Design and Simulation of Direct Digital Synthesizer for Wireless Applications

Monika kushwaha^{1*}, U. M Gokhale²

¹Department of Electronics & Telecommunication, GHRIETW, Nagpur university, India

²Department of Electronics & Telecommunication, GHRIETW, Nagpur university, India

¹monika.kushwaha91@gmail.com; ²umgokhale@gmail.com

***Corresponding Author: -**

Email: monika.kushwaha91@gmail.com

Abstract: -

Direct Digital Synthesis is an improved method of producing analog waveforms where the generation is done completely in the digital sphere. In this paper, a design is presented for a Direct Digital Synthesizer (DDS) which generates multiple waveforms. The architecture is based on a 32-bit phase accumulator and a look-up table (LUT) as phase to amplitude converter. This design will be simulated in Xilinx. This DDS will offer qualities like fast switching, good frequency resolution and good stability. This DDS can become highly portable if it is designed on a Field Programmable Gate Array (FPGA).

Keywords: -*Direct Digital Synthesizer (DDS), NCO, look-up table, phase accumulator, phase to amplitude converter*

Distributed under Creative Commons CC BY-NC 4.0 OPEN ACCESS

I. INTRODUCTION

Waveform generation is highly important in high-speed wireless applications such as radar, telecommunication, instrumentation and many other fields. The common waveform generators till early 1970 were the Direct analog synthesizers (DAS). They generated frequency by mixing frequencies from different crystal and use their harmonics. The most common analog waveform synthesizer is the analog Phase locked loop (PLL) which uses a phase detector and a voltage-controlled oscillator. These analog waveform synthesizers had slow switched time, poor frequency resolution, less flexibility and were bulky.

DDS is the new improved method for waveform generation in the radio-frequency zone compared to its analog counterparts. Much work has been done on DDS since the last 40 years. Early DDS designs were capable of producing output waveforms with frequencies upto 3 MHz and spurs of -40 dB whereas modern DDS have gone upto producing 400 MHz as output frequency with less spurious content of about -60 dB to -80 dB [7]. Direct digital synthesizers have entered the market in the form of ASICs by manufacturers like Analog devices, Novatech Instrument, SITAR- Bangalore (India).

Now a day's FPGAs are very popular for implementing digital circuits because of less time to market, ease of programming at the consumer level compared to ASIC. So, designing the DDS on FPGA makes our device highly efficient. Also till now the main focus was on the generation of sine wave only but multiple waveform generation will increase the area of application. VHDL a Hardware Description language is generally used for design and simulation. In the second section, we will present the literature review about the work done of DDS on FPGA. In the third section, we will present a basic signal generation method. In the fourth section we will propose a frequency independent DDS. In the fifth section we will present the design of sub-modules required to build the DDS. Sixth section will cover the simulation results and in the final section, we will present a conclusion.

II. LITERATURE REVIEW

Matt Bergeron and Alan N. Willson [2] proposed an architecture that used an angle rotation algorithm based on multipliers. This algorithm is used for phase to amplitude conversion. The architecture used 32-bit phase accumulator and the clock frequency applied is 1GHz which generates an output frequency of 400 MHz on Xilinx Virtex 7 FPGA dissipating only 54.9mW. Sunny Raj Dommaraju, Saiyu Ren and George Y.Lee [3] proposed an architecture that replaced the large power hungry Read Only Memory (ROM) as phase to amplitude converter by a set of band pass finite impulse response filters. The architecture used 16-bit phase accumulator. The clock frequency applied is 200 MHz which generates an output frequency of 71 MHz on Xilinx Virtex 6 FPGA. Xiaochu Wang and Qiujun Mei [4] proposed an architecture that used pipelined accumulator for increasing speed and a ROM look-up table based on Co-ordinate Rotational Digital Computer (CORDIC) algorithm to achieve compression. The clock frequency applied is 100 MHz which generates an output frequency of 1.56 MHz on Altera EP3C5M164C7 FPGA. Qing Wang, Songbai He and Ziming Zhong [5] proposed an architecture with pipelined accumulator with a compression of ROM look-up table to one-fourth. Phase jitter is injected to eliminate truncation error which may result due to consideration of higher 16 bits for address. This architecture uses 32-bit phase accumulator. The clock frequency applied is 92.16 MHz which generates an output frequency of 22.16 MHz on Altera EP2S60F672C5 FPGA. Muhammad Asim Butt and Shahid Masud [6] proposed an architecture that has employed the Digital clock manager of the clock of the system. Here also ROM compression is achieved. A single bit digital to analog converter has been employed internally. The clock frequency applied is 50 MHz which generates an output frequency of 59.60 Hz on Xilinx Spartan FPGA.

As we see all the designs have employed techniques to improve one or the other parameter. But they have some drawbacks too. They all have focused towards generation of sine wave only thus minimizing its applications. The techniques employed are also complex. So, our proposed design works towards generation of multiple waveforms to increase the device applications. Also, our aim is to generate much higher frequency using a low-cost version of FPGA making our prototype model best choice for actual implementation of the DDS from market point of view.

III. BASIC SIGNAL GENERATION

The basic process of signal generation is first performed and simulated. In this method, a sine package is created which contains a LUT. The LUT has been compressed for optimizing the design. It stores only 1/4 values from the total values needed to generate the waveform as it exploits the symmetrical nature of the sine function. We have created an 8-bit sine wave. The signal is designed by VHDL a Hardware Description language. The simulation result is implemented by MODELSIM SE 6.2 plus software.

Fig. 1 Simulation of sine waveform

As shown in figure 1 the output frequency achieved is 20 KHz. The clock applied is 10 MHz. This kind of basic signal generation process has a serious disadvantage as we cannot control the output frequency and also the output frequency is very small.

IV. PROPOSED METHODOLOGY

The proposed DDS method overcomes the problem of adjusting the output frequency. It gives us the ability of producing a wide range of output frequency. The block diagram of the Direct digital synthesizer is shown in figure 2. This will be composed of a reference clock, a register containing a binary word, a Numerically controlled oscillator (NCO) and a digital to analog converter (DAC). As shown below the register, NCO and the converter will be governed by the same global clock.

Fig. 2 Block diagram of DDS

Fig. 3 Block diagram of NCO

As shown in figure 3 the NCO will be made up of a Phase accumulator and a Phase to amplitude converter. The output frequency of DDS will be given by:

$$F_{out} = W \cdot \frac{f_{clk}}{2^N} \quad \text{----- (1)}$$

Here we see that the output frequency depends on the binary word, bit length of the adder and the system clock. So to change the output frequency, only one of the above parameters need to be changed. This adds great flexibility to the device output frequency. Here sine, cosine, sawtooth and square waveforms will be generated. The design will be simulated by Xilinx ISE tool and MODELSIM software which would be required to see the analog waveforms.

V. DESIGN OF SUBMODULES OF DDS

A. Design of phase to amplitude converter

This section of DDS contains a Look up table in a Read only memory (ROM). Two separate LUT's for sine and cosine are prepared. This LUT is used to convert the phase accumulators output value which is linear into the waveforms amplitude information which gives the waveform it's basic shape. It is basically a memory which contains pre sampled 12-bit binary values which are stored in hexadecimal format. These values act as amplitudes corresponding to phase

addresses which are taken into account as integer values. The phase accumulator output is of 32 bit. If we use all the 32 bits as address, we would require 4.29GB space for the look up table. Hence phase truncation is done. We use only the top 12 bits of phase accumulator to address the LUT. Therefore, the total memory locations required to store the amplitude values are only $2^{12} = 4096$.

B. Design of phase accumulator

The block diagram of phase accumulator as shown in figure 4 consists of a divider, a multiplier and an adder. The accumulator structure will be of 32 bits. Here the phase incrementer stores a binary value (W) which is the key factor in deciding the output frequency. The phase incrementer is achieved by the divider and the multiplier which is shown by the formula:

$$\text{Phase-inc (W)} = (F_{\text{out}}/F_{\text{clk}}) * 2^{32} \quad \text{-----} \quad (2)$$

Fig. 4 Block diagram of Phase accumulator

On each tick of the clock pulse, the adder will add the phase increment value with the output of adder which is the result of previous clock pulse because of the feedback connection. This process will help in giving some slope to the waveform.

Fig. 5 Digital phase wheel

As shown in figure 5 the phase accumulator function could actually be represented as a digital phase wheel representing 360°. The complete sine wave oscillation can be visualized as a point moving around the circle. Each designated point like a, b as shown in figure 5 corresponds to equivalent point on the cycle of the sine wave. One complete rotation of the cycle results in one oscillation of the wave. Once the cycle is complete the rotation starts over again. The number of designated points on the cycle is generally determined by N which is 32 here i.e 2^{32} points. But due to phase truncation the number of points are equal to 2^{12} . So the address “000000000000” corresponds to 0° and the address “111111111111” is equal to 359.98°. The distance between two points is determined by the value of W which is the phase increment. Hence varying the output frequency is possible by changing the value of W. For higher frequency, W should be large and vice versa.

VI.SIMULATION RESULTS

The above-mentioned modules are simulated and synthesized by Xilinx ISE 13.1i.

Fig. 6 RTL view of NCO block

As shown in figure 6, the RTL view gives the description of the LUT block, the Lut address register and the phase accumulator. The input's are clk, enable, reset and phase-incrementer and the four waveforms as the output.

The simulation showing the phase increment value with a clock frequency of 100MHz and to get an output frequency of 1.7MHz as an example.

Fig. 7 Simulation of Phase incrementer

As shown in figure 7 the phase incrementer takes the correct value of X"045A1CAC" which in binary form equals to 00000100010110100001110010101100 for generating 1.7 MHz output frequency.

The simulation of the LUT is shown in the next figure which confirms the design of LUT by providing some random 12 bit addresses to check for the corresponding amplitudes.

Fig. 8 Simulation result of LUT

As shown in figure 8 for addresses X"001", X"010", X"101", X"011", X"110" we get the corresponding amplitudes from sine and cosine LUT's.

VII. FUTURE WORK

This complete system can be designed on Field programmable gate array (FPGA). The reference clock will be the clock present on the FPGA board. This adds great portability and the complete device will be less weighted. We can extend our DDS to generation of hyperbolic functions. This could further improve the area of application since hyperbolic functions are important part of engineering. There most common practical uses include catenaries, calculation of speed of ocean waves etc.

VIII. CONCLUSION

In this paper, we have presented different work done on FPGA based DDS. We have designed a 32 bit adder and a look up table for amplitude conversion. Multiple waveforms that are sine, cosine, square and sawtooth are generated increasing the area of application. Since these waveforms are of 12 bit each, the result will also have greater accuracy. We also have a phase resolution of $2\pi/4096 = 0.088$ degrees. This complete design has the capability to produce waveforms in the MHz range.

REFERENCES

- [1]. Changhong Shan, Zhongze Chen, Hua Yuan and Wei Hu, "Design and implementation of FPGA- based Direct Digital Synthesizer", *Electrical and Control Engineering (ICECE), IEEE international conference*, 2011, pp. 614-617.
- [2]. Matt Bergeron and Alan N. Willson, "A 1-GHz Direct Digital Frequency Synthesizer in an FPGA", *Circuits and Systems (ISCAS), IEEE International Symposium*, 2014, pp.329-332.
- [3]. Sunny Raj Dommaraju, Saiyu Ren and George Y. Lee, "Design and implementation of a 16 bit flexible ROM less Direct Digital synthesizer", *Circuits and Systems (MWSCAS), IEEE International Symposium*, 2013, pp.1350-1353.
- [4]. Xiaochu Wang and Qiujun Mei "High-precision design of DDS Based on FPGA", *Intelligent Systems (GCIS), IEEE Third Global Congress*, 2012, pp.386-389.
- [5]. Qing Wang, Songbai He and Ziming Zhong, "Design and Simulation of an Optimized DDS", *Wireless Communication and Networking and Mobile Computing (WiCOM), IEEE International Conference*, 2010.
- [6]. Muhammad Asim Butt and Shahid Masud, "Fpga based bandwidth adjustable all Digital Direct Frequency Synthesizer", *Communications and Information Technology (ISCIT), IEEE International symposium*, 2009, pp.1399-1404.
- [7]. Jeffrey H. Reed, Brian D. Woerner, "A tutorial on Direct Digital Synthesizer", *The 9th international symposium on personal, Indoor and mobile Radio Communications*, 1998.