Journal of Applied Sciences

ISSN 1812-5654

science alert

ANSInet
an open access publisher
http://ansinet.com

Journal of Applied Sciences 12 (20): 2172-2177, 2012
ISSN 1812-5654 / DOI: 10.3923/jas.2012.2172.2177

Direct Digital Frequency Synthesizer Simulation and Design by means of Quartus-ModelSim

A.A. Alsharef, M.A. Mohd. Ali and H. Sanusi
Department of Electrical Electronic and Systems Engineering, Faculty of Engineering,
Universiti Kebangsaan Malaysia, 43600, Bangi, Malaysia

Abstract: A new full simulation, design and verification of a Direct Digital Frequency Synthesizer (DDFS), utilizing only one quarter of a given sinusoidal wave, are presented in this study. A reduction in the size of the LUT is accomplished as the new design requires storing only a quarter of the sine wave. The Register Transfer Level (RTL) and the Gate level is implemented by the Quartus II. The Quartus II will then invoke the ModelSim Altera software to simulate the output. The DDFS consists of three major models, mainly a Phase Accumulator (PA), a Phase Register and a Look Up Table (LUT). All of the mentioned models are realized by a Verilog code. The spurious free dynamic range is achieved with a value of -73 dB using a 16 bit phase accumulator. The proposed design is verified through the application of different input frequencies and obtained results showed that output frequency is directly proportional to the tuning input frequency.

Key words: Field programmable gate array, direct digital synthesis, read only memory lockup table, digital phase locked loop

INTRODUCTION

Direct Digital Frequency Synthesis (DDFS) is a multi step process of generating sinusoidal analogue waveforms. DDFS has a wide application in the modern communication era such as radio receivers, mobile telephones, radiotelephones, walkie-talkies, CB radios, satellite receivers and none the less GPS systems (Grayver and Daneshrad, 1998). The focus of this paper is on the design, analysis and simulation of DDFS using Quartus-ModelSim. Traditional designs found in literature of high bandwidth frequency synthesizers make use of a Phase Locked-Loop (PLL) approach. The PLL offers very good wide tuning bandwidth due to the use of a programmable divider as compared to DDFS approach (Sunderland *et al*., 1984). On the other hand, DDFS provides many significant advantages such as fast settling time, sub-Hertz frequency resolution, continuous-phase switching response and low phase noise (Hegazi *et al*., 1998; Mortezapour and Lee, 1999). One key design parameter of the DDFS is a Look Up Table (LUT). The response time, the power consumption and the size of the DDFS are factors that depend on the size of the LUT. In addition to that, the resolution and the size of DDFS are also dependable on the size of the phase accumulator (Bramble, 1981). In this method, the phase/amplitude of an analogue sine wave cycle is sampled with an equal phase intervals to obtain a discrete sequence of a single sine wave period thus quantifying its analog amplitude. Consequently, a single period of a sine wave is converted into a binary sequence. This binary sequence represents the quantized amplitude of the sine wave to be stored in a ROM. Therefore, the content of the ROM matches the phase sampling for a single full cycle. The main purpose of this study is to propose a new technique through the implementation of a quarter of the input analogue waveform that results in the size reduction of the LUT unit. The design and the implementation of DDFS systems supported by VERILOG is simple and less demanding thus progressively became the trend towards DDFS design (Yi-Yuan and Xue-Jun, 2011). Altera's Quartus II, an appropriate platform for VERILOG compilation and synthesis, offers a strong simulation tool therefore, can extremely simplify the overall design procedure.

This study begins by introducing the basic architecture of DDFS with a review of the available design methods. Followed by that, an extensive developmental procedure of the proposed design using the Quartus II environment is presented.

BASIC ARCHITECTURE OF DIRECT DIGITAL SYNTHESIZERS

The general structure of any given Direct Digital Synthesizer (DDS) may seem complex. DDS major components include the phase accumulator, the lookup table and the phase resistor. In this brief description the

Corresponding Author: A.A. Alsharef, Department of Electrical Electronic and Systems Engineering, Faculty of Engineering, Universiti Kebangsaan Malaysia, 43600, Bangi, Malaysia

last component, the phase resistor, is the easiest part to start with as to simplify the flow of the proposed scheme. The original task of DDS's is to obtain an output signal in the form of a sinusoidal wave given a specified reference frequency. Since the output of the DDS is in a form of digital signals, a Digital to Analogue Converter (DAC) is needed. This means that the structure of any DDS should contain a DAC. The DAC output should pass through a Low Pass Filter (LPF), an anti-aliasing filter, to suppress any reproduced images of the output spectrum. To obtain an output sinusoidal signal, a sequence of input sampling sinusoidal signals is applied to the DAC. Obtaining a sinusoidal signal through a digital system is not direct and may need different orientation for implementation such as a lookup table method. A Look up Table (coding table) is most often placed in a ROM. A code which is fed to the address inputs of the ROM, is the argument x of the sinusoidal function sin(x) while the obtained output code from the ROM represents the calculated value of sin(x). Forming a linear time-varying sequence of codes is simple and can be implemented via a binary counter. Therefore, the simplest type of DDS is a binary counter that generates the address of the ROM. The ROM would then comprise a table of one period of the sinusoidal function in a form of digital codes. These codes are then sent to the DAC that regenerates the analogue sinusoidal output signal then filtered by LPF to generate the main output of the DDS as shown in Fig. 1.

To adjust the frequency of the DDS output there is a need for a frequency divider with variable division factor that receives a clock signal from a reference generator. This structure of DDS has some drawbacks. The main drawback is the poor regenerative frequency process. Indeed, since the clock frequency undergoes a division by an integer, steps of adjustments became variable, therefore, the smaller the division ratio the greater the value of the step adjustment. This step becomes unsatisfied at a small division factor. In addition to that, when adjusting the output frequency, the sampling frequency will also change. This complicates the filtration process of the output and leads to a suboptimal use of high-speed DACs. Regardless of the weaknesses described above, the structure can be developed by replacing the counter of the address memory with other digital devices, called accumulators (Vankka and Halonen, 2001). For each cycle reboot, the accumulator adds a specific value, called a tuning word, with its feedback output as shows in Fig. 2. The content of the registers increases linearly with time with a constant step size. If the accumulator is used to generate the phase code it is called the phase accumulator. The output code of the phase accumulator is the code of the instantaneous phase of the output sinusoidal signal.

The tuning word represents a constant increment per cycle value added continuously to the phase accumulator. The faster the phase varies in time, the greater is the

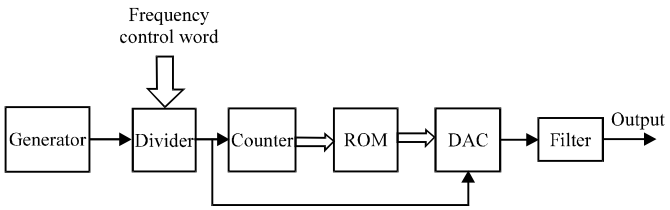

Fig. 1: Simplest direct digital synthesizer, ROM: Read only memory, DAC: Digital to analogue converter

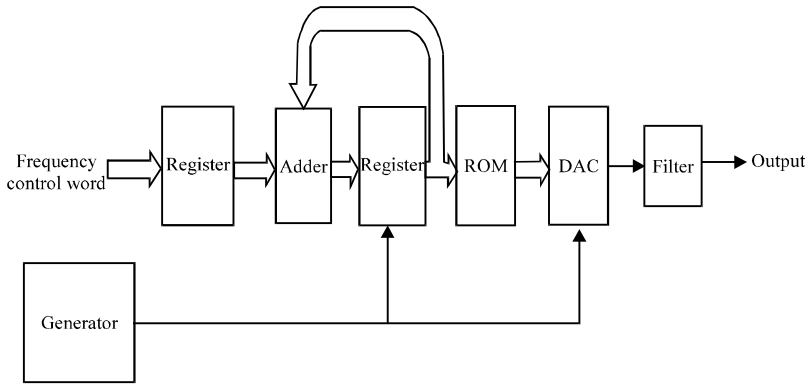

Fig. 2: Direct digital synthesizer based on the accumulator, ROM: Read only memory, DAC: Digital to analog converter

frequency of the generated signal. Therefore, the value of the phase increment is actually a frequency tuning word. Indeed, if the phase increment is equal to unity, the accumulator is no different from the a regular binary counter. But if the phase increment is equal to, for example, two, then the code phase will change twice as fast. In this case, the DAC code will come up with the same frequency but will not be a neighbor. If the frequency of a generated signal is twice as large, the sampling rate will stay unchanged. The phase accumulator is operated during a single stage of overflow, within arithmetic modulo 2^n. This stage of overflow corresponds to a sinusoidal wave with a period of 2π. In other words; the frequency of the overflow of the phase accumulator is the frequency of the output signal. This frequency is determined by the formula (Sharma and Upadhyaya, 2010):

$$F_{out} = M \times \frac{F_{clk}}{2_n} \quad (1)$$

where, F_{out} is the output frequency, F_{clk} is the clock frequency, M is the code rate; and n is the bit of phase accumulator. In essence, the clock frequency is divided by a number which is determined by the code frequency and the phase accumulator word length. At the same time the frequency step is equal to:

$$D \times F_{out} = D \times M \times \frac{F_{clk}}{2_n} \quad (2)$$

From this formula, it may be concluded that if the word length n is increased, the frequency step size will reduce with no special restrictions. For example, if the bit accumulator is 32 bits and the clock frequency is 50 MHz, the frequency resolution is in the order of 0.01 Hz. This means increasing the bit size (number of bits) of the phase accumulator does not require increasing the bit address of the ROM (Nicholas *et al*., 1988). The address may use only the necessary number of significant bits of the phase code. To reduce the size of the ROM, the sinusoidal symmetrical properties of the function may be used (Bellaouar *et al*., 2000). Therefore, only a quarter of the sinusoidal wave is needed to be stored in the ROM, As a result, a more complicated logic of forming the address (address forming) is required. Thus, in DDS the phase accumulator generates a sequence of codes of the instantaneous phase signal that varies linearly. The change in the rate of phase gives the frequency code. Further, by using a ROM, the linear change of the phase is converted to an output varying sinusoidal signal. Samples are then fed to the Digital to Analogue Converter (DAC), therefore the output of the DAC is a step like sinusoidal signal. Afterwards the step like wave is filtered by a LPF and its output represents the desired sinusoidal wave.

SYSTEM LEVEL SIMULATION

In order to carry out the system level modelling and simulation of . a the DDFS a Quarts-ModelSim is utilized. This would help in visualizing the functionality and the flow sequence from the input part to the output part. The model of DDS consists of a phase register, a Phase Accumulator (PA) and a Look Up Table (LUT). Figure 3 shows the RTL level schematic of the DDS. Each model of

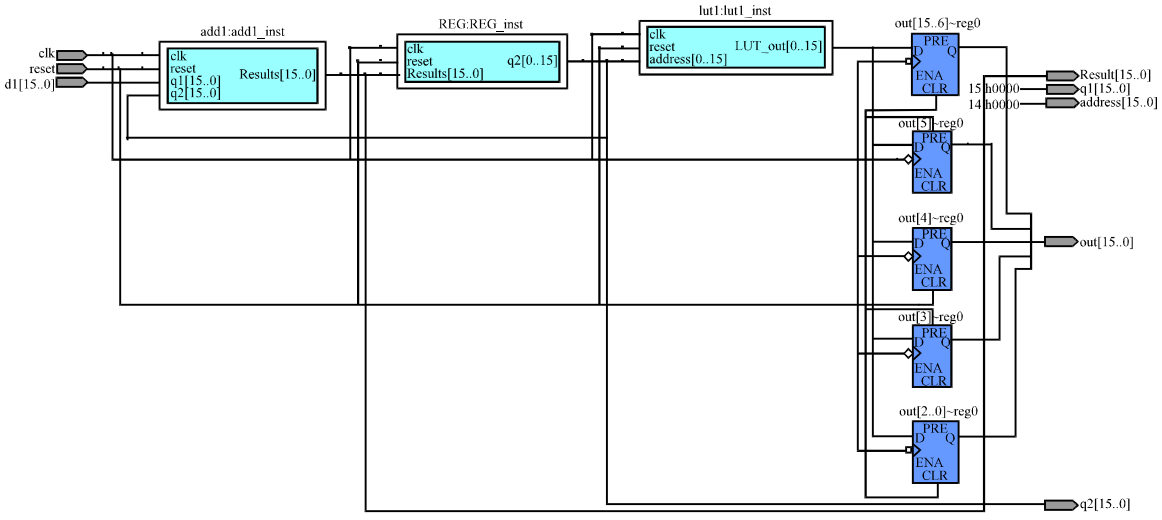

Fig. 3: Quarts-ModelSim model of direct digital synthesizer

them is built up using a Verilog then are assembled in the top level model. Thus, it is possible to modify each model according to the predetermined design requirements. The phase register is implemented using a 16 bit register. The phase accumulator is implemented via a register along with an adder and a feedback loop. The LUT is implemented using a ROM. The size of the memory will decrease as only one quarter of the sinusoidal wave is stored as given in Fig. 3. The model flow would be as follows. The content of the phase register is added to the phase accumulator at each clock pulse. The phase accumulator produces the phase value of the output sinusoidal wave. The phase value of the PA provides the address for the LUT. The LUT contains sample values of the quarter of the sinusoidal wave during one cycle overflow of the phase accumulator. The data is generated by the MATLAB. The calculation of a single full period sine wave would be achieved as follows. The full period of the sine wave is divided into quarters with an index number given as $n\times\sigma/2$, where n = 1-4. The description would be given for the first . half halve of the full period mainly for n = 1 and 2. For n = 1, the program will read the contents of the ROM normally during the first quarter until the phase argument is $\sigma/2$. The address bus is normally incremented during this part from 0 to the maximum value. During the second quarter where n = 2, the address of the ROM is decremented down to the start point. Data are read from the ROM opposite to the first stage. For the next half period, where n = 3 and 4, the same procedure would be followed given that the contents of the ROM would be first multiplied by -1. As a result, a full sine wave period is generated using the data of only a quarter of the wave and utilizing a special address formatting procedure. Such procedure has the advantage of saving the total ROM size down to 25% of the actual needed size. The overflow rate of the phase accumulator depends on the bit size of the phase accumulator and the frequency tuning word. The larger the size of the frequency tuning word, the faster the PA overflows. The output frequency of the direct digital synthesizer is directly proportional to the frequency tuning word. Therefore, the larger the frequency tuning word, the faster the PA overflows and the higher the output frequency.

RESULTS AND DISCUSSION

Examples are given to clarify the above suggested method of the new designed DDS. Assume for example a reference frequency (clock) of 16.6 MHz with a tuning word of 10. The time reading of an oscilloscope output would read a full period of 0.863864×10^5 psec (picosecond) which represents a frequency of 115758 Hz as shown in Fig. 4. By changing the tuning word to be 20 and keeping the same reference frequency will result in a time reading of the oscilloscope of 0.456932×10^5 psec. This represents a frequency of 218850.94 Hz as shown in Fig. 5. Examples are also given to clarify the gate level simulation of the new designed DDS. Assume for example a reference frequency (clock) of 16.6 MHz with a tuning word of 10. The oscilloscope time reading of one full period output would be 0.863864×10^5 psec which represents a frequency of 115758 Hz as shown in Fig. 6. By altering the tuning word to be 20 and keeping the same reference frequency, the time reading of the oscilloscope would be 0.456932×10^5 psec. This represents a frequency of 218850.94 Hz as shown in Fig. 7. As a conclusion the output frequency is directly proportional to the input frequency tuning word.

Measurement results: A real time result is carried out by a DE2 board Altera Cyclone®-II 2C35 FPGA. A summary of the devices usage is given in Table 1. The result is

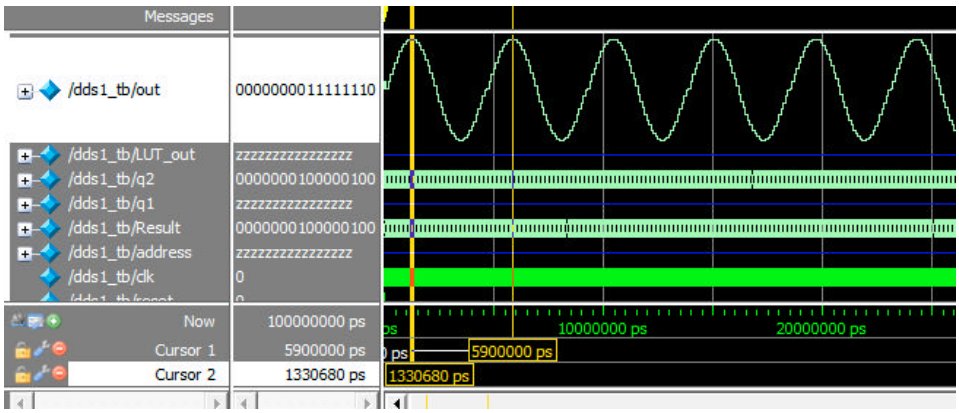

Fig. 4: RTL Output for tuning word is equal to 10

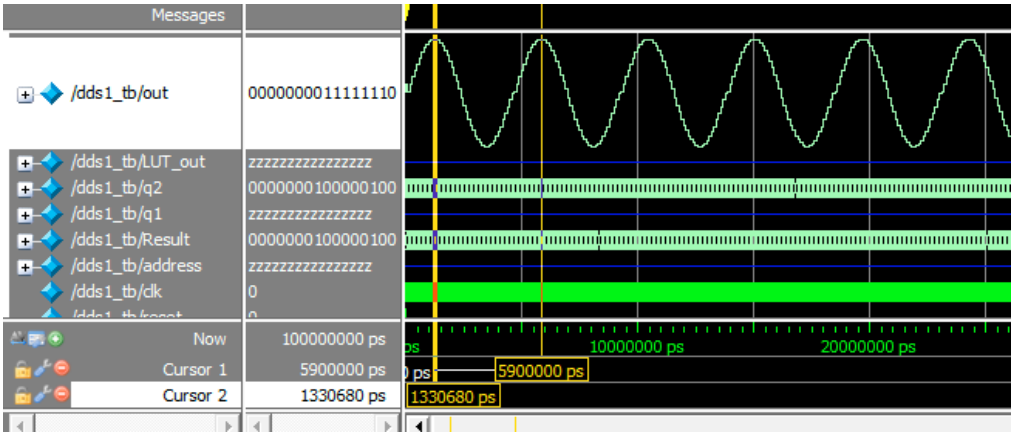

Fig. 5: RTL Output for tuning word is equal to 20

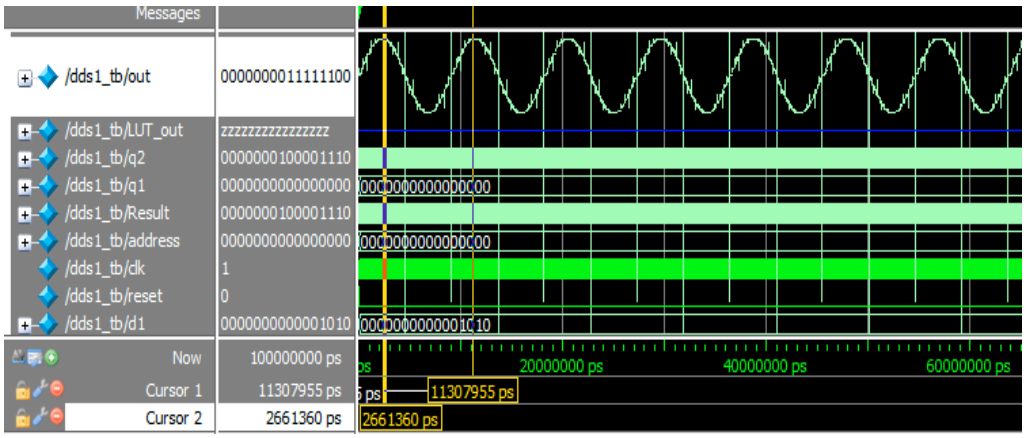

Fig. 6: GLS Output for tuning word is equal to 10

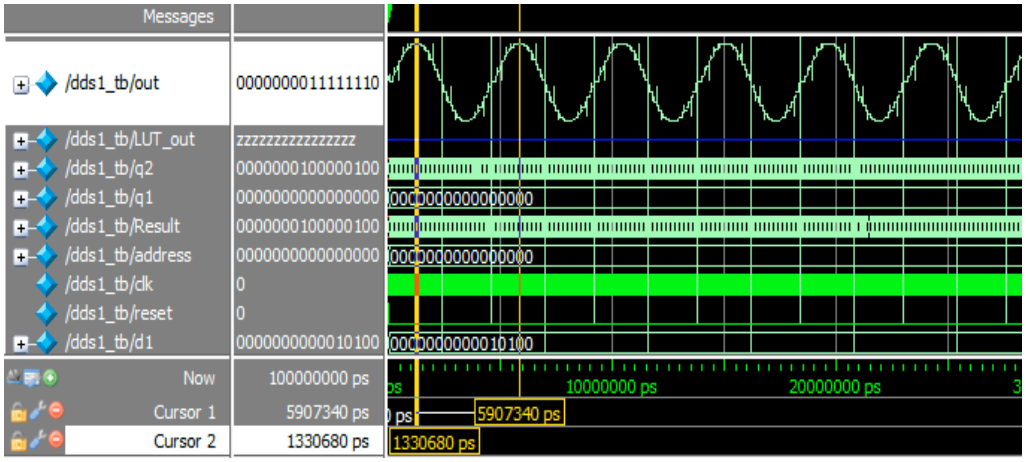

Fig. 7: GLS Output for tuning word is equal to 20

Table 1: Device utilization summary

Logic utilization	Used	Available	Utilization (%)
Total combination function	33,216	244	<1
Dedicated logic registers	33,216	42	<1
Total pins	98	475	21
Total element logic	33,216	258	<1
Total memory bits	0	483,840	0
Embedded multiplier 9-bit element	0	70	0
Total PLLs	0	4	0

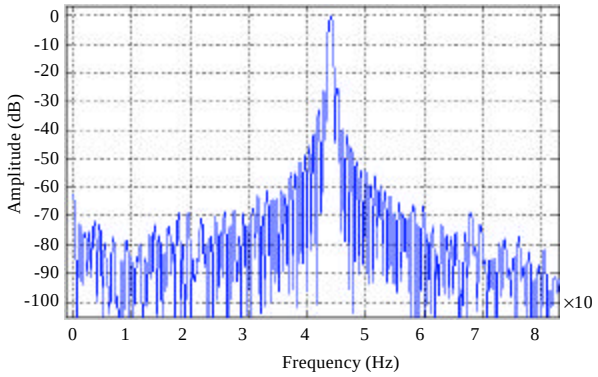

Fig. 8: Measuring power spectrum

measured using a logic analyzer. The clock is taken from the pattern generator of the logic analyzer. The data are saved in a form of a text file in the logic analyzer. Using a MATLAB routine, the power spectrum is plotted as shown in Fig. 8. The calculated spurious free dynamic range is about -73 dB using a 16 bit phase accumulator which is Adequate compared with other studies using a 32 bit phase accumulator (Vankka *et al*., 1998; Wang *et al*., 2010).

CONCLUSION

This study demonstrates a new design and the simulation of a direct digital frequency synthesizer. The DDFS's digital part includes a phase register, a phase accumulator (PA) and a ROM. The design is created using Verilog HDL.

The RTL -level modelling and simulation of a DDFS is implemented using Quarts-ModelSim. The phase register is achieved via a register. A phase accumulator is achieved by a register along with an adder and a feedback loop. The LUT is implemented by a Read Only Memory (ROM). In this proposed process, the size of the ROM is reduced. This is done by saving only a quarter of the sinusoidal signal in the LUT. The required address formatting for such design is also fully described. The spurious free dynamic range by the value of -73 dB using a 16 bit phase accumulator is obtained. The result of the simulation shows that the output frequency is directly proportional with the frequency tuning word.

REFERENCES

Bellaouar, A., M.S. O'brecht, A.M. Fahim and M.I. Elmasry, 2000. Low-power direct digital frequency synthesis for wireless communications. IEEE J. Solid-State Circ., 35: 385-390.

Bramble, A.L., 1981. Direct digital frequency synthesis. Proceedings of the 35th Annual Frequency Control Symposium, May 27-29, 1981, Philadelphia, PA., USA., pp: 406-414.

Grayver, E. and B. Daneshrad, 1998. Direct digital frequency synthesis using a modified CORDIC. Proceedings of the IEEE International Symposium on Circuits and Systems, Volume 5, May 31-June 3, 1998, Monterey, CA., USA., pp: 241-244.

Hegazi, E.M., H.F. Ragaie, H. Haddara and H. Ghali, 1998. A new direct digital frequency synthesizer architecture for mobile transceivers. Proceedings of the IEEE International Symposium on Circuits and Systems, Volume 3, May 31-June 3, 1998, Monterey, CA., USA., pp: 647-650.

Mortezapour, S. and E.K.F. Lee, 1999. Design of low-power ROM-less direct digital frequency synthesizer using nonlinear digital-to-analog converter. IEEE J. Solid-State Circ., 34: 1350-1359.

Nicholas, H., H. Samueli and B. Kim, 1988. The optimization of DDFS performance in the presence of finite word length effects. Proceedings of the 42nd Annual Frequency Control Symposium, June 1-3, 1988, Baltimore, MD., pp: 357-363.

Sharma, R.K. and G. Upadhyaya, 2010. Memory reduced and fast DDS using FPGA. Int. J. Comput. Theory Eng., 2: 1793-8201.

Sunderland, D.A., R.A. Strauch, S.S. Wharfield, H.T. Peterson and C.R. Cole, 1984. CMOS/SOS frequency synthesizer LSI circuit for spread spectrum communications. IEEE J. Solid-State Circ., 19: 497-506.

Vankka, J. and K. Halonen, 2001. Direct digital synthesizers-Theory, design and application. Kluwer Academic Publishers, Boston.

Vankka, J., M. Waltari, M. Kosunen and K.A.I. Halonen, 1998. A direct digital synthesizer with an on-chip D/A-converter. IEEE J. Solid-State Circ., 33: 218-227.

Wang, Q., S. He and Z. Zhong, 2010. Design and simulation of an optimized DDS. Proceedings of the 6th International Conference on Wireless Communications Networking and Mobile Computing, September 23-25, 2010, Chengdu, China, pp: 1-3.

Yi-Yuan, F. and C. Xue-Jun, 2011. Design and simulation of DDS based on quartus II. Proceedings of the IEEE International Conference on Computer Science and Automation Engineering, Volume 2, June 10-12, 2011, Shanghai, China, pp: 357-360.