A Low Power Digital Phase Locked Loop With ROM-Free Numerically Controlled Oscillator

M. Saber *mohsaber@tsubaki.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-u.ac.jp*
Ritsumeikan Asia Pacific University, College of Asia Pacific Studies
1-1 Jumonjibaru, Beppu, Oita, 874-8577, Japan

Abstract

This paper analyzes and designs a second order digital phase-locked loop (DPLL), and presents low power architecture for DPLL. The proposed architecture reduces the high power consumption of conventional DPLL, which results from using a read only memory (ROM) in implementation of the numerically controlled oscillator (NCO). The proposed DPLL utilizes a new design for NCO, in which no ROM is used. DPLL is designed and implemented using FPGA, consumes 237 mw, which means more than 25% saving in power consumption, and works at faster clock frequency compared to traditional architecture.

Keywords: Digital Phase locked loop (DPLL), Field Programmable Gate Array (FPGA), Software Defined Radio (SFDR), Read Only Memory (ROM), Spurious Free Dynamic Range (SFDR).

1. INTRODUCTION

Software Defined Radios (SDRs) are leading the integration of digital signal processing (DSP) and radio frequency (RF) capabilities. This integration allows software to control communications parameters such as the frequency range, filtering, modulation type, data rates, and frequency hopping schemes. SDR technology can be seen in wireless devices used for different applications in military, civil applications, and commercial network. Compared to conventional RF transceiver technologies, the advantage of SDR is its flexibility. SDR provides the ability to reconfigure system performance and functions on the fly [1].

In order to take advantage of such digital processing, analog signals must be converted to and from the digital domain. This is done using analog-to-digital (ADC) and digital-to-analog (DAC) converters. To take full advantage of digital processing, SDRs keep the signal in digital domain as much as possible, digitizing and reconstructing as close as possible to the antenna. Despite an ADC or DAC connected directly to an antenna is a required end goal, there are issues with selectivity and sensitivity that need an analog front [2].

Phase-locked loop (PLL) is one of the most important building blocks necessary for modern digital communications, which is used as a frequency synthesizer in RF circuits, or to recover time and carrier in the baseband digital signal processing. A complete understanding of the concept of PLL includes many study areas such as RF circuits, digital signal processing, discrete time control systems, and communication theory [3]. Traditional PLL consists of three parts; phase frequency detector (PFD), loop filter, and voltage controlled oscillator (VCO).

The traditional analog PLL faces many design problems such as voltage supply noise, temperature noise, and large area consumed by loop filter components like resistors and capacitors. On the other hand DPLL, formed of all digital components, provides a high immunity to supply voltage noise and temperature variation. Moreover, DPLL can be designed by using hardware description language (HDL) with any standard cell library. Thus, the time for redesign and check for errors is reduced. Therefore, DPLL provides a good solution to analog PLL design problems. Unfortunately, DPLL has a critical disadvantage, i.e., high power consumption resulting from the numerically-controlled oscillator (NCO) [4].

The high power consumption of NCO is the result of using ROM, which contains the sampled amplitudes of a sinusoidal waveform. As accuracy of the generated signal increases, the size of ROM increases, which causes high power consumption and reduces the speed of the circuit. We propose a DPLL architecture in which the traditional NCO is replaced by a circuit which generates a cosine waveform using a piecewise-linear approximation.

In section 2, PLL operation is explained. The traditional NCO is described in section 3. Section 4 illustrates a modified NCO which can solve the problems of traditional NCO. In section 5 mathematical model of DPLL in both Z-domain and S-domain is illustrated. In section 6 simulation results. In section 7 hardware implementation of modified NCO and modified DPLL is presented and in the end some conclusions are given.

2. PHASE LOCKED LOOP

PLL is an important component in many types of communication systems. It works in two different manners; to synchronize a carrier in frequency and phase or to operate as a synthesizer. The block diagram of DPLL is shown in Fig. 1. It consists of three main blocks, phase/frequency detector (PD), loop filter and NCO.

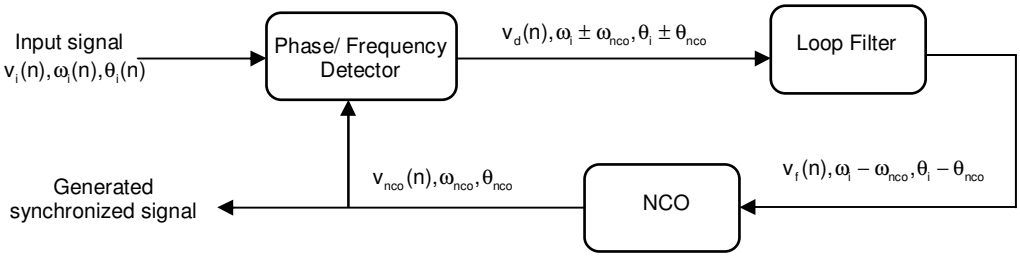

FIGURE 1: Digital phase locked loop in discrete time domain.

The operation of DPLL is as follows: without input signal applied to the system, NCO generates a signal with a center frequency (f_c), which is called the free running frequency. The input signal applied to the system is

$$v_i(n) = A_i \sin(\omega_i\ n + \theta_i), \tag{1}$$

where A_i is the amplitude, ω_i is the angular frequency, and θ_i is the phase of the input signal. Feedback loop mechanism of PLL will force NCO to generate a sinusoidal signal $v_{nco}(n)$

$$v_{nco}(n) = A_o \sin(\omega_{nco}\ n + \theta_{nco}), \tag{2}$$

where A_o is the amplitude, ω_{nco} is the angular frequency and θ_{nco} is the phase of the signal generated by NCO. θ_{nco} is given by

$$\theta_{nco}(n) = k_v \sum_{i=-\infty}^{N} v_f(i), \tag{3}$$

where k_v is the NCO gain constant and $v_f(n)$ is the filter output. If k_m denotes the phase detector (multiplier) gain, then output of the phase detector is

$$\begin{aligned} v_d(n) &= \frac{k_m A_i A_o}{2} \sin(\omega_i\ n + \theta_i) \cos(\omega_{nco}\ n + \theta_{nco}) \\ &= \frac{k_m A_i A_o}{2} [\sin((\omega_i + \omega_{nco})\ n + \theta_i + \theta_{nco}) + \sin((\omega_i - \omega_{nco})\ n + \theta_i - \theta_{nco}], \end{aligned} \tag{4}$$

The first term in (4) corresponds to high frequency component, and the second term corresponds to the phase difference between $v_i(n)$ and $v_{nco}(n)$. Loop filter will remove the first term in (4). If $\omega_i = \omega_{nco}$, then phase difference can be obtained as

$$v_f(n) = k_d[\sin(\theta_i - \theta_{nco})], \tag{5}$$

where $k_d = \frac{k_m A_i A_o}{2}$. If $(\theta_i - \theta_{nco}) \ll 1$, then $V_f(n)$ is approximated by

$$v_f(n) \approx \frac{k_d A_i A_o}{2} (\theta_i - \theta_{nco}). \tag{6}$$

This difference voltage is applied to the NCO. Thus, the control voltage $v_f(n)$ forces the NCO output frequency to change up or down to reduce the frequency difference between ω_{nco} and ω_i. The equation of the generated frequency of NCO is

$$\omega_{nco}(n) = \omega_c + v_f(n), \tag{7}$$

where ω_c is the center frequency of NCO. If the input frequency ω_i is close to ω_{nco}, the feedback manner of PLL causes NCO to synchronize or lock with the incoming signal. Once it is locked, the generated signal of NCO will synchronize the input signal in phase and frequency.

3. TRADITIONAL NCO

Voltage Controlled Oscillator (VCO), which is used in analog PLL generates a sinusoidal waveform whose frequency depends on the input voltage. NCO, which is used in DPLL, generates a digital (sampled) sinusoidal waveform with a fundamental frequency determined by the digital input value (n-bits). As shown in Fig. 2, NCO consists of ROM, and accumulator. The output signal of the accumulator is used as address to the ROM. The input signal to the accumulator consists of the sum of an offset (ω_c) corresponding to the free running frequency, and v_f which is the output of the loop filter [5]. The general equation of generated frequency from NCO is

$$f_{nco} = \left(\frac{v_f}{2^j} + \omega_c\right) \times f_{clk}. \tag{8}$$

where f_{nco} is the generated frequency, ω_c is the center frequency, v_f is an integer value and lies in the range $(-2^{j-1} \le v_f \le 2^{j-1})$, j is number of bits or width of the accumulator, which is 16 bits, and f_{clk} is the clock frequency.

The operation of NCO is as follows: first assuming that the system clock frequency is 50MHz, j=16 and $\omega_c = 1310$, the free running frequency is 1 MHz. Then, as shown in Fig. 3 there are 50 sampling points in one cycle of 1 MHz sinusoidal waveform. NCO generates exactly one cycle of sinusoidal waveform when the input value (v_f) is equal to zero. Since the offset value is 1310, every clock cycle the accumulator accumulates the offset value. Then in 50 cycles the accumulated value will increase by one. The accumulator output will address this value to the ROM and extract the cosine amplitudes values stored in it.

When the input value is greater than zero, the accumulation speed becomes higher. Thus in less than 50 cycles of clock frequency the accumulator increases by 1, this will generate a higher frequency than 1MHz. When the input value is less than 0, a frequency lower than 1 MHz is generated. The problem with using a ROM is that, its size increases to achieve a high spectral purity of the generated waveform. This leads to high power consumption and slow operation of the system.

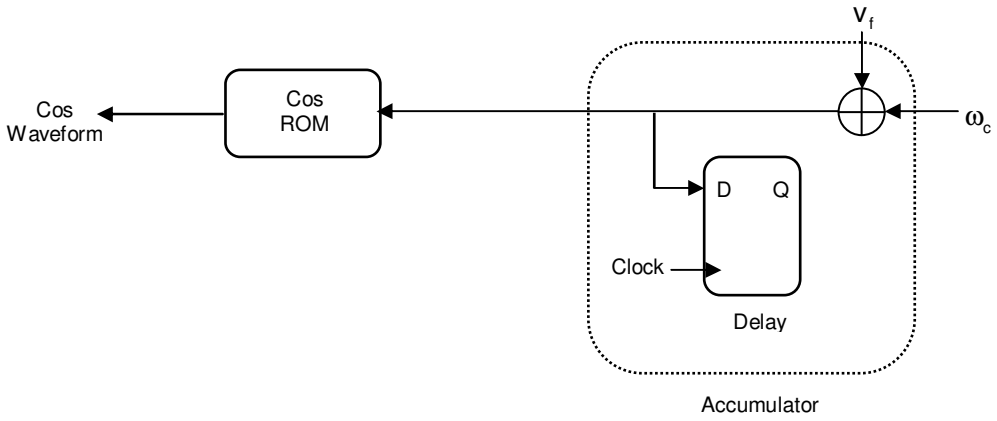

FIGURE 2: Numerically controlled oscillator structure.

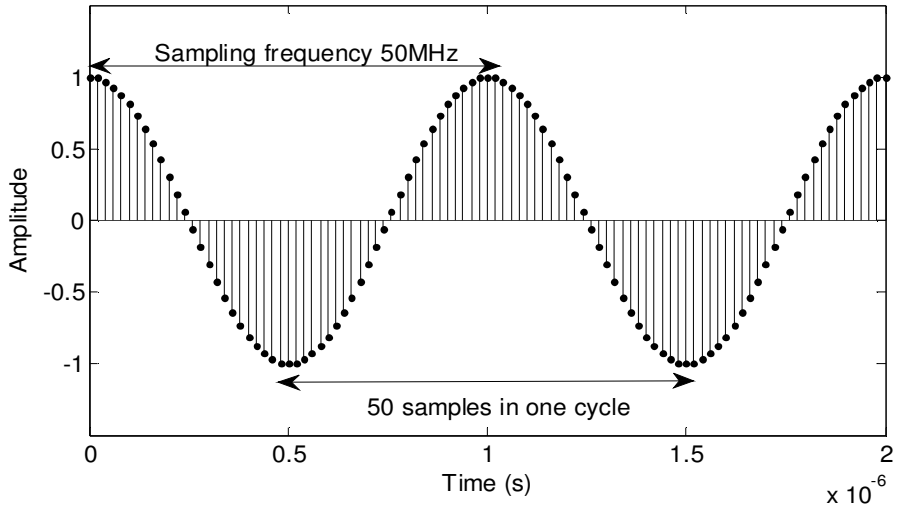

FIGURE 3: Output waveform of NCO.

3.1 Previous Work

NCO which generates sine or cosine output as shown in Fig. 2 differs mostly in the implementation of ROM block. This block is the slowest and consumes high power. The problem of ROM is that, its size grows exponentially with the width of the phase accumulator. Since one normally desires a large number of bits to achieve fine frequency tuning and high spectral purity, several techniques have been invented to limit the ROM size while maintaining suitable performance.

One technique uses the quarter wave symmetry of sine function to reduce the number of saved samples by 4, in which ROM saves only the amplitudes of first quarter and through additional hardware the other quarters are generated [6]. Truncating accumulator output (remove number of most significant bits (MSBs)) is a common method to reduce the size of ROM but this method introduces spurious harmonics [7].

Different angular decomposition techniques proposed to reduce the ROM size consist of splitting the ROM into a number of smaller ROMs, each ROM is addressed by a portion of truncated accumulator output. Generated samples of each ROM are added to form a complete sinusoidal waveform. In order to introduce more reduction in the ROM size, many techniques have been proposed to make an initial approximation of the sine amplitude from the value of the phase angle, and to use the ROM or a combination of ROMs to store correction values [8:11]. Although these methods reduce the power consumption but they still use ROM which causes a residual of high power consumption.

Many other techniques have been proposed using piecewise continuous polynomials to approximate the first quadrant of the sine function. One of them is based on a Taylor-series expansion [12], a simplified 4th degree polynomial [13] and 4th degree Chebyshev polynomials [14]. The drawbacks of the above techniques are that they require additional hardware to make extra computations which increase the complexity of the circuit. The additional hardware consumes power consumption which supposed to reduce.

4. MODIFIED NCO

4.1 Proposed Architecture

In proposed architecture no ROM is used, to provide fast switching, and less power consumption. Instead of using a ROM a piecewise linear approximation is used, that is representing the first quarter of the cosine waveform as linear lines, each line fits a linear equation with slope and bias. Depending on the symmetry of the cosine waveform (have 4 quarters), it can easily deduce the other 3 quarters of the cosine waveform from only the first quarter. The first quarter of the cosine function is divided into eight piecewise linear segments of equal length of the form:

$$\cos(t) \approx a_i t + b_i, \quad \frac{i}{16}\pi \le t < \frac{i+1}{16}\pi, \quad i=0,1,.....7, \tag{9}$$

where a_i is the segment slope and is limited to 4 bits, and b_i is the constant or bias limited to 8 bits. Slopes and biases are chosen using the minimum mean square error (MMSE) criterion, that minimizes the integrated mean square error between the ideal $\cos(t)$ and the approximated cosine function $p(t)$.

$$\text{mmse} = \int_{t=0}^{\pi/2} [\cos(t) - p(t)]^2 dt. \tag{10}$$

Fig. 4 shows a comparison between ideal and approximated cosine waveforms. It seems to be the same except the top and bottom of the waveform, that is because of the linear segments.

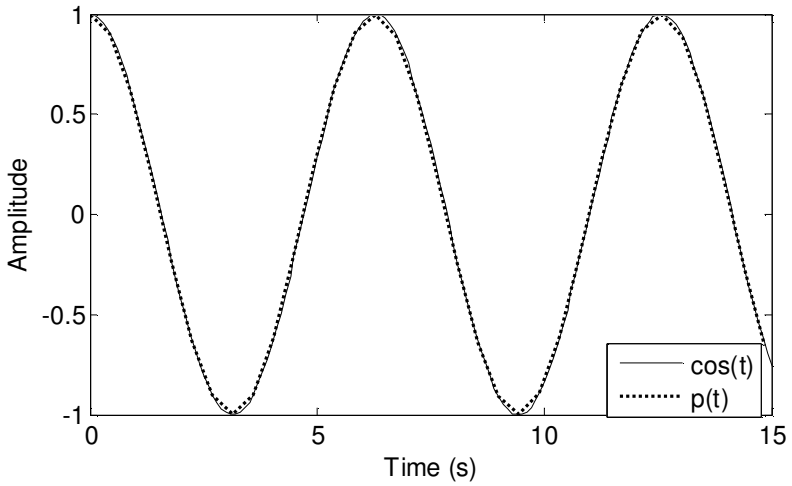

FIGURE 4: Approximated and Ideal cosine waveforms.

The modified NCO consists of two main components and two negation units. Fig. 5 shows the block diagram of each component and the corresponding waveform. Accumulator receives the input signal $v_f(n)$ which represents the phase difference between θ_i and θ_{nco}. The accumulator works as a circular counter. A complete rotation of the accumulator represents one cycle of the output waveform. The accumulator receives a signal with eight bits-length, and the width of the accumulator is j=16 bits, so truncation is done to the output signal of the accumulator to be X signal with L=10 bits' length. The first two most significant (MSBs) bits of the accumulator are used to control the operation of NCO. 2nd MSB controls the sign of signal X before performing the piecewise linear calculation. This negative sign is needed to substitute in the linear function to generate all quarters of the cosine waveform. Second negation is done at the output stage to correct position of second and third quarters. This negation is controlled using XOR function between 1st, and 2nd MSB.

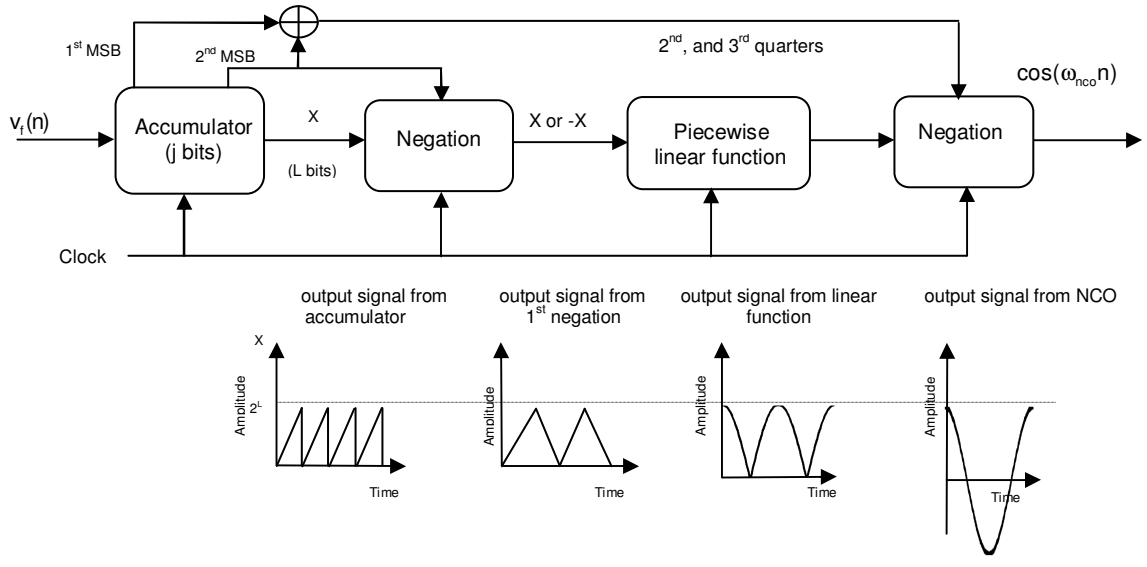

FIGURE 5: Structure of modified NCO.

4.2 Spurious Free Dynamic Range (SFDR)

SFDR is defined as the ratio between the RMS value of the fundamental frequency (maximum signal component) and the RMS value of the next largest noise or harmonic distortion component, (which is referred to as a "spurious" or a "spur") at its output. SFDR is usually measured in dBc (i.e. with respect to the carrier frequency amplitude) or in dBFS (i.e. with respect to the ADC's full-scale range). Depending on the test condition, SFDR is observed within a pre-defined frequency window or from DC up to Nyquist's frequency of the converter (ADC or DAC). Fig. 6 shows how SFDR is measured [15]. Since the modified NCO depends on linear approximation to generate digital samples of cosine waveform, the spectrum of the generated waveform contains spurs at all the spectrum frequencies, and SFDR is used to measure the spectral purity of the generated frequencies.

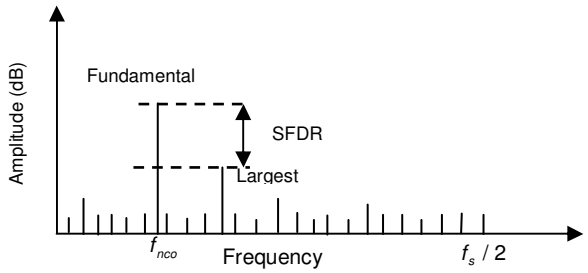

FIGURE 6: SFDR measure.

5. DPLL MATHEMATICAL MODEL

A mathematical model for DPLL is built in z-domain, and s-domain to study the ability of the system to maintain phase tracking when exited by phase steps, frequency steps, or other excitation signals. Fig. 7 and Fig. 8 shows mathematical model of the system in both Z-domain and S-domain respectively.

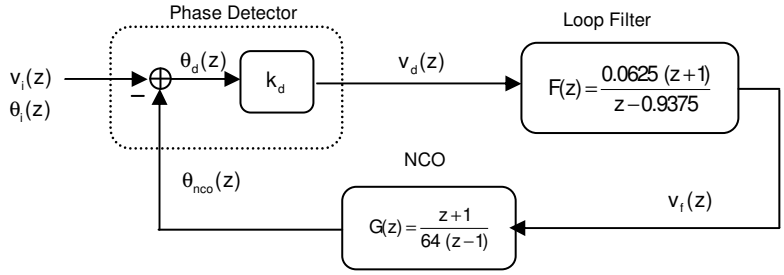

FIGURE 7: DPLL in Z-domain.

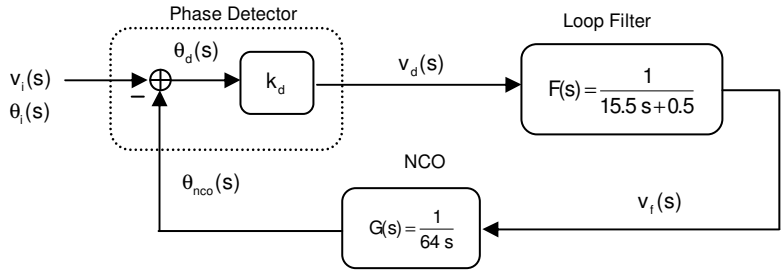

FIGURE 8: DPLL in S-domain.

The phase transfer function of the system in Z-domain is

$$\frac{\theta_{nco}(z)}{\theta_i(z)} = \frac{k_d\ F(z)\ G(z)}{1+K_d\ F(z)\ G(z)} = \frac{1+z^{-1}+z^{-2}}{1025-1982\ z^{-1}+961\ z^{-2}}. \quad (11)$$

To get the step response of the system a relation between $v_f(z)$ and $v_i(z)$ is needed. Assuming the input signal is a unit step of frequency at constant phase

$$\frac{v_f(z)}{v_i(z)} = \frac{F(z)}{1+F(z)\ G(z)} = \frac{64\ (1-z^{-2})}{1025-1982\ z^{-1}+961\ z^{-2}}. \quad (12)$$

Using bilinear transformation, the previous equations are obtained in S-domain

$$\frac{\theta_{nco}(s)}{\theta_i(s)} = \frac{k_d\ F(s)\ G(s)}{1+k_d\ F(s)\ G(s)} = \frac{1}{992\ s^2+32\ s+1} \quad (13)$$

$$\frac{v_f(s)}{v_i(s)} = \frac{F(s)}{1+G(s)\ F(s)} = \frac{64\ s}{992\ s^2+32\ s+1} \quad (14)$$

In the test for stability, DPLL is subjected to a test signal representing a unit step of frequency at constant phase using (14) with $f_s = 50$ MHz [16-17]. As shown in Fig. 9, the system is stable with overshoots at the transient state.

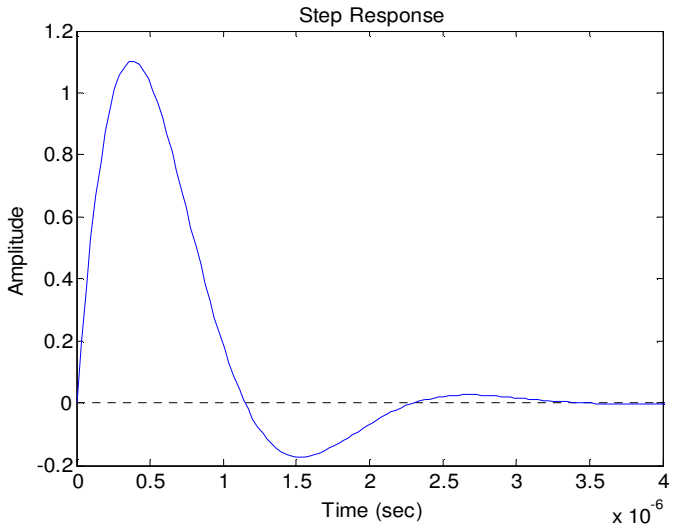

FIGURE 9: DPLL in S-domain.

6. SIMULATION RESULTS

6.1 SFDR of Modified NCO

To measure the SFDR a discrete Fourier transform (DFT) is done for a long repetition period of the generated signal from modified NCO. Difference between the amplitude of the fundamental output frequency and the amplitude of the largest spurs in the dynamic range is noted. Fig. 10 shows the output spectrum for input word of value 1317 representing $v_f(n)$, at a clock frequency

of 50 MHz and an accumulator width j=16. The fundamental frequency is approximately 2 MHz with -30.057 dB, and the spurious appears at 14.46 MHz with -89.925 dB, so SFDR=59.868 dBc.

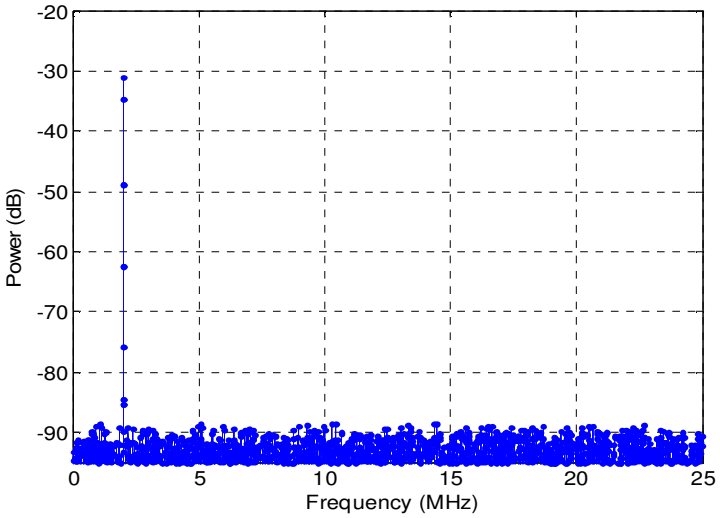

FIGURE 10: SFDR for fundamental frequency of 2 MHz.

6.2 DPLL Synchronization

In this section, we investigate the performances of proposed DPLL's using computer simulations. The proposed DPLL has the following parameters:

$f_s = 10$ MHz, $k_m = 1$, $k_v = 1024$, $A_i = A_o = 1$, $\omega_{nco} = 1$ MHz, and $\theta_{nco} = 0$.

Two types of simulations are done. In the first one, DPLL receives a signal with phase difference ($\omega_i = 1$ MHz, $\theta_i = \pi/4$), DPLL response is shown in Fig. 11. In the second case input signal has both phase difference and frequency difference ($\omega_i = 1.01$ MHz, $\theta_i = \pi/4$), DPLL response is shown in Fig.12.

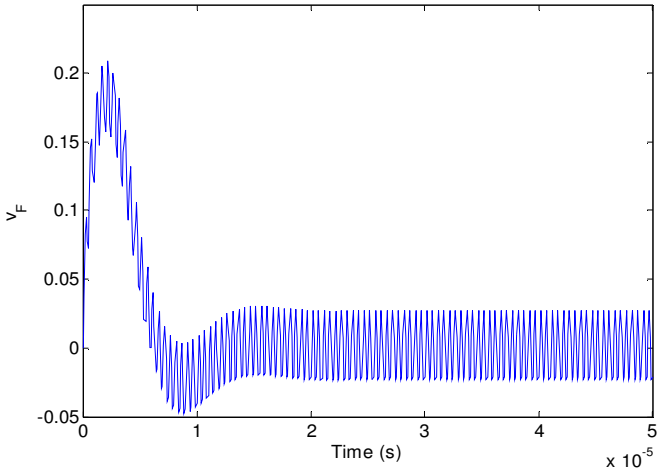

FIGURE 11: DPLL response in case of phase difference.

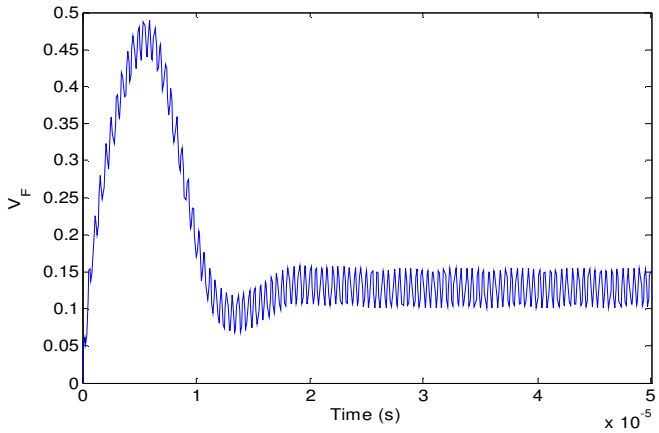

FIGURE 12: DPLL response in case of phase and frequency difference.

6.3 Proposed DPLL vs. Traditional DPLL

The main objective of this simulation is to compare the performance of the proposed DPLL with traditional DPLL; to be sure that replacing ROM with linear approximation did not affect the operation of DPLL. In this simulation both architectures have the same parameters.

$f_s = 10$ MHz, $k_m = 1$, $k_v = 1024$, $A_i = A_o = 1$, $\omega_{nco} = 1$ MHz, and $\theta_{nco} = 0$.

An input signal with $\omega_i = 1.02$ MHz and $\theta_i = \pi/2$. is applied to both architectures. Both responses are shown in Fig.13 which indicates that the performance of the proposed DPLL is not affected by the modified NCO. i.e. the ability of locking phase or frequency of the input signal is not affected. This means the proposed DPLL saves power consumption compared to traditional DPLL without affecting the performance of DPLL.

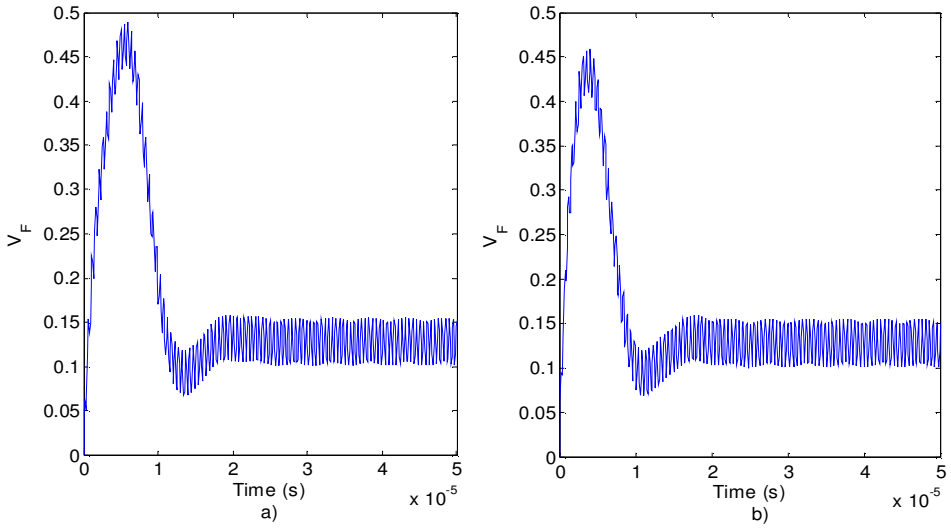

FIGURE 13: DPLL response in case of phase and frequency difference
a) Response of traditional DPLL. b) Response of proposed DPLL.

7. HARDWARE IMPLEMENTATION

Hardware implementation of modified NCO, and modified DPLL is done using VHDL code using Xilinx system generator Simulink tool [18:20]. The architecture of modified NCO is shown in Fig. 14.

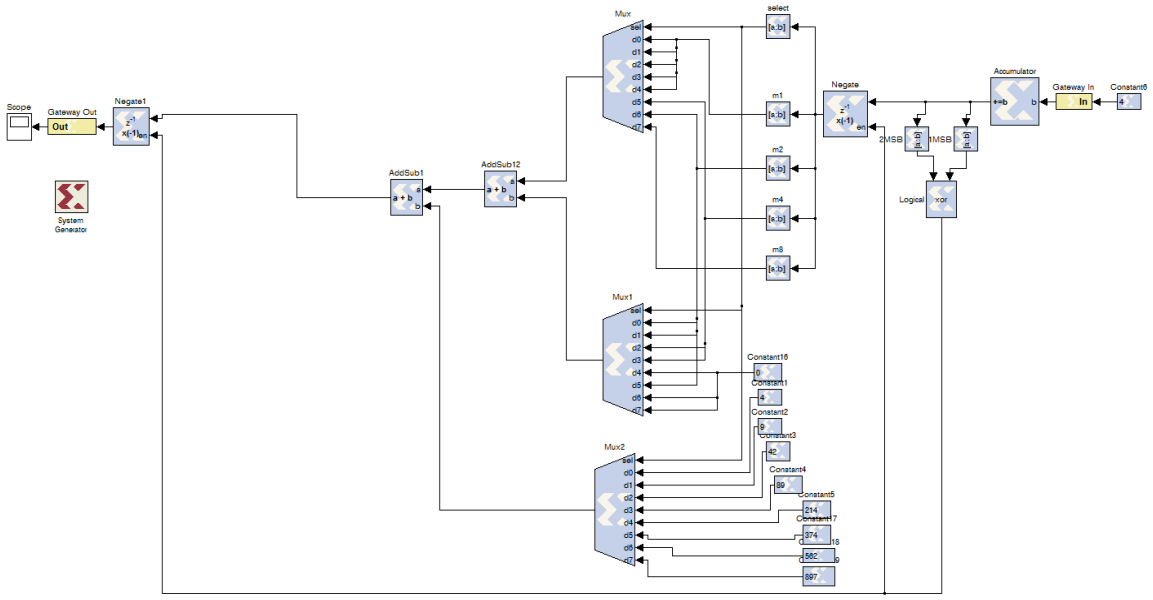

FIGURE 14: Modified NCO model

Implementation of linear segments requires slopes and constants. The slopes are chosen using MMSE as mentioned before and the slopes' accuracy is limited to a fraction four bits. m1 represents the full truncated output from the accumulator, m2 is half m1, m4 is half m2 (quarter m1) and m8 is half m4 (eighth m1). The first three MSBs generated from the accumulator are used to control three multiplexers. The first two multiplexers are forming the slope value, and the third multiplexer form the constant value. According to the selected signal, the linear equations are chosen through the multiplexers to form the complete linear equation.

The architecture of modified DPLL is shown in Fig. 15; the architecture uses the modified NCO instead of traditional NCO. The simulation is done at clock frequency 50 MHz. All signals are binary signals with different widths. The input signal is a binary signal of 8 bits width representing a sinusoidal signal at frequency 1 MHz. Fig.16 shows the simulation waveforms as an analog signal, the input signal (input1) of frequency 1 MHz is multiplied by the modified NCO signal (input2), and the output signal is passed through the digital filter. The final output shows that digital implementation agrees with the simulation waveform.

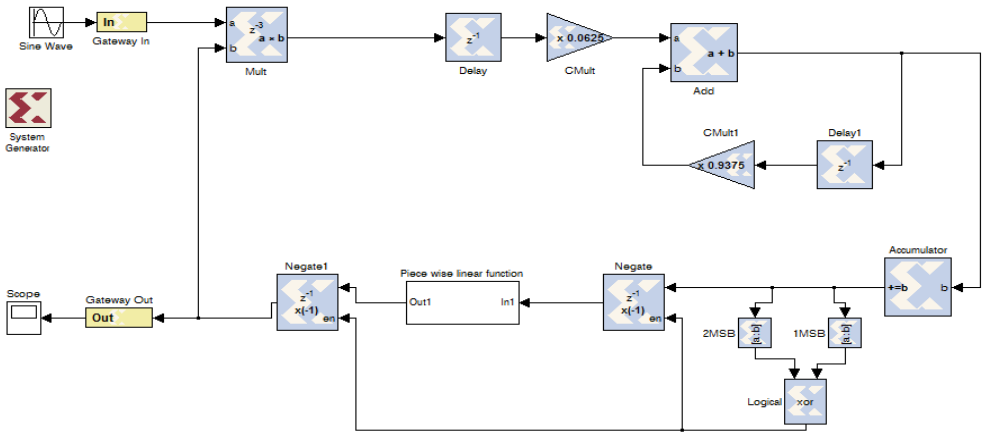

FIGURE 15: Modified DPLL model.

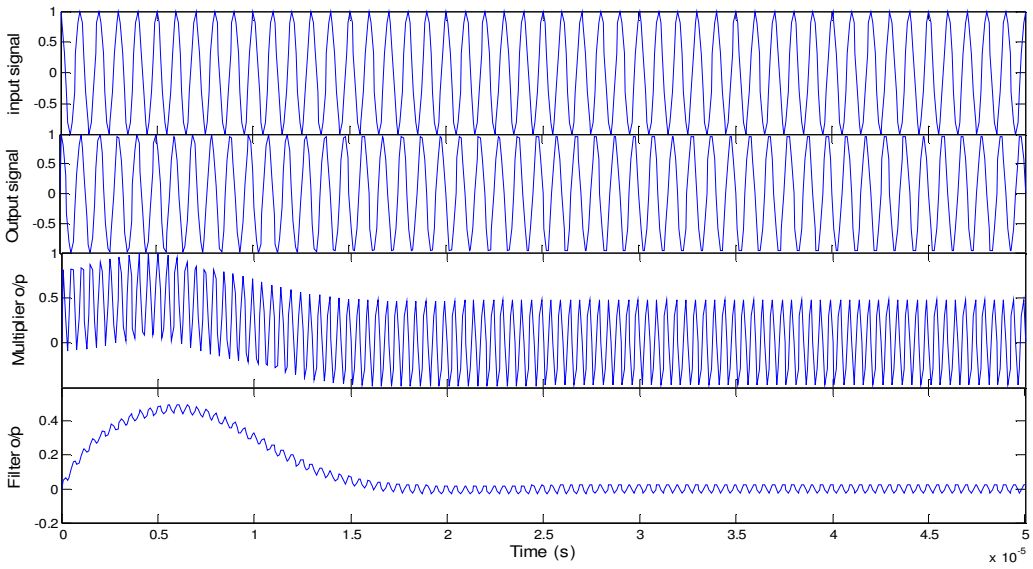

FIGURE 16: VHDL simulations of DPLL.

To recognize how much the modified NCO reduces the power consumption, logic elements and operation with faster frequency. A comparison between traditional NCO, which uses ROM block and modified NCO, is done by implementing both architectures on the same FPGA device (Xilinx-Spartan-3A DSP Xc3d3400a-5fg676). This comparison gives an idea of how much could be the improvements in power consumption, reduction in the occupied number of logic elements and faster frequency. As illustrated in Table1, the modified NCO reduces about 40% of total logic elements used in traditional NCO, and did not use memory bits, which leads to save the power consumption by about 25% and operation at a faster frequency about 1.8 times the speed of traditional NCO. Comparison is also done with the traditional DPLL (which uses a traditional NCO) and modified DPLL (which uses modified NCO). Table 2 shows the result of comparison; it is clear that the modified DPLL consumed less power, occupied less area and worked faster than the traditional DPLL, with no degradation in system operation such as locking range.

	Traditional NCO	**Modified NCO**
Slices	108	64
Flip Flops	21	17
Block RAMs	60	0
Look up table (LUT)	210	116
IOBs	24	24
Maximum Frequency	151.461 MHz	284.738 MHZ
Power consumption	0.264 Watt	0.197 Watt

TABLE 1: Implementation results comparison of NCO.

	Traditional DPLL	**Modified DPLL**
Slices	162	64
Flip Flops	37	17
Block RAMs	60	0
Look up table (LUT)	299	116
IOBs	24	24
Maximum Frequency	101.241 MHz	205.279 MHZ
Power consumption	0.314 Watt	0.237 Watt

TABLE 2: Implementation results comparison of DPLL.

8. CONCLUSION

Second order DPLL architecture has been described, analyzed and implemented to be suitable for any application. The problem of high power consumption of DPLL has been solved by replacing the traditional NCO (the main component in DPLL) with a modified ROM. The traditional NCO uses ROM, which results in high power consumption as well as slower operation. The proposed architecture reduces power consumption, area consumption and works at a higher frequency than the traditional one.

9. 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.

10. REFERENCES

[1] M. Dillinger. K. Madani, N. Alonistioti. Software Defined radio: Architectures, systems, and functions. John Willey & Sons Inc., 2003.

[2] T. J. Rouphael. RF and Digital Signal Processing For Software-Defined Radio: A Multi standard Multi-Mode Approch. John Willy & Sons Inc., 2008

[3] R. E. Best. Phase-Locked Loops: Design, Simulation, and Application. 6th ed, McGraw-Hill, 2007.

[4] S. Goldman. Phase Locked-Loop Engineering Hand Book of Integrated Circuit. Artech House Publishers, 2007

[5] B. Goldberg. Digital Frequency Synthesis Demystified: DDS and Fractional-N PLLs. Newnes ,1999.

[6] V.F. Kroupa, Ed. Direct Digital Frequency Synthesizers. IEEE Press,1999.

[7] V.F. Kroupa, V. Cizek, J. Stursa, H. Svandova. “Spurious signals in direct digital frequency synthesizers due to the phase truncation.” IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control, vol. 47, no. 5, pp. 1166-1172, September 2000.

[8] H.T. Nicholas III, H. Samueli and B. Kim. “The optimization of direct digital frequency synthesizer performance in the presence of finite word length effects,” in Proc. of the 42nd Annual Frequency Control Symposium, 1988, pp. 357-363.

[9] A. Yamagishi, M. Ishikawa, T. Tsukahara, and S. Date. "A 2-V, 2-GHz low-power direct digital frequency synthesizer chipset for wireless communication." IEEE Journal of Solid-State Circuits, vol. 33, pp.210-217, February 1998.

[10] A. M. Sodagar, G. R. Lahiji, “Mapping from phase to sine-amplitude in direct digital frequency synthesizers using parabolic approximation.” in IEEE Transactions on Circuits and Systems-II: Analog and Digital Signal Processing, vol. 47, pp. 1452-1457, December 2000.

[11] J.M.P. Langlois, D. Al-Khalili. “ROM size reduction with low processing cost for direct digital frequency synthesis,” in Proc. of the IEEE Pacific Rim Conference on Communications, Computers and Signal Processing, August 2001, pp. 287-290.

[12] L.A. Weaver, R.J. Kerr. “High resolution phase to sine amplitude conversion.” U.S. patent 4 905 177, Feb. 27,1990.

[13] A.M. Sodagar, G.R. Lahiji. “A novel architecture for ROM-less sine-output direct digital frequency synthesizers by using the 2nd-order parabolic approximation,” in Proc. of the 2000 IEEE/IEA International Frequency Control Symposium and Exhibition, 7-9 June 2000, pp. 284-289.

[14] K.I. Palomaki, J. Niitylahti. “Direct digital frequency synthesizer architecture based on Chebyshev approximation,” in Proc. of the 34th Asilomar Conference on Signals, Systems and Computers, Oct. 29th – Nov. 1st., 2000, pp. 1639-1643.

[15] J. Rudy. CMOS Integrated Analog-to-Digital and Digital-to-Analog Converters. Springer, 2003.

[16] J. G. Proakis, G. Dimitri, Manolakis. Digital Signal Processing. Prentice Hall,1996.

[17] Naresh K. Sinha. Linear Systems. John Wiley & Sons Inc.,1991.

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

[19] W.Y. Yang. Matlab/Simulink for Digital Communication. A-Jin, 2009.

[20] P. Chu. FPGA Prototyping by VHDL Examples: Xilinx Spartan-3 Version. Wiley-Interscience, 2008.