

DESIGN AND IMPLEMENTATION OF DIRECT DIGITAL FREQUENCY SYNTHESIZER USING VHDL

UYYALA SRINIVASA RAO 1*, Mr. M MOSHE 2*, AJMERA KRISHNA MURTHY 3 *

1. Asst Prof, JJIIT, HYDERABAD.
2. Asst.Prof, Princeton Women's Engineering College, Hyderabad.
3. Asst.Prof, Princeton Women's Engineering College, Hyderabad.

Abstract:

Direct digital frequency synthesis is widely used in modern mobile communication receivers in order to tune to the frequency of interest. DDFS will eliminate the manual tuning of station which is present in present day receivers, by software tuning. DDFS can generate signal waveforms like triangular, square, and ramp and modulated waveforms like ASK, FSK and PSK signals. We can also use DDFS as modulator for modulating base band signals. DDFS will be useful for implementing software defined radio front end. The features of this DDFS are we can generate frequencies of the order 10MHz and the output will be 16bit. The system will be optimized to give better spurious free dynamic range.

1. INTRODUCTION

The term "frequency synthesis" applies to a technique that accepts some reference input and then generates one or more signals of predefined type as according to a control word or method. The stability, accuracy, and spectral purity are the performance measures of frequency synthesizer. Three conventional frequency synthesis techniques are popular throughout the industry: phase-lock loop (PLL, or "indirect"), mix/filter/divide (direct-analog), and direct-digital. Each of these methodologies has advantages and disadvantages; hence each application requires selection based upon the most acceptable combination of compromises.

1.1 DIRECT ANALOG SYNTHESIS

Direct-analog synthesis, also called mix/filter, uses echelons of multiplication, division, and other mathematical manipulations to produce the desired new frequency. The process is called "direct" because the error correction process is avoided; hence the quality of the output correlates directly with the quality of the input. Phase noise is typically excellent because of the direct process, particularly close to the carrier, and switching speed can be very fast. It's an expensive process, however, and finer steps require additional circuitry that further increase complexity and cost.

1.2 PLL BASED INDIRECT SYNTHESIS

Indirect synthesis also called "phase-lock-loop" or PLL, compares the output of a voltage controlled oscillator (VCO)

with some reference signal. As the output drifts, detected errors produce correction commands to the VCO, which responds accordingly. Error detection and corrections occur in a phase detector, which adds phase noise close to the carrier, though a PLL can outperform direct synthesis at larger offsets. Fine steps degrade phase noise, and fast switching is difficult to achieve with a PLL design even with the use of aggressive VCO pretuning techniques.

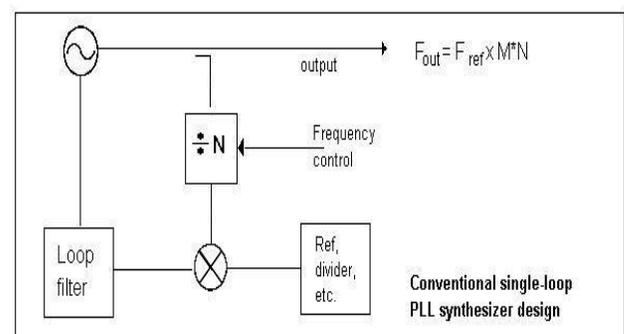


Figure 1. PLL based synthesizer

1.3 DIRECT DIGITAL SYNTHESIS

Direct-Digital synthesis (DDS) is the most recently developed frequency synthesis technique, dating from the early 1970s. All three techniques have been available to designers for

decades, but it is the direct-digital synthesizer (DDS) that is evolving the most rapidly today. In fact, only a few years ago the DDS was a phenomenon with little utility, but now it is an important design tool that can not be ignored by the architects of any system demanding frequency agility. All other signal generation techniques begin with some sort of oscillator, the output of which is manipulated or controlled by the synthesizer. The DDS is unique because it is digitally deterministic; the signal it generates is synthesized from a digital definition of the desired result.

A DDS uses logic and memory to digitally construct the desired output signal, and a data conversion device to convert it from the digital to the analog domain. Therefore, the DDS method of constructing a signal is almost entirely digital, and the precise amplitude, frequency, and phase are known and controlled at all times. Due to its basic principle it also called Direct Digital Frequency Synthesizer (DDFS)

2. DDFS BASIC PRINCIPLE

Direct Digital Frequency Synthesizer is a technique to produce desired output waveforms with full digital control (hence also called Numerically Controlled Oscillator). Direct digital synthesis (DDS) is becoming increasingly popular as a technique for frequency synthesis, especially if high frequency resolution and fast switching between frequencies over a large bandwidth are required.

In DDFS The instantaneous phase of a sinusoidal signal is given by a number stored in a digital accumulator. The accumulator is incremented by adding a constant amount at each clock period, its content will represent a phase value which increases linearly with time. When the accumulator exceeds a value equivalent to 2π radians, it overflows, multiples of 2π are discarded, and the incrementation process continues to the next cycle. The number held in the accumulator is used to address a look-up table held in ROM (read- only memory) which converts phase information to a series of discrete, digitized samples of the amplitude of a sine-wave. A DAC (digital-to-analogue converter), followed by a low-pass filter, convert the digital samples into an analogue signal. Different Look Up Tables can be used to produce desired output waveform such as square wave and triangular wave etc.

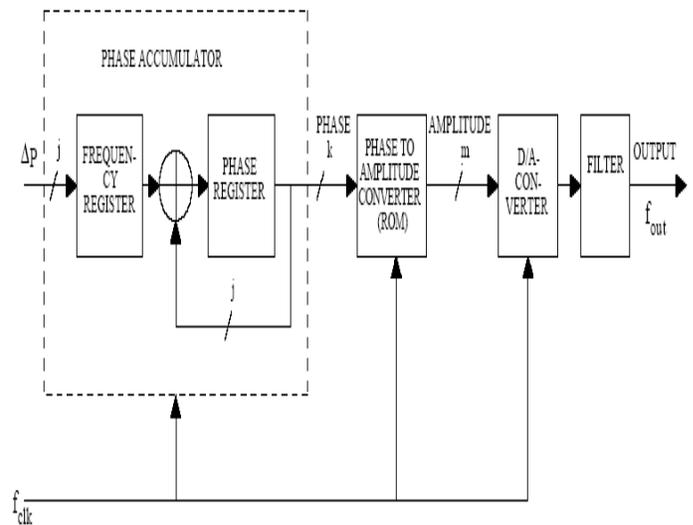


Figure 2. Basic Block Diagram of DDFS

The read only memory (ROM) is a sine look-up table, which converts the digital phase information into the values of a sine wave. In the ideal case with no phase and amplitude quantization, the output sequence of the table is given by

$$\sin\left(2\pi \frac{P(n)}{2^j}\right),$$

where $P(n)$ is a (the j -bit) phase register value (at the n th clock period). The numerical period of the phase accumulator output sequence is defined as the minimum value of Pe for which $P(n) = P(n+Pe)$ for all n . The numerical period of the phase accumulator output sequence (in clock cycles) is

$$Pe = \frac{2^j}{\text{GCD}(\Delta P, 2^j)},$$

where $\text{GCD}(\Delta P, 2^j)$ represents the greatest common divisor of ΔP and 2^j . The numerical period of the sequence samples recalled from the sine ROM will have the same value as the numerical period of the sequence generated by the phase accumulator. Therefore, the spectrum of the output waveform of the DDS prior to a digital-to-analog conversion is characterized by a discrete spectrum consisting of Pe points. The ROM output is presented to the D/A converter, which develops a quantitated analog sine wave. The D/A-converter output spectrum contains frequencies $n f_{clk} \pm f_{out}$, where $n = 0, 1, \dots$ etc. The amplitudes of these components are weighted by a function.

$$\text{sinc}\left(\frac{f}{f_{clk}}\right).$$

This effect can be corrected by an inverse $\text{sinc}(f/f_{clk})$ filter. The filter that is after the D/A converter removes the high frequency sampling components and provides a pure sine wave output. As the DDS generates frequencies close to $f_{clk}/2$, the first image ($f_{clk} - f_{out}$) becomes more difficult to filter. This results in a narrower transition band for the filter. The complexity of the filter is determined by the width of the transition band. Therefore, in order to keep the filter simple, the DDS operation is limited to less than 40 percent of the clock frequency.

2.1 DDFS Architecture for Modulation Capability

It is simple to add modulation capabilities to the DDS, because the DDS is a digital signal processing device. In the DDS it is possible to modulate numerically all three waveform parameters

$$s(n) = A(n) \sin (2 \pi (\Delta P(n) + P(n)))$$

where $A(n)$ is the amplitude modulation, $\Delta P(n)$ is the frequency modulation, and $P(n)$ is the phase modulation. All known modulation techniques use one, two or all three basic modulation types simultaneously. Consequently any known waveform can be synthesized from these three basic types within the Nyquist band limitations in the DDS. Figure 1.3 shows a block diagram of a basic DDS system with all three basic modulations in place

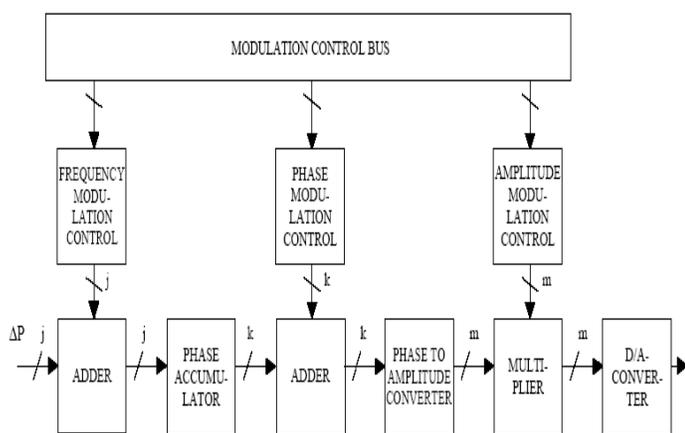


Figure 3. DDS architecture with modulation capabilities.

Frequency modulation is made possible by placing an adder before the phase accumulator. The phase modulation

requires an adder between the phase accumulator and the phase to amplitude converter. The amplitude modulation is implemented by inserting a multiplier between the phase to amplitude converter and the D/A-converter. The multiplier adjusts the digital amplitude word applied to the D/A-converter. Also, with some D/A-converters it is possible to provide an accurate analog amplitude control by varying a control voltage.

3.FIELD PROGRAMMABLE GATE ARRAY (FPGA)

Traditional gate arrays contain a number of building blocks or primitive cells etched on a single silicon substrate. The connections between cells are permanent and made later. These are non-reprogrammable high-density devices containing about 5 millions gates. The FPGAs have similar structure to gate arrays however they have programmable elements. The programmable cell is called Logic Element (LE) in case of Altera device and Configurable Logic Block (CLB) in Xilinx devices. FPGA use the Complementary Metal Oxide Semiconductor SRAM technology and are thus reset at power off.

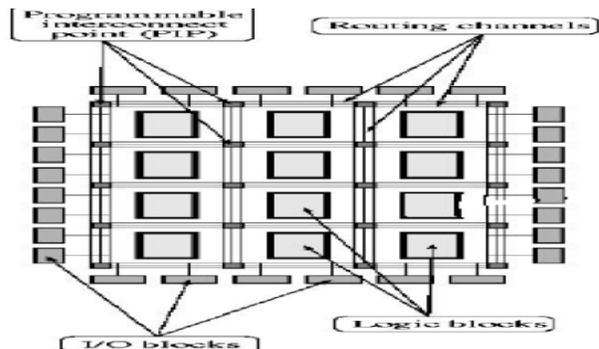


Figure 4. Medium Density FPGA Architecture

Altera FLEX 8000 and FLEX 10000 FPGAs

Altera’s FLEX 8000 series consists of a three-level hierarchy much like that found in CPLDs. However, the lowest level of the hierarchy consists of a set of lookup tables, rather than an SPL D like block, and so the FLEX 8000 is categorized here as an FPGA. It should be noted, however, that FLEX 8000 is a combination of FPGA and CPLD technologies. FLEX 8000 is SRAM-based and features a four-input LUT as its basic logic block. Logic capacity ranges from about 4000 gates to more than 15,000 for the 8000 series. The overall architecture of FLEX 8000 is illustrated in Figure 2.6. The basic logic block, called a Logic Element (LE) contains a four-input LUT, a flip-flop, and special-purpose carry circuitry for arithmetic circuits. The LE

also includes cascade circuitry that allows for efficient implementation of wide AND functions. Details of the LE are illustrated in Figure 2.7.

In the FLEX 8000, LEs are grouped into sets of 8, called Logic Array Blocks. As shown in Figure 2.8, each LAB contains local interconnect and each local wire can connect any LE to any other LE within the same LAB. Local interconnect also connects to the FLEX 8000's *global interconnect*, called, FastTrack is similar to Xilinx long lines in that each FastTrack wire extends the full width or height of the device.

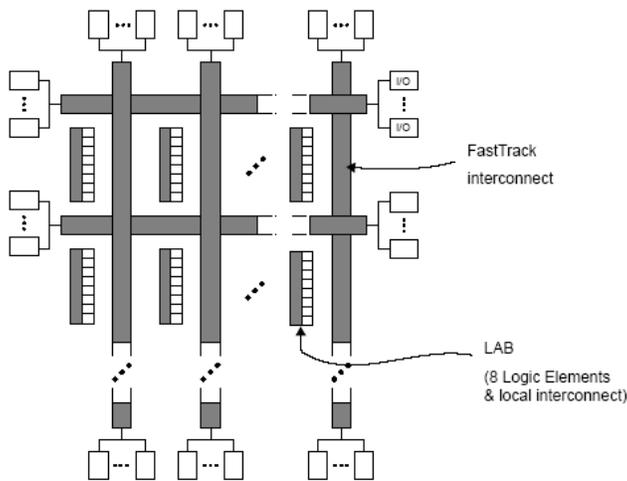


Figure 5. Architecture of Altera FLEX 8000 FPGAs.

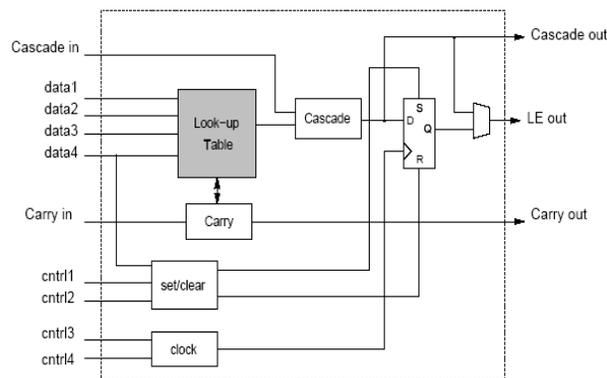


Figure 6 Altera FLEX 8000 Logic Element (LE).

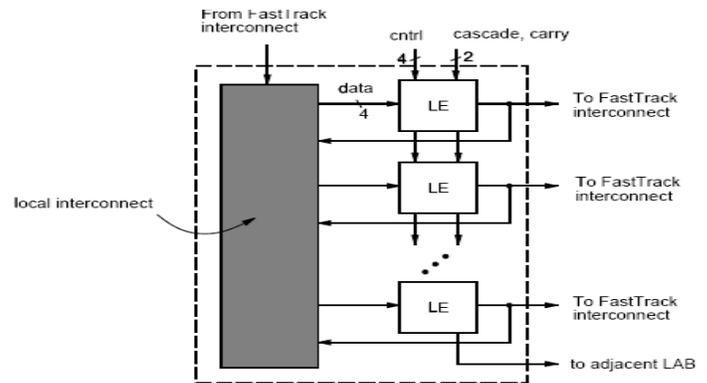


Figure 7. Altera FLEX 8000 Logic Array Block (LAB).

This makes the FLEX 8000 easy for CAD tools to automatically configure. All Fast-Track wires horizontal wires are identical, and so interconnect delays in the FLEX 8000 are more predictable than FPGAs that employ many smaller length segments because there are fewer programmable switches in the longer paths. Predictability is furthered aided by the fact that connections between horizontal and vertical lines pass through active buffers.

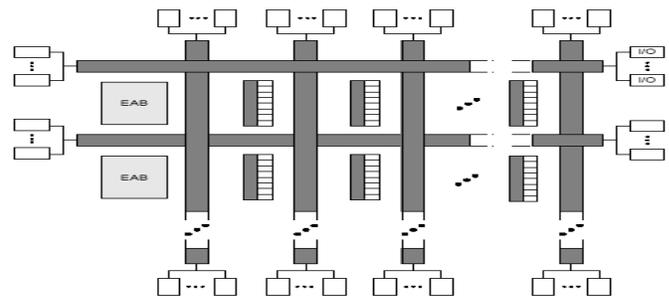


Figure 8 Architecture of Altera FLEX 10K FPGAs.

Altera provides, as part of their CAD tools, several macro-functions that implement useful logic circuits in EABs. Counting the EABs as logic gates, FLEX 10000 offers the highest logic capacity of any FPGA, although it is hard to provide an accurate number.

Applications of FPGAs

FPGAs have gained rapid acceptance and growth over the past decade because they can be applied to a very wide range of applications. A list of typical applications includes: random logic, integrating multiple SPLDs, device controllers, communication encoding and filtering, small to medium sized systems with SRAM blocks, and many more. Other interesting applications of FPGAs are prototyping of designs later to be implemented in gate arrays, and also emulation of entire large hardware systems. The former of these applications might be

possible using only a single large FPGA (which corresponds to a small Gate Array in terms of capacity), and the latter would entail many FPGAs connected by some sort of interconnect; for emulation of hardware. Another promising area for FPGA application, is the usage of FPGAs as custom computing machines. This involves using the programmable parts to “execute” software, rather than compiling the software for execution on a regular CPU.

4. IMPLEMENTATION OF DDFS

The basic block diagram of DDFS implemented is shown in the below figure. All the blocks are connected with common clock and reset signals. The delta phase value decides the phase increment for each clock pulse. Hence decides the resulting signal frequency. The Frequency modulating instantaneous value is added to the delta phase value which causes instantaneous change in frequency. Due to the digital nature of the modulator only at each clock tick the modulating signal value shall affect the resulting frequency. If the modulating signal is analog then an Analog Digital converter

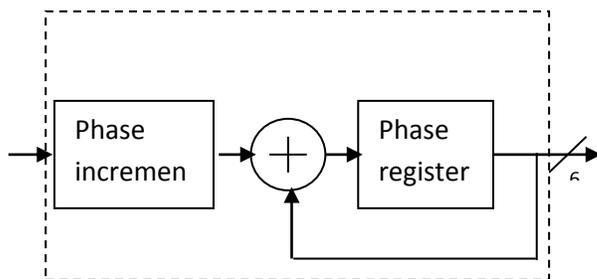


Figure .9 Phase Accumulator

The outputs of four Look Up Tables are given to the input lines a 4 to 1 Multiplexer. This multiplexer connects one of the inputs to the output depending on the select lines. The output of Multiplexer consists the 6 amplitude bits which is the final output in case required modulation schemes are FM or PM. In case of Amplitude modulation, the output of Multiplexer is multiplied with instantaneous modulating signal. In three modulation schemes if modulating signal is analog in nature then an appropriate Analog to Digital converter is required to convert into 6 bit digital output.

must be used to digitize the modulating signal which can be used in DDFS.

The phase accumulator produces accumulated phase value for each clock pulse. In case if the DDFS is used for phase modulation then instantaneous phase modulating signal value is added to the phase output of phase accumulator. This resulting phase value is given to the four Look Up Tables. Each Look Up Table is configured to produce a specific waveform. The logic used to generate the Look Up Tables is discussed in the further sections.

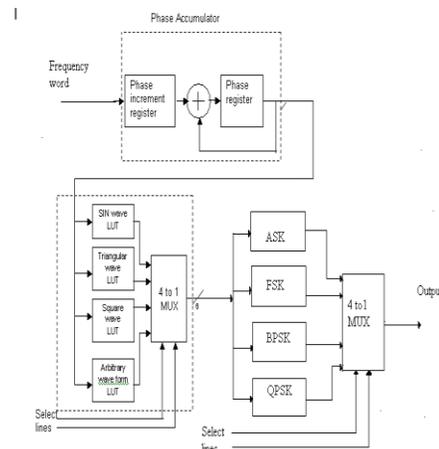


Figure 10. Basic Block diagram of Direct Digital Frequency Synthesizer implemented in this project.

From the figure 9, the basic blocks in DDFS can be identified as PIPO registers, adders, Look Up Tables and other combinational circuits. The following sections presents the implementation details and results obtained for all these blocks.

4.1 PIPO n bit generic register

The Parallel in Parallel Out shift register cells are required in phase accumulator block to hold frequency and phase values. Synchronization is required between the phase increment register and phase register. This is achieved by connecting a common clock signal. This module with behavioral style in

VHDL code. The Generic is used in VHDL implementation which allows to instantiate the PIPO component any bit size. The figure 3.2 shows RTL diagram for PIPO register.

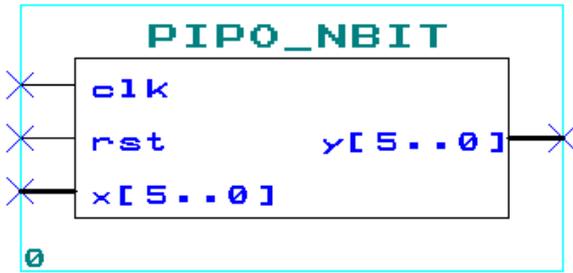


Figure 11. PIPO N bit shift register

4.2 N bit generic adder

The N-bit generic adder is implemented in VHDL with simple ripple carry adder logic. The VHDL code for N bit adder is implemented in behavioral model. The RTL diagram for N bit adder show in bellow.

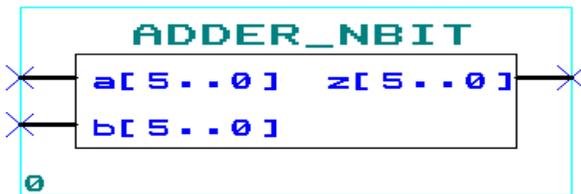


Figure 12.PIPO N bit generic adder

4.3 Phase Accumulator

The phase accumulator consists of phase increment register, adder and phase register. The figure 3.4 shows the inner blocks of phase accumulator.

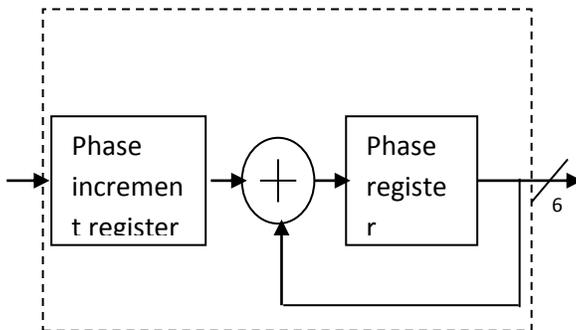


Figure 13. Phase Accumulator

The phase increment register stores the instantaneous phase increment values resulting from frequency modulation control block. This is fed to a 6 bit adder as one of its input. The other input for adder is phase register output. The phase register holds the instantaneous phase for each clock pulse. The accumulated phase also is represented by 6 bits, which limits the maximum phase by 111111, and addition by 1 to maximum value causes the phase to become 000000 This is expected and desired since the Look Up Tables are programmed to consider 63 as highest phase value and phase increment by one results next cycle of waveform. Since 6 bits are used to represent the 0⁰ to 360⁰ the increment in digital phase value by one causes effective increment of 5.625⁰ (results by dividng 360⁰ with 64 maximum possible combinations of 6 bits) . This also implies that outputs can't have more that 64 samples for one cycle.

The VHDL code for phase accumulator implemented in behaviourl model.The RTL diagram for phase accumulator show in figure 3.5.

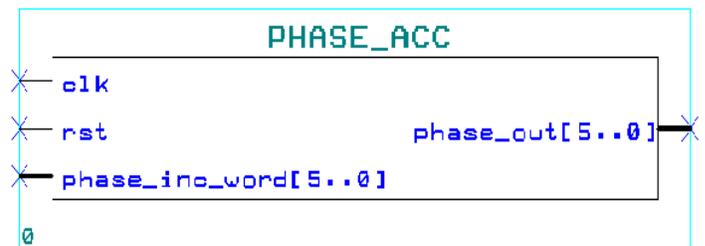


Figure 14. RTL diagram for phase accumulator

The output of phase accumulator when the phase increment value is 000100 (decimal four) is given in figure 3.6. It can be observed that the resulting phase value after each clock pulse is four added to the previous phase value. In the following figure initial phase is 0 and further with clock pulses resulting in 4,8,12,16 ...

4.4 Look Up Tables

Four Look Up Tables are implemented to produce four different output waveforms, namely sine wave, square wave, triangular wave and arbitrary waveform. As a standard practice these LUTs are implemented using VHDL CASE statement. The generation of square wave requires producing only two amplitude levels in one cycle; it can be implemented without Look up Table. A stair case waveform Look up Table is implemented for arbitrary waveform. It is possible to implement any arbitrary waveform, by appropriately changing the content of LUT. To fully characterize arbitrary waveform in 6bit DDS, we require 64 amplitude values corresponding to 64 phase values. The ports of all four look up tables are same. The RTL diagram for sin wave shown figure 3.6.

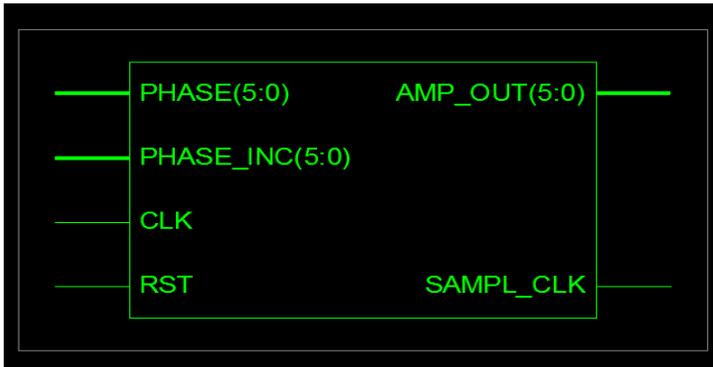


Figure 15. RTL diagram for sin wave

In these waveform are use to generate the different types digital modulating techniques like ASK (Amplitude shift keying), FSK (Frequency shift keying), BPSK (Binary shift keying), QPSK (Quadrature phase shift keying). In this project we implementing the different types of waveforms

5. RESULTS

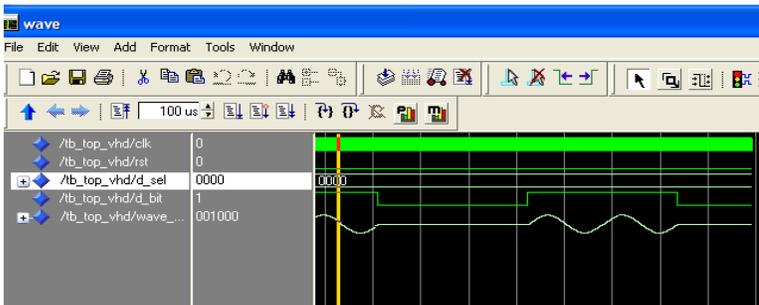


Figure 16. Output wave ASK using SIN

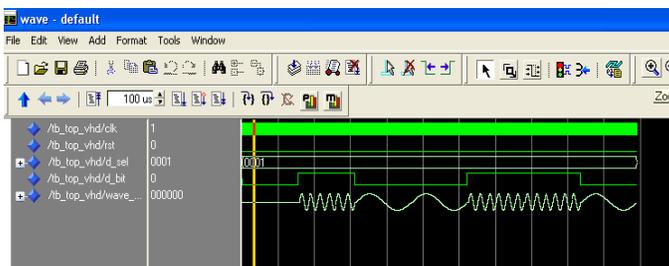


Figure 17. Output wave FSK using SIN

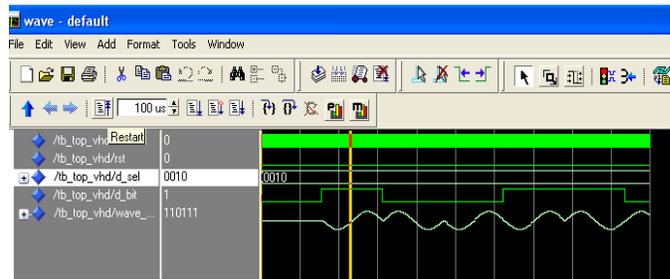


Figure 18. Output wave APSK using SIN

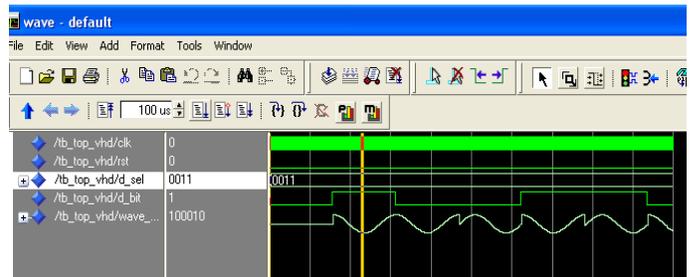


Figure 19. Output wave QPSK using SIN

6. CONCLUSIONS

A major advantage of a direct digital synthesizer (DDS) is that its output frequency, phase and amplitude can be precisely and rapidly manipulated under digital control. Other inherent DDS attributes include the ability to tune with extremely fine frequency and phase resolution. In current technology DDFS is a viable alternative to analog based phase-locked loop (PLL) technology for generating agile analog output frequency in consumer synthesizer applications. It is easy to include different modulation capabilities in the DDS by using digital signal processing methods, because the signal is in digital form. By programming the DDS. The flexibility of the DDS makes it ideal for different types of signal generators. The digital circuits used to implement signal-processing functions do not suffer the effects of thermal drift, aging and component variations associated with their analog counterparts. The implementation of digital functional blocks makes it possible to achieve a high degree of system integration. Recent advances in IC fabrication technology, particularly CMOS, coupled with advanced DSP algorithms and architectures are providing possible single-chip DDFS solutions to complex communication and signal processing subsystems as modulators, demodulators, local oscillators (LOs), programmable clock generators, and chirp generators. The DDS addresses a variety of applications, including cable modems, measurement equipments, arbitrary waveform generators, cellular base stations and wireless local loop base stations.

7. FUTURE SCOPE

In DDFS the quantization at DAC output waveform cause the generation of spectral power at unwanted frequencies, corrupting the DDFS output. This effect is measured as SFDR which is the difference in power between the synthesised signal and the next most powerful (unwanted) frequency in the output spectrum. The SFDR requirement of a Bluetooth system is -50 dBc]. The Bluetooth specification also requires a channel bandwidth of 1MHz which is frequency hopped over 79 channels every 625 μ s, with frequency resolution of 1ppm. The ROM is a bottleneck for high frequency performance since its functions cannot be pipelined to increase the sampling rate. The data held in ROM was compressed by interpolation techniques which exploit the symmetry between the quadrants of a sinusoid. By appropriately inverting the phase and amplitude of the sine-wave output, the look-up table need only hold information for phase values between 0 and $\pi/2$.

7. REFERENCES

1. A Direct Digital Frequency Synthesis System for Low Power Communications by Alistair McEwan, Sunay Shah and Steve Collins, Department of Engineering Science University of Oxford Parks Road Oxford UK
2. QAN19 Modulating Direct Digital Synthesizer in a QuickLogic FPGA
3. Digital frequency synthesis avoids problems with analog from <http://dataweek.co.za>
4. "Direct Digital Synthesis (DDS) using FPGA" from <http://www.hunteng.co.uk>
5. <http://sss-mag.com/destopics.html>
6. Choosing a waveform generator: www.ednmag.com
7. Spread Spectrum Scene Direct Digital Synthesizer (DDS) Stuff from <http://sss-mag.com>
8. Analog Devices - DDS info center <http://www.analog.com/en/subCat/0,2879,770%255F843%255F0%255F%255F0%255F0,00.html>
9. GSPS DDS - <http://www.analog.com/en/prod/0,,AD9858,00.html> Fairchild (Raytheon Electronics) - Coordinate Transformer, Digital Synthesizer
10. <http://www.logicdevices.com/ps/index.htm> (Arithmetics Functions) Intel (Stanford Telecom) - Frequency Synthesis Products
11. <http://www.intel.com/design/digital/index.htm>
12. Douglas L. Perry , "VHDL", Second Edition, McGraw-Hill, Inc, 1993
13. Zainalabedin Navabi, "VHDL Analysis and Modelling of Digital Systems", McGraw – Hill, Second Edition
14. Altera MAX+PLUS II getting started from www.altera.com

Mr. Uyyala Srinivasa Rao, Working as a Asst. Prof. in the



Department of ECE, He Completed his B.Tech (ECE) from JNTU and M.Tech (ECE) in the specialization of DE&CS from JNTUK , He is Interested Communications , Microprocessors , and Antenna wave Propagation .

Presently he is working in JJ Institute of Information Technology, Maheswaram, R. R Dist.



Mr. Maddu. Moshe, Asst. Prof, He Completed his B.Tech(ECE) from JNTUH and M.Tech (ECE) from ANU Bapatla , he is interested Communication and Signal Processing , Linear Integrated Circuits and Electromagnetic Waves and

Transmissions Lines . Presently he is working in Princeton Institute of Engineering & Technology for Women Ghatkesar R. R Dist. in the Department of Electronics and Communications Engineering.



Mr. Ajmera. Krishnamurthy, Asst. Prof, He Completed his B.Tech(ECE) from JNTUH and M.Tech (ECE) from JNTUH , he is interested Communications , Signals and Systems , Pulse and Digital Circuits . Presently he is working in Princeton Institute of Engineering &

Technology for Women Ghatkesar R. R Dist. in the Department of Electronics and Communications Engineering.