbol aided synchronization of ML oriented data-aided algorithms for nonselective fading channels. *IEEE Trans. Comm*. 1995; COM-46 (2).

[15] Hu. Yunpeng, Yu. Hongyi, Hu. Hanying. *Semi-Data-Aided Frequency estimation Algorithm for BPSK burst transmissions*. 2nd International conference on communications and Networking. China. 2007: 655-658.

[16] Estathiou, Aghvami H. Preamble-less non-decision aided (NDA) feedforward synchronization techniques for 16-QAM TDMA demodulators. IEEE Transaction on Vehicular Technology. 1998; 47(2): 673-685.

[17] Marc Moeneclaey, Geert de Jongh. Tracking performance comparison of two Feedforward ML-oriented carrier-independent NDA symbol. IEEE Trans. Comm. 1992; 40(9): 1423-1425.

[18] K Arun, K Selvajyothi. Single Phase Variable Sampling Phase Locked Loop using Composite Observer. *Indonesian Journal of Electrical Engineering and Computer Science.* 2016; 2(1): 49-60.

[19] Imamul Muttakin, Arbai Yusuf, Rohmadi, Wahyu Widada, Warsito. Taruno. *Hardwar Design for Quadrature Phase Detection Algorithm in ECVT*. In Proc. of International Conference on Electrical Engineering, Computer Science and Informatics (EECSI).Indonesia. 2014: 232-235.

[20] Talbot D. Frequency Acquisition Techniques for Phase Locked Loops. Wiley.2012.

[21] Tranter W, Bose T, Thamvichai R. *Basic Simulation Models of Phase Tracking Devices Using MATLAB*, ser. Synthesis lectures on communications. Morgan & Claypool, 2010.

[22] Zhu Huabing, Lian Benlei, Dong Bolin, Feng Xiao. Research on FPGA-based Programmable Logic Controllers' Technology*. Indonesian journal of Electrical Engineering and Computer Science.* 2013; 11(12): 7655-7663.