Implementation of High Frequency Direct Digital Frequency Synthesizer on FPGA

Prof. Akash I. Mecwan[1] (akash.mecwan@nirmauni.ac.in)
Prof. Vijay G. Savani[1] (vijay.savani@nirmauni.ac.in)
Prof. N. P. Gajjar[2] (nagendra.gajjar@nirmauni.ac.in)
[1]Assistant Professor, Institute of Technology, Nirma University
[2]Assosiate Professor, Institute of Technology, Nirma University

Abstract **-- *In* the modern era the software is taking charge of the hardware wherever possible. The traditional hardware radio is also replaced by the modern software defined radio. SDR makes the foundation of the cognitive radio domain. The direct digital synthesizer can be considered as the heart of any software radio system. This is also called Numerically Controlled Oscillator (NCO). The paper discusses the way to implement the Direct Digital Frequency Synthesizer (DDS). It has been implemented in VHDL for the Sparten 3E, xcv400 pq208 with speed grade of -4. The results for all designs are compared with respect to the area utilization.**

Index Terms **-- Direct Digital Synthesizer (DDS), LUT (Look Up Table), Quantizer.**

I. INTRODUCTION

There are a number of applications that need precision signal sources. Examples include impedance meters, network analyzers, various sources for test equipment, and even low-cost hobbyist function generators. The modern Software Defined Radio (SDR) requires sophisticated digital oscillator to carry out the synchronous operations. The development of semiconductor technology opened new possibilities to using digital techniques for solving this task. Various methods can be employed to create a digitally demanded waveform. DDS is one of the easiest and most reliable techniques to get the digitally generated waveform.

In its simplest form, a DDS can be implemented from a precision reference clock, an address counter, an LUT, and a D/A converter (figure 1). In this case, the digital amplitude information that corresponds to a complete cycle of a sine/cosine wave is stored in the LUT. The address counter steps through and accesses each of the LUT's memory locations and the contents (the equivalent sine amplitude words) are presented to a high-speed D/A converter.

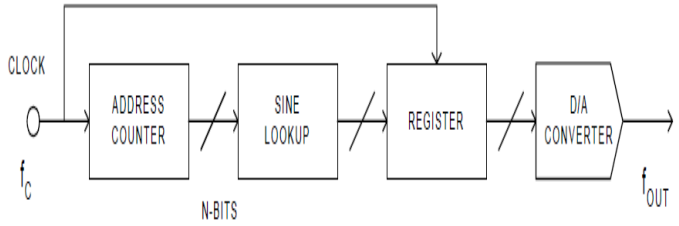

Figure. 1: Simplest form of DDS [1]

The D/A converter generates an analog sine wave in response to the digital input words from the LUT. The output frequency of this DDS implementation is dependent on:

- Frequency of the reference clock
- Sine wave step size that is programmed into the LUT.

While the analog output fidelity, jitter, and AC performance of this simplistic architecture can be quite good, it lacks tuning flexibility. The output frequency can only be changed by changing the frequency of the reference clock or by reprogramming the LUT. Neither of these options supports high-speed output frequency hopping. With the introduction of a phase accumulator function into the digital signal chain, this architecture becomes a numerically-controlled oscillator which is the core of a highly-flexible DDS device. [1]

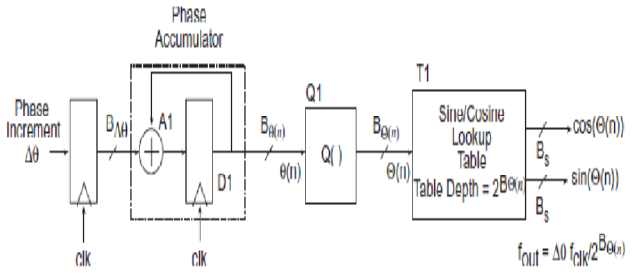

Figure. 2: Phase Accumulator based DDS [2]

A high-level view of the DDS Core is presented in figure 2. The integrator (components D1 and A1) computes a phase slope that is mapped to a sinusoid (possibly complex) by the look-up table T1. The quantizer Q1, which is simply a slicer, accepts the high-precision phase angle and generates a lower precision representation of the angle denoted as in the figure. This value is presented to the address port of a look-up table that performs the mapping from phase-space to time. The fidelity of a signal formed by recalling samples of a sinusoid from a look-up table is affected by both the phase and amplitude quantization of the process. The length and width of the look-up table affect the signal's phase angle resolution and the signal's amplitude resolution respectively. These resolution limits are equivalent to time base jitter and to amplitude quantization of the signal, and add spectral modulation lines and a white broad-band noise floor to the signal's spectrum. Direct digital synthesizers use an addressing scheme with an appropriate look-up table to form samples of an arbitrary frequency sinusoid. If an analog output is required, the DDS presents these samples to a digital-to-analog converter (DAC) and a low-pass filter to obtain an analog waveform with the specific frequency structure. Of course, the samples are also commonly used directly in the digital domain. The look-up table traditionally stores uniformly spaced samples of a cosine and a sine wave. These samples represent a single cycle of a length prototype complex sinusoid and correspond to specific values of the sinusoid's argument. [2]

II. IMPLEMENTATION

The DDS design was implemented using three techniques and the comparisons of all the three techniques is presented in this paper. The first technique is simple DDS.

In the simple DDS the sine table is generated using direct addressing using the case statement available in the Xilinx ISE suit. The input frequency defines the step size by which the accumulator is incremented. The output of accumulator is quantized to the specified number of bits. These bits represent the address which holds the data to be presented at the output. In this case the ISE tool takes the LUT as a multiplexer whose size is input address to the single output of number of bits required at the output.

The second method implemented was the generation of LUT using ROM table. The facility of making the ROM table is available in the ISE suit. The only difference in the first and second method is just the addressing scheme. The first method is direct addressing using "case" statement while the second is using the ROM table.

The third and the most efficient method is component based DDS. In this scheme all the internal block of the DDS are made as individulae and independed component. One by one the components are called to make a full DDS design. The architecture of component based DDS is as shown in figure 3.

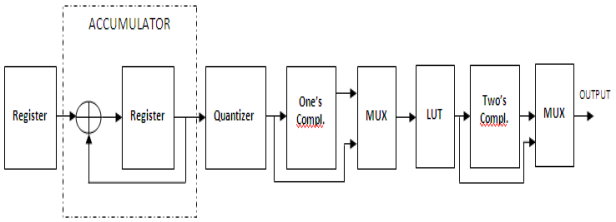

Figure 3. Component Based DDS

The size of LUT can be decreased by storing only one fourth of the sine wave. The full sine wave can be easily obtained by the one fourth of sine wave. The one's complement and tow's complement block are used to get the half sine wave and full sine wave respectively from the one fourth of sine wave. The multiplexers are used to select the quadrants of the sine wave whether first or second or third or fourth. The first two MSBs of quantizer's output is used as a selection line of first and second multiplexer respectively to create the full sine wave.

III. RESULTS

The comparison of all the three methods is given in table 1.

A: Simple DDS
B: ROM based DDS
C: Component based DDS

TABLE 1: COMPARISON OF VARIOUS DDS IMPLEMENTATION (SPARTEN 3E, XCV400, PQ208 WITH SPEED GRADE OF -4.)

Parameters	A	B	C
Number of Slice	423	268	61
Number of Slice flip flops	101	90	64
Number of 4 input LUT	817	505	88
Number of Bonded IOB	47	44	59
Number of GCLK	1	1	1

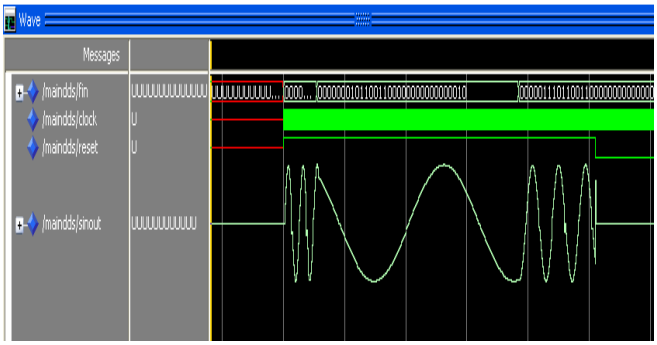

Figure 4. Output waveform for component based DDS.

IV. CONCLUSION

The direct conclusion of this paper is that the area requirement of the component based method is very less compare to both the other method. Another conclusion is that due to less number of components in the component based method the timing requirement is also very less. While synthesizing the code it is also observed that the synthesis of component based method is fast compare to both the other methods. One can also see that the ROM based method is also area utilizing compare to the simple DDS methods. This paper has only analyzed the area requirement of the design and the speed of the output. The other parameters like noise or quality of output will be same in all the cases as the method of obtaining the output remains same in all the three cases. The results are successfully simulated for 100 MHz of the clock frequency but the same design can work for 1 GHz also.

V. REFERENCES

[1] A technical tutorial on digital signal synthesis – Anlog Devices

[2] http://www.xilinx.com/support/documentation/application_note s/xapp154.pdf

[3] http://www2.itu.edu.tr/~pazarci/ddstutor.html

[4] http://www.digilentinc.com/

[5] H. T. Nicholas, H. Samueli, "An Analysis of the Output Spectrum of Direct Digital Frequency Synthesizers in the Presence of Phase-accumulator Truncation", in Proc. Annual Frequency Control Symp., 1987, pp.495-502.

[6] Soenkemehrgard, "Noise Spectra of Digital Sine Generation Using the Table-Lookup Method", IEEE Trans. Acoustics, Speech, and Signal Processing. Vol. ASSP-31, No.4, August, 1983.

[7] V. Reinhardt, K. Gould, K. McNab, and M. Bustamante, "A Short Survey of Frequency Synthesizer Techniques", in Proc. 40th Annual Frequency Control Symp., May 1986, pp. 355-365.

[8] B. Goldberg, “Digital Frequency Synthesis Demystified”. LLH technology Publishing, ISBN: 1-878707-47-7, 1999.

[9] Direct-Digital Frequency Synthesizer a basic tutorial, Osicom Technologies Inc., http://www2.itu.edu.tr/~pazarci/ddstutor.html

[10] J. Vankka, "Methods of Mapping from Phase to Sine Amplitude in Direct Digital Synthesis", IEEE Trans. Ultrasonics Ferroelectrics, and Frequency Control. vol. 44, pp. 526-534, March 1997.

[11] J. Tierney, C.M. Rader, and B. Gold, "A Digital frequency synthesizer”, IEEE Trans. Audio Electroacoust, vol. AU-19, pp. 48-57, 1971.

[12] Alin Grama, and Gabriel Muntean, "Direct Digital Frequency Synthesis Implemented on a FPGA Chip", in Proc. ISSE- 2006, Maienthal, Germany, 2006, pp. 92-97.

[13] A. M. Sodagar, G. R. Lahiji, "A Pipelined ROM-Less Architecture for Sine-Output Direct Digital Frequency Synthesizers Using the Second-Order Parabolic Approximation", IEEE Trans. Circuits and Systems - II: Analog and Digital Signal Processing. vol. 48, no. 9, September, 2001.

[14] J.M.P. Langlois, and D. Al-Khalili, "Phase to Sinusoid Amplitude Conversion Techniques for Direct Digital Frequency Synthesis", in IEE Proceedings Circuits Devices and Systems. Dec. 2004, pp. 519-528, John Rogers , Calvin Plett , Foster Dai, Integrated Circuit Design for High-Speed Frequency Synthesis (Artech House Microwave Library), Artech House, Inc., Norwood, MA, 2006