Military Technical College
Kobry El-Kobbah,
Cairo, Egypt

6th International Conference
on Electrical Engineering
ICEENG 2008

FPGA Implementation of a Direct Digital Synthesizer for Carrier Phase Synchronizer in Software Radio Receiver

By

Sherif Welsen Shaker* Salwa Hussien El Ramly** Khaled Ali Shehata***

Abstract:

More recently there has been a lot of discussion about the emergence of so-called Software Defined Radio (SDR). Due to its high reconfigurability, Field Programmable Gate Array technology (FPGA) can be viewed as an attractive option for implementing many of the tasks performed in SDR. Synchronization is one of the most complicated signal processing performed in SDR. This paper proposes an all-digital QPSK carrier phase synchronizer that is based on Phase-Locked Loop. The paper proposes the implementation of one of the basic block of the synchronizer which is the Numerically Controlled Oscillator (NCO) based Direct Digital Synthesizer (DDS) on Altera EPF10K70RC240-4 FPGA chip. A comparison between the simulation results and Hardware test of the DDS has been made. The used tools are FPGA Advantage Pro provided by Mentor Graphics and Quartus synthesizer provided by Altera.

Keywords:

Software defined radio, phase synchronization, FPGA and DDS.

* Modern Academy for Engineering and Technology, Cairo, Egypt.
** College of Engineering, Ain Shams University, Cairo, Egypt.
*** Arab Academy for Engineering and Technology, Cairo, Egypt.

1. Introduction:

When multiple technologies align in time to make it possible things that were only dreamed, a certain convergence occurs. A great convergence is occurring in radio communications through Digital Signal Processing (DSP) software to perform most radio functions at performance levels previously considered unattainable. Recently, there have been many discussions about the emergence of so-called Software Defined Radio (SDR) [1]. SDR is characterized by its flexibility so that modifying or replacing software programs can completely change its functionality. They provide a quick and easy way to upgrade into new modes and improve the performance without the need to replace any hardware. In the early 1990's Field Programmable Gate Arrays (FPGAs) have became a considerable option in digital communication hardware where they were often applied as a configurable logic cells to support memory controller tasks, complex state machines and bus interfacing [2]. Carrier phase synchronization may be considered as one of the most challenging signal-processing tasks performed in SDR. This paper proposes a technique for developing and modeling all-digital QPSK carrier phase synchronization and the implementation of its Direct Digital Synthesizer (DDS) module on FPGA chip.

2. QPSK Carrier Phase Synchronization:

The synchronizer proposed in this paper is based on the discrete-time PLL. The structure of a discrete-time PLL is essentially the same as that of continuous-time PLL. Continuous-time PLLs have three basic components: a phase detector voltage-controlled oscillator (VCO) and a loop filter [3]. The phase detector measures the difference between phases of the local oscillator and input carrier. This signal is fed to a loop filter that governs the response of the PLL to variations in the error signal. The loop filter is designed to track changes in the error signal. The discrete-time that mimics the continuous-time PLL is illustrated in figure 1. The loop filter uses a simple filter with a pole at z = 1. The discrete-time PLL uses DDS in place of the VCO [4]. A single-pole filter is used to integrate the DDS input to calculate the instantaneous phase. As illustrated in figure 2, and for the decision-directed QPSK carrier phase synchronizer, the in-phase and quadrature matched filter outputs, $x(kTs)$ and $y(kTs)$, are rotated by $-\phi'(k)$, to align the signal space projection ($x'(kTs)$, $y'(kTs)$) with the constellation points [5].

Computing the phase error is understood in geometric terms as illustrated in figure 3. The phase angles of the de-rotated matched filter output and that of the nearest constellation point are given by:

[redacted] (1)

The phase error for the k-th symbol is thus:

$e(k) = \phi_r(k) - \phi_d(k)$ (2)

[redacted] (3)

The phase error $e(k)$ is extracted from the point $(x'(kT_s), y'(kT_s))$ by computing the residual phase difference between $(x'(kT_s), y'(kT_s))$ and the nearest constellation point $(\hat{a}_0(k), \hat{a}_1(k))$ where for QPSK $\hat{a}_0(k) = \text{sgn}\{ x'(kT_s)\}$ and $\hat{a}_1(k) = \text{sgn}\{ y'(kT_s)\}$.

$$e(k) = \tan^{-1}\left\{\frac{y'(kT_s)}{x'(kT_s)}\right\} - \tan^{-1}\left\{\frac{\text{sgn}\{y'(k)\}}{\text{sgn}\{x'(k)\}}\right\} \quad (4)$$

cos (Ω[redacted](n)

Figure (1:) *Second order discrete-time PLL with proportional plus integrator.*

Figure (2): *Carrier phase synchronization using post-matched filter de-rotation operation*

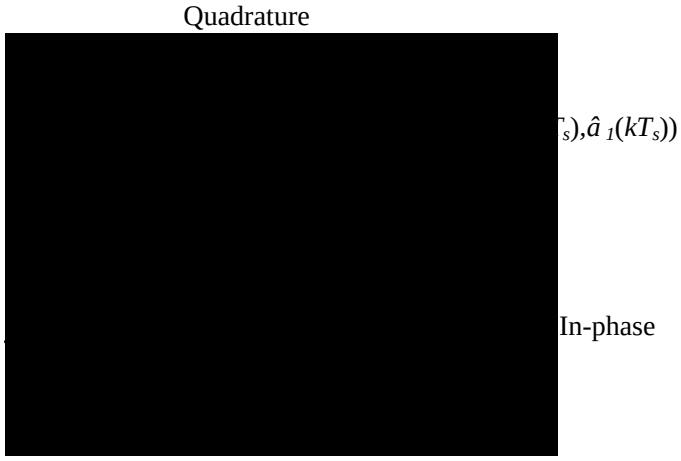

Figure (3): *Geometric representation of the phase error computation in a QPSK carrier PLL*

The phase detector requires two four-quadrant arctangent operations and a subtraction. A reduced complexity phase detector can be obtained by using the sine of the phase error in place of the phase error [5].

$$\sin(\phi_r(k) - \phi_d(k)) = \sin(\phi_r(k))\cos(\phi_d(k)) - \cos(\phi_r(k))\sin\phi_d(k) \quad (5)$$

$$= \frac{y'(kT_s)\hat{a}_0(k) - x'(kT_s)\hat{a}_1(k)}{\sqrt{x'^2(kT_s) + y'^2(kT_s)a_0(k)}\sqrt{a_0{}^2(k) + a_1{}^2(k)}} \quad (6)$$

To avoid the division suggested by the last equation, the numerator alone can be used as the error signal while the denominator terms are absorbed in to the phase detector gain. Thus $e(k) = y'(kT_s)\, a_0(k) - x'(kT_s)\, a_1(k)$.

3. NCO-based Direct Digital Synthesizer:

Direct digital synthesizers (DDS) based on Numerically Controlled Oscillators (NCO), are important components in many digital communication systems. Quadrature synthesizers are used for constructing digital down and up converters, demodulators, and implementing various types of modulation schemes, including PSK (Phase Shift Keying), FSK (Frequency Shift Keying), and MSK (Minimum Shift Keying). A common method for digitally generating a complex or real valued sinusoid employs a look-up table scheme. The look-up table stores samples of a sinusoid. A digital integrator is used to generate a suitable phase argument that is mapped by the look-up table to the desired output waveform [6].

A DDS based on an NCO takes a fixed frequency reference clock and generates a digital waveform of variable frequency. An NCO-based frequency synthesizer, as shown in figure 4, consists of the following pieces. The first piece is the NCO, which is just a big accumulator, which we describe as holding the "phase" of the output signal. As the NCO accumulator changes, the values represent a linear phase ramp. The second piece the quantizer, which is simply a slicer that accepts the high precision phase angle and generates a lower precision representation of the angle. This value is presented to the third piece which is the mapping Read Only Memory (ROM) that maps the most significant bits (MSBs) of the NCO into a sinusoid. In other words, the linear phase from the NCO is converted to a series of values representing a sinusoidal waveform.

4. FPGA Implementation of the NCO-based DDS:

The structure of the design is shown in figure 4. The design consists of the NCO, "phase accumulator" holding the phase of the output signal followed by a mapping ROM. The NCO itself is an 18-bit adder followed by a register. One of the 18-bit inputs of the adder is the phase increment input value of the DDS; the other input is a feedback of the

18-bit register. One other adder is used as a phase offset register for applying another 18-bit constant phase offset input to the phase slope computed in the phase accumulator register [7]. A 1K-byte memory representing the programmable memory is made so that the ROM can map the most 10 significant bits (MSBs) of the NCO output from the quantizer into a cosine. In other words, the linear phase from the NCO is converted to a series of values representing a cosine waveform. The ROM has 10 address lines input carrying the truncated phase address used for addressing the 1024 locations and 8 data output lines. The quantizer, accepts the high precision phase angle (18-bits) and generates a lower precision phase angle of size (10-bits). This value is represented to the address port of the mapping ROM [8]. A VHDL code has been written for describing each of the above-mentioned blocks using structural architecture.

Figure (4): *Structure of NCO-based DDS with 18-bit phase accumulator*

FPGA Advantage Pro. tool provided by Mentor Graphics has been used for the VHDL design description. A series of Full Adders (FAs) is used for representing each of the 18-bit adders. A data flow VHDL architecture is used for describing a look-up table which stores uniformly spaced samples of a cosine wave representing the mapping ROM. These samples represent a signal cycle of a length equal 2^{10} prototype complex cosine and correspond to specific values of the cosine's argument = n $(2\times\pi / 2^{10})$, where n is the time series sample index. The number of words in the ROM determines the phase quantization error while the number of bits in each word determines the amplitude quantization error.

Figure 5 provides the simulation of the NCO-based DDS. The ModelSim simulator provided by Mentor Graphics has been used for that purpose. The DDS input offset value is set at fixed value equal 1/16. Simulator is run and the DDS accumulator input "phase increment word" has been given different values. The DDS output waveform is plotted in an analog view setting the height at suitable value. The DDS output waveform shows how the DDS use the fixed frequency clock to generate a digital waveform of variable frequency and phase by changing the input phase increment word.

Quartus simulator provided by Altera has been used for the synthesization purposes to map the design to the FPGA target technology. The Design of the NCO-based DDS has been downloaded onto the ALTERA EPF10K70RC240-4 FPGA chip after pin assignment for all the inputs and outputs. The download process has been done using the ByteBlaster II download cable, which is a hardware interface to a standard parallel port.

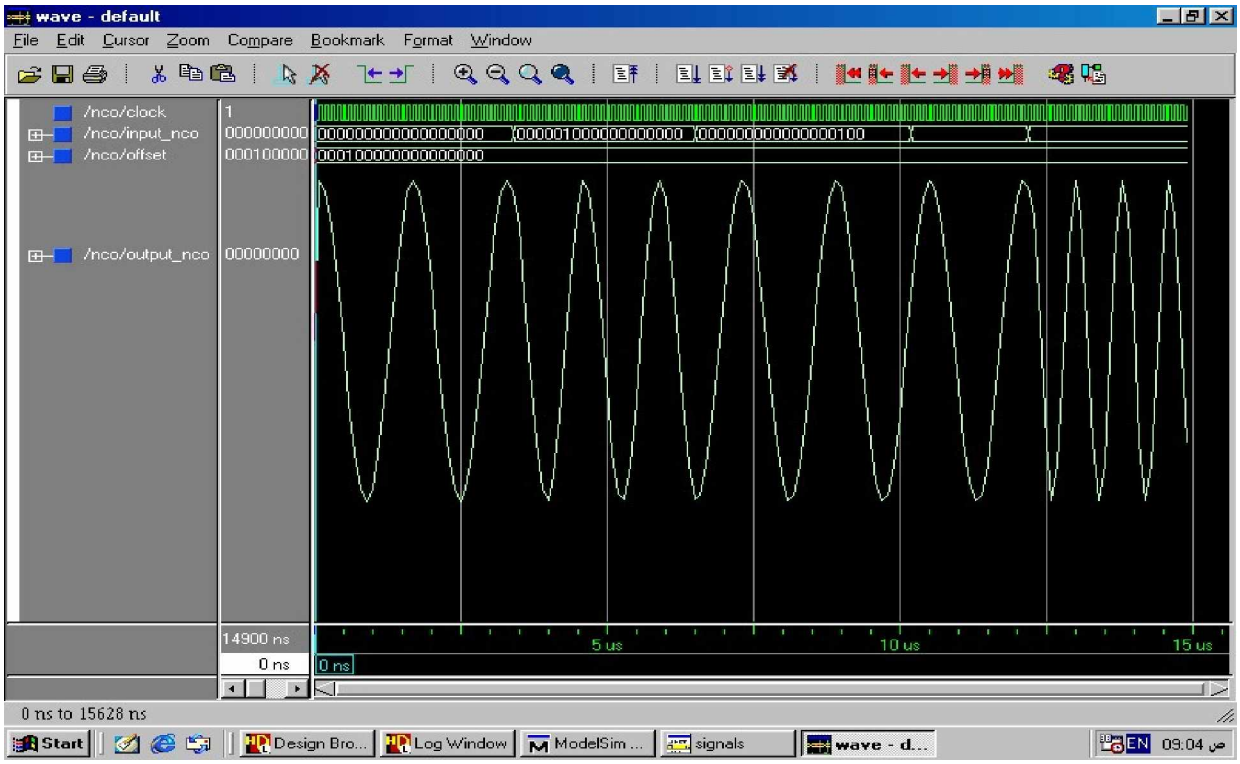

Figure (5): *NCO-based DDS waveform output*

5. Hardware testing and verification:

In the testing phase the logic analyzer has been used to plot the output waveforms for the downloaded design on the FPGA chip. Figure 6 shows the waveforms from the logic analyzer while figure 7 shows the waveforms from the ModelSim simulator, it is clear that both waveforms are identical.

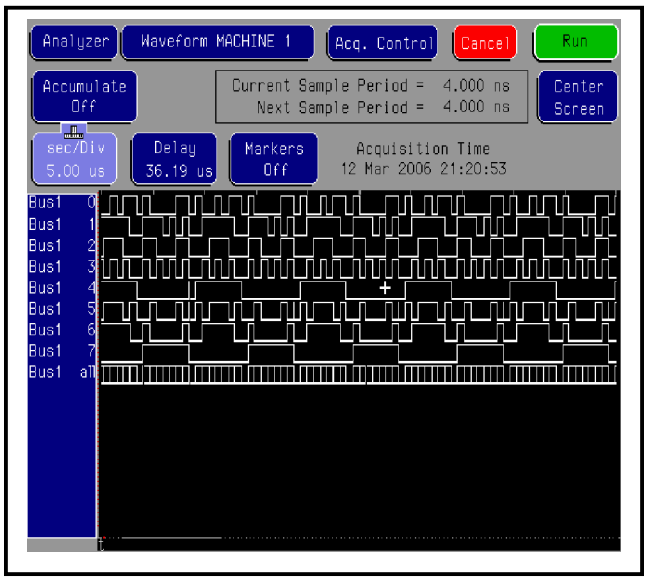

Figure (6): *Complete period waveforms from the logic analyzer*

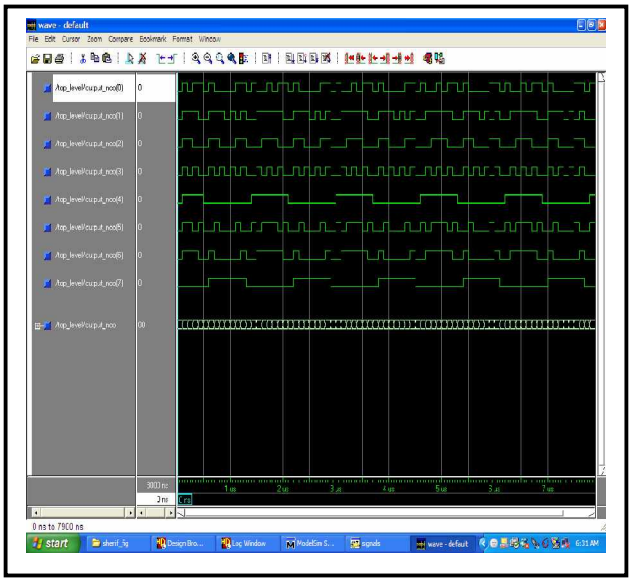

Figure (7): *Complete period waveforms from the logic analyzer*

6. Conclusions:

paper has demonstrated a practical approach for designing and implementation of a carrier phase synchronization module on FPGA chip. An all-digital, discrete time model of a QPSK carrier phase synchronizer has been proposed that uses an approach performs the phase compensation at the output of the matched filter. The Implementation of the DDS which is one of the basic building blocks in the proposed model has been

implemented on Altera EPF10K70RC240-4 FPGA chip. The design took about 96/3,744 (2%) of the total chip logic elements. The maximum operating frequency is 45MHz and it could be increased by using sophisticated FPGA families. Hardware testing and verification showed identical results with those from the simulation.

References:

[1] Joseph Mitola III, "*Software Radios*", IEEE Communications Magazine, volume: 33 no. 5, pp.24-25, May 1995.

[2] M. Cummings and S. Huruyama, "*FPGA in the Software Radio*", IEEE Communication Magazine, volume: 37, no. 2, pp. 108-112, February 1999.

[3] B. Sklar, *Digital communication fundamentals and applications*, Prentice Hall, Englenwood Cliffs, New Jersey, (2000).

[4] Joseph F. Gravey and Daniel Babitch, "*An exact spectrum analysis of a number controlled oscillator based synthesizers*", IEEE fourty fourth annual symposium on frequency control 1990.

[5] M. Rice, C. Dick and F. Harris, "*Maximum Likelihood Carrier Synchronization in FPGA-based Software Defined Radios*", ICASSP-2001, International Conference on Acoustics, Speech and Signal Processing, Salt Lake City, Utah, May 7-11, 2001.

[6] Jouko Vankka, "*Methods of mapping from phase to sin amplitude in direct digital synthesis*", IEEE trans. on ultrasonic, feroelectronics and frequency control, volume: 2, pp. 526-534, March 1997.

[7] Xilinx core generator system, DDS v5.0 <http://www.xilinx.com/products/logicore/coregen/index.htm>.

[8] Atalla Ibrahim Hashad, Khaled Ali Shehata, Mohamed Abu El-Dahab, and Sherif WelsenShaker, "*FPGA Implementation of Direct Digital Synthesizer for Software Radio Receiver*", First Workshop on VLSI Technology and EDA Tools, Arab Academy for Science and Technology, College of Engineering and Technology, Cairo, Egypt, December, 2005.