International Multidisciplinary Research Journal Reviews (IMRJR)
A Peer-reviewed journal
Volume 2, Issue 7, July 2025
DOI 10.17148/IMRJR.2025.020702

Design and Implementation of a Direct Digital Frequency Synthesizer (DDFS) And Quadrature Direct Digital Frequency Synthesizer (QDDFS) Using FPGA

Dr. Kamal Aboutabikh

Faculty of Informatics Engineering, Ittihad Private University, Damascus Syria

Abstract:
In this paper, the direct digital frequency synthesizer (DDFS) and Quadrature direct digital frequency synthesizer (QDDFS) specifications were improved by increasing the number of accumulator bits, increasing the memory capacity of the generated signal samples, using a digital analog convertor (DAC) by increasing the number of its bits, as well as generating different types of analogue signals in a digital way.
Also, we discuss a practical mechanism of a Quadrature direct digital frequency synthesizer (QDDFS) based on a direct digital frequency synthesizer (DDFS) using Cyclone II EP2C20F484C7 FPGA from ALTERA placed on education and development board DE-1 with the following parameters:
-Output waveforms: sin, cos.
-Frequency range: (3Hz…..10000 KHz).
-Frequency Resolution (3Hz).
-Signal amplitude (5V).
-With Reset of the generator.
-Frequency of the generated signal for all types (1MHz).

Keywords: DDFS, QDDFS, FPGA, DAC.

I. INTRODUCTION

Forming signals with different types (sine, triangle, sawtooth, Square,…) using analogue methods (as it known) depends on using familiar elements as transistors, ICS, resistances, coils, capacitors, control elements, and adjusting and calibration elements. To change the parameters of this signals (frequency, amplitude, phase), we must change the value of the elements that form this signal.The digital method forms these signals depending on calculating their samples values during one period using a mathematical equation, and store these values into a digital memory to read it serially with every clock pulse. The frequency of clock pulses determines the frequency range of the synthesizer depending on the well-known Shannon Equation.
Frequency synthesizers have the feature of changing the parameters of the signal by changing the accessing method to the ROM that contains the formed signal samples values (amplitude, phase), and also apply mathematical processes on this values to change the amplitude and make the necessary modification as we will see in this paper.
In this paper a quadrature direct digital frequency synthesizer has been designed as an application on DDFS.
In reference [1], the DDFS was designed for only simulation with the following specifications:
-sinusoidal signal.
-Frequency Range: (3Hz…..10000 KHz).
-Frequency Resolution (5Hz).
In reference [2], the author focused on reducing the size of the memory allocated to store the values of the generated signal samples, but in this paper, we do not care about the size of the memory because modern FPGA chips have a huge memory size. Rather, we considered that increasing the size of the memory increases the accuracy of shaping the digital signal and improves its specifications significantly.

II. RESEARCH IMPORTANCE AND ITS OBJECTIVES

The importance of the research stems from the urgent need to use a digital frequency synthesizer in some applications, measurement systems, and modern communication systems, as some applications require great accuracy in the value of the generated frequency and high frequency switching rates, especially in frequency hopping systems used in wireless communications, and this is what can be achieved through DDFS and QDDFS, in addition to that, DDFS , QDDFS

International Multidisciplinary Research Journal Reviews (IMRJR)
A Peer-reviewed journal
Volume 2, Issue 7, July 2025
DOI 10.17148/IMRJR.2025.020702

gives great flexibility in design and implementation, also is useful in minimising the circuits and electronic parts used in implementation, which facilitates the maintenance process and reduces size, weight and cost. It also reduces power consumption and the emission of heat that is difficult to dissipate in some applications.

III. RESEARCH MATERIALS AND ITS WAYS

To design, and test the QDDFS and DDFS for different types of signals, the following tools and software are used:
- Cyclone II EP2C20F484C7 FPGA chip from ALTERA with highly accuracy, speed, and level specifications, placed on education and development board DE-1 [3].
-DDFS and QDDFS which is considered as highly accuracy techniques in Sinusoidal signals synthesizing on FPGA chips.
-VHDL programming language with Quartus II 9.1 design environment [4].
- Design Environment MATLAB R2008a.
-GDS-1052 digital oscilloscope with Free Wave program to take the results.
-PC computer for designing and injecting the design in the chip FPGA .

IV. BLOCK DIAGRAM OF THE DDFS AND SPECIFICATIONS

The functional diagram of the DDFS is shown in figure (1) and is composed of:

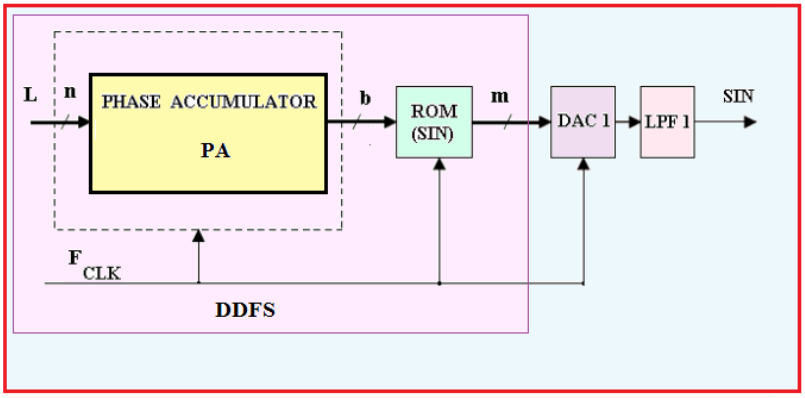

Fig. (1): functional diagram of the DDFS

1- A phase accumulator with an input of (n = 24) bits and an output of (b= 13 bits), which is allocated to form the reading address from the ROM memory. According to the value of the frequency to be formed, the value of (n), and the value of the clock pulse frequency are determined depending on the basic mathematical relationship of the digital synthesizer [5]:

$$f_{out} = \frac{F_{CLK} \cdot L}{2^n} \quad \Rightarrow L = \frac{2^n \cdot f_{out}}{F_{CLK}} \qquad (1)$$

Where:
(f_{out}): output frequency of DDFS, ($T_{out} = 1/(f_{out})$) output period of DDFS.
(n): the bits number of phase accumulator (bits).
(L): frequency code of output frequency(f_{out}).
F_{CLK}: frequency of clock pulses generator.

2- A ROM(SIN) with a capacity of 8KB contains (8192X8 bits) samples of the sinusoidal signal during one cycle.
3- A Digital-analogue converter (DAC-1) with 8 bits input to convert memory sample values from digital to analogue form.
4- A Low Pass Filter LPF-1
5- A 50 MHz clock pulse generator.

Specifications of DDFS
-Signal type is: sin signal.
-$F_{CLK} = 50$ MHz .
-Frequency Range: (3Hz…..10000 KHz)

International Multidisciplinary Research Journal Reviews (IMRJR)
A Peer-reviewed journal
Volume 2, Issue 7, July 2025
DOI 10.17148/IMRJR.2025.020702

-Frequency Resolution (δf) is [5]:

$$\delta f = \frac{F_{CLK}.L}{2^n} \qquad (2)$$

For L=1 we get:

$$\delta f = \frac{F_{CLK}.L}{2^n} = \frac{50\ X\ 10^6\ X\ 1L}{2^{24}} = 3\ \text{Hz}$$

-Frequency of the generated signal for all signals types equal : (1MHz).

For f_{out}=1MHz we get:

$$L = \frac{2^n\ .f_{out}}{F_{CLK}} = \frac{2^{24}\ X\ 1X\ 10^6}{50\ X\ 10^6} = 335544$$

V. THE BLOCK DIAGRAM AND SPECIFICATIONS OF THE DDFS FOR SINUSOIDAL SIGNAL

The stored values of the sine signal in ROM are calculated according to the relationship [5]:

$$U_{sin}(i) = INT\left[(2^{m-1}-1).sin\left(\frac{360\ \ i}{2^b}\right)\right] \qquad (3)$$

For:

$$m = 8\ bits\ \ , b = 13\ bits\ , i = (0 \ldots . 2^b - 1) = (0 \ldots . 8191)$$

$$U_{sin}(i) = INT\left[(2^{8-1}-1).sin\left(\frac{360\ \ i}{2^{13}}\right)\right] = INT\left[127\ sin\left(\frac{360\ i}{8192}\right)\right]$$

Then:

$$U_{sin}(i) = (-127 \ldots \ldots . +127)$$

To avoid negative values that are difficult to store in memory, we add a value off-set= 128, then we get:

$$U_{sin}(i) = (0 \ldots \ldots + 255)$$

To calculate the samples of the sinusoidal signal, we use (MATLAB R2008a) according to the following program section for one period:

```
i=0:1:8191;
y=floor(127*sin(2*pi*i/8192));
plot(i,y);
grid;
title ('A sin signal');
xlabel ('Time');
ylabel ('Amplitude')
```

Then we store these values in a notepad file and load it into the sin signal generator's ROM.

The design result of the sinusoidal signal using MATLAB program is shown on the figure (2).

International Multidisciplinary Research Journal Reviews (IMRJR)
International Multidisciplinary Research Journal Reviews (IMRJR)
A Peer-reviewed journal
Volume 2, Issue 7, July 2025
DOI 10.17148/IMRJR.2025.020702

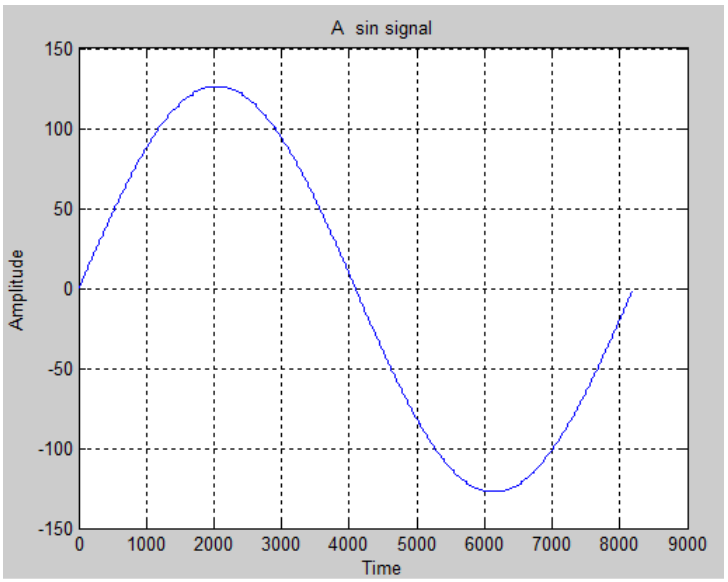

Fig. (2): sinusoidal signal using MATLAB program for one period.

The block diagram of the sinusoidal signal synthesizer in Quartus II 9.1 design environment is shown in figure (3) [6].

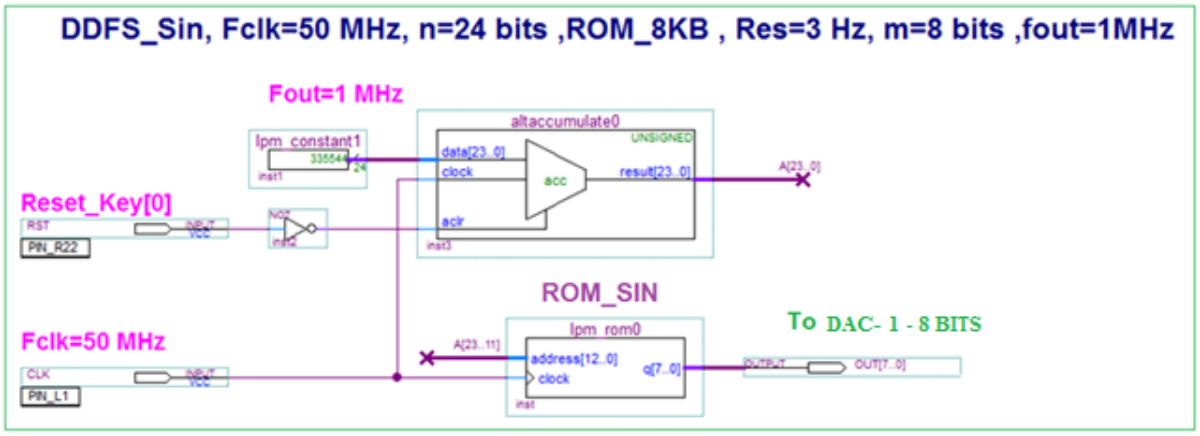

Fig. (3): Block diagram of the sin signal synthesizer in Quartus II 9.1 design environment.

The design result of the sinusoidal signal generator is taken from the digital oscilloscope screen and shown on the figure (4) for f_{out}=1MHz.

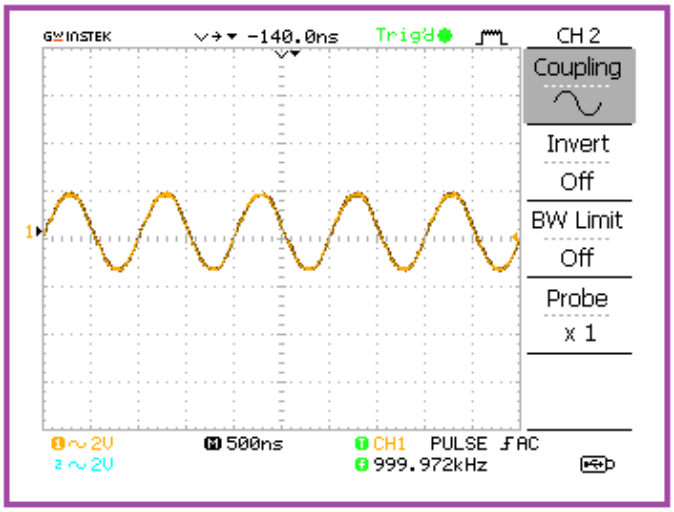

Fig. (4): The sin signal on the digital oscilloscope screen.

International Multidisciplinary Research Journal Reviews (IMRJR)
A Peer-reviewed journal
Volume 2, Issue 7, July 2025
DOI 10.17148/IMRJR.2025.020702

VI. THE BLOCK DIAGRAM AND SPECIFICATIONS OF THE DDFS FOR COSINE SIGNAL

The stored values of the cosine signal in ROM are calculated according to the relationship [5]:

$$U_{cos}(i) = INT\left[(2^{m-1} - 1).cos\left(\frac{360\ i}{2^b}\right)\right] \quad (4)$$

For:

$$m = 8\ bits\ , b = 13\ bits\ , i = (0 \ldots . 2^b - 1) = (0 \ldots . 8191)$$

Then:

$$U_{cos}(i) = INT\left[(2^{8-1} - 1).cos\left(\frac{360\ i}{2^{13}}\right)\right] = INT\left[127\ cos\left(\frac{360\ i}{8192}\right)\right]$$

$$U_{cos}(i) = (-127 \ldots . = 127)$$

To avoid negative values that are difficult to store in memory, we add a value off-set= 128

$$U_{cos}(i) = (0 \ldots \ldots 255)$$

To calculate the samples of the sinusoidal signal, we use (MATLAB R2008a) according to the following program section for one period:

```
i=0:1:8191;
y=floor(127*cos(2*pi*i/8192));
plot(i,y);
grid;
title('A cos signal');
xlabel('Time');
ylabel('Amplitude')
```

Then we store these values in a notepad file and load it into the cosine signal generator's ROM.

The design result of the cosine signal using MATLAB program is shown on the figure (5).

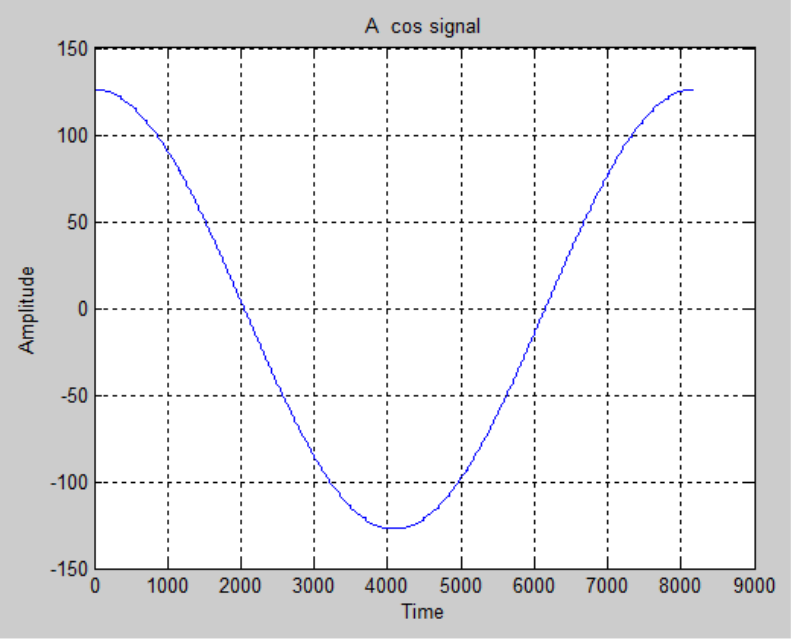

Fig. (5): cosine signal using MATLAB program for one period.

The block diagram of the cosine signal synthesizer in Quartus II 9.1 design environment is shown in figure (6) [6].

International Multidisciplinary Research Journal Reviews (IMRJR)
A Peer-reviewed journal
Volume 2, Issue 7, July 2025
DOI 10.17148/IMRJR.2025.020702

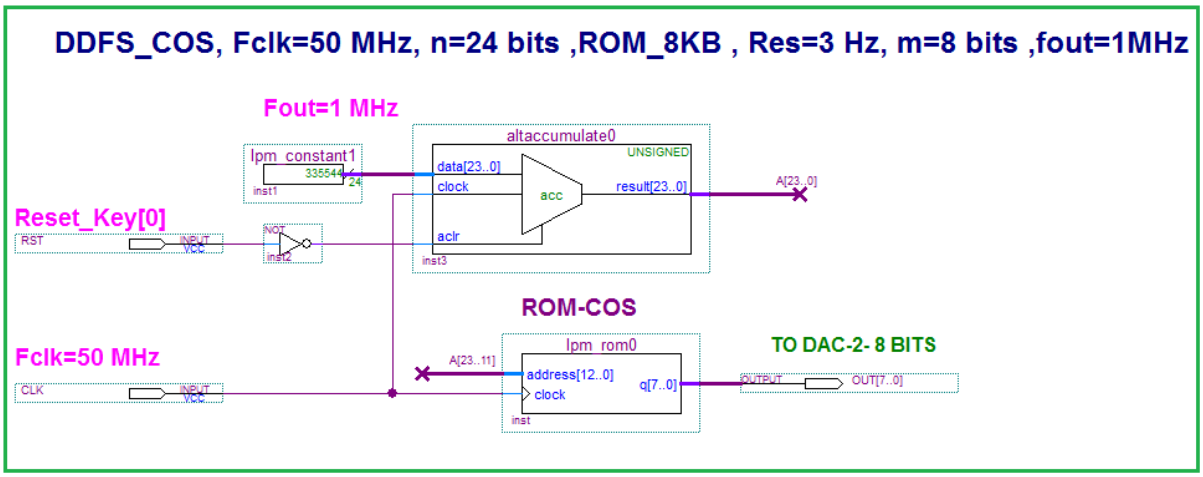

Fig. (6): Block diagram of the cosine signal synthesizer in Quartus II 9.1 design environment.

The design result of the sinusoidal signal generator is taken from the digital oscilloscope screen and shown on the figure (7) for f_{out}=1MHz.

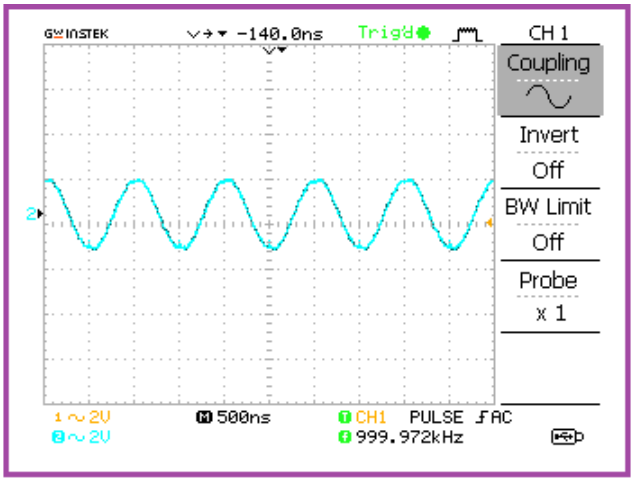

Fig. (7): The cosine signal on the digital oscilloscope screen.

VII. THE BLOCK DIAGRAM AND SPECIFICATIONS OF THE QDDFS

The functional diagram of the QDDFS is shown in figure (8) and is composed of:

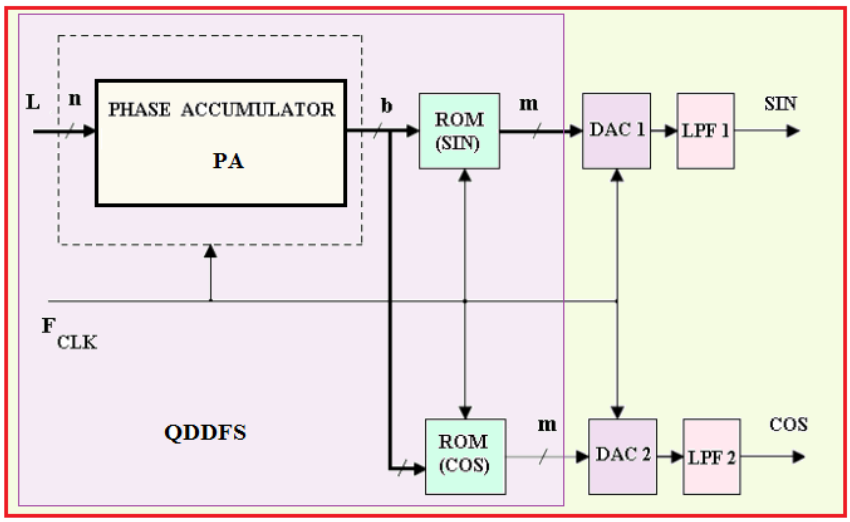

Fig. (8): Block diagram of the QDDFS

International Multidisciplinary Research Journal Reviews (IMRJR)
A Peer-reviewed journal
Volume 2, Issue 7, July 2025
DOI 10.17148/IMRJR.2025.020702

1- A phase accumulator with an input of (n = 24) bits and an output of (b= 13 bits), which is allocated to form the reading address from the (ROM-SIN and ROM -COS) memories. According to the value of the frequency to be formed, the value of (n), and the value of the clock pulse frequency are determined depending on the basic mathematical relationship of the quadrature direct digital synthesizer (QDDFS) [5]:

$$f_{out-QDDFS} = \frac{F_{CLK} . L_{QDDFS}}{2^n} \quad \Rightarrow L = \frac{2^n \ . f_{out-QDDFS}}{F_{CLK}} \qquad (5)$$

Where:
$(f_{out-QDDFS})$: output frequency of QDDFS.
(n): the bits number of phase accumulator of QDDFS (bits).
(L_{QDDFS}): frequency code of output frequency $(f_{out-QDDFS})$.
F_{CLK}: frequency of clock pulses generator.
2-The first ROM_SIN with a capacity of 8KB contains (8192X8 bits) samples of the sinusoidal signal during one cycle.
3-The second ROM_COS with a capacity of 8KB contains (8192X8 bits) samples of the cosine signal during one cycle.
4- Digital-analogue converters (DAC-1, DAC-2) with 8 bits input to convert memory sample values from digital to analogue form.
5- Low pass filter LPF-1, LPF-2.
6- A 50MHz clock pulse generator.

Specifications of QDDFS

-Signal type is: sin and cos signals.
-$F_{CLK} = 50$ MHz .
-Frequency Range of QDDFS: (3Hz…..10000 KHz).
-Frequency Resolution of QDDFS (δf_{QDDFS}) [5]:

$$\delta f_{QDDFS} = \frac{F_{CLK} . L_{QDFS}}{2^n} \qquad (6)$$

For L=1 we get:

$$\delta f_{QDDFS} = \frac{F_{CLK} . L_{QDFS}}{2^n} = \frac{50\ X\ 10^6\ X\ 1L}{2^{24}} = 3 \text{ Hz}$$

-Frequency of the generated signal for all signals types equal: (1MHz):

For $f_{out\text{-}QDDFS}$ =1MHz we get:

$$L_{QDFS} = \frac{2^n \ . f_{out-QDDFS}}{F_{CLK}} = \frac{2^{24}\ X\ 1X\ 10^6}{50\ X\ 10^6} = 335544$$

The block diagram of the QDDFS in Quartus II 9.1 design environment is shown in figure (9).

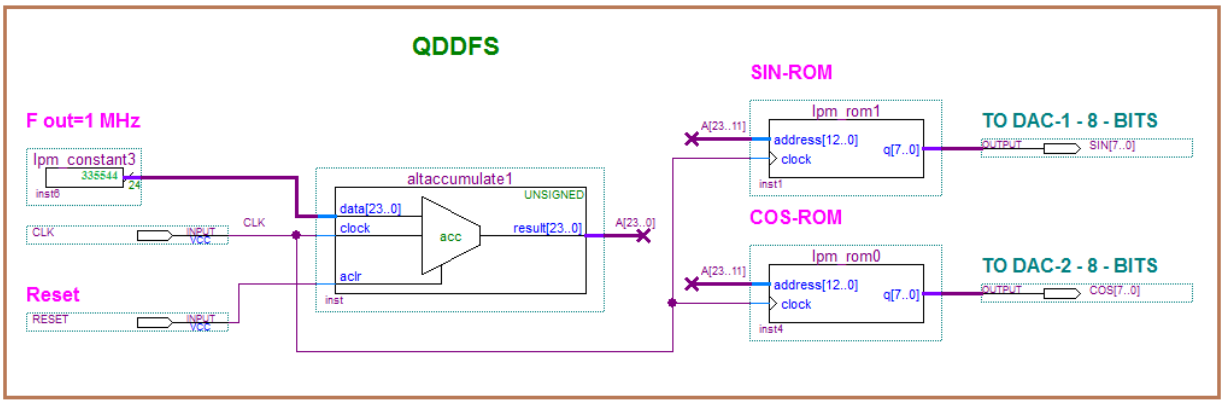

Fig. (9): Block diagram of the QDDFS in Quartus II 9.1 design environment.

International Multidisciplinary Research Journal Reviews (IMRJR)
A Peer-reviewed journal
Volume 2, Issue 7, July 2025
DOI 10.17148/IMRJR.2025.020702

The design result of the sinusoidal and cosine signals of QDDFS is taken from the digital oscilloscope screen and shown on the figure (10) for $f_{out\text{-}QDDFS}$=1MHz.

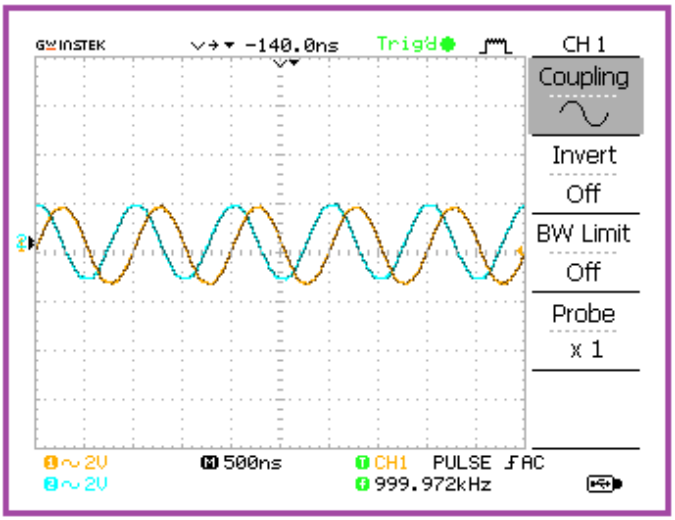

Fig. (10): the sin and cos signals on the digital oscilloscope screen.

VIII. CONCLUSIONS AND RESULTS

Based on the theoretical analysis and practical implementation that took place in this research, some important points can be extracted, which we summarize as follows:

-The use of DDFS and QDDFS allows generating any signal no matter how complex, as well as it allows achieving modulation of any type, and processing of frequency hopping at high speeds.

- The use of FPGA in the design process allows the possibility of modifying the design, amending its specifications and improving them according to demand by changing the software design only.

- Any type of signal can be formed with very high accuracy، regardless of its complexity. We only need a formula to describe the signal mathematically, tabular, graphical, or other.

- Increase the frequency resolution of the DDFS and QDDFS by increasing the number of accumulator bits.

-The shape of the generated signal can be improved by increasing the memory capacity of the samples ROM and bits number of the DAC.

- The design can be developed to obtain two signals with a variable phase shift within the range $(0 \ldots .2\pi)$,to obtain the Lissajous figures.

REFERENCES

[1]. Hemanth Kumar S N, Venkateshappa, "Design and Implementation of DDS Module on FPGA", International Journal of Advanced Trends in Computer Science and Engineering, Volume 10, No.2, March - April 2021.

[2]. Aswathi Anil, "FPGA Implementation of DDS for Arbitrary wave generation", International Journal of Engineering Research and Applications, Vol. 11, Issue 7, (Series-II) July 2021, pp. 56-64www.altera.com.

[3]. Volnei A. Pedroni," Circuit Design With VHDL", MIT Press Cambridge, Massa- chusetts London, England (2004) 364.

[4]. GOLDBERG B "Digital Frequency Synthesis Demystified", LLH Technology Publishing, united states, 334.1999.

[5]. K. Aboutabikh , A. Garib , "Design and Implementation of a Digital Matched Filter for Square Pulses Signals using FPGA",International Journal of Advanced Research in Computer and Communication Engineering , Peer-reviewed & Refereed journal, Vol. 13, Issue 4, April 2024.

[6]. Afaq Ahmad, Sayyid Samir Al-Busaidi and Mufeed Juma Al-Musharafi. "On Properties of PN Sequences Generated by LFSR – a Generalized Study and Simulation Modeling". Indian Journal of Science and Technology-2013.

[7]. C. S.Rawat, Deepak Balwani, Dipti Bedarkar, Jeetan Lotwani, Harpreet Kaur Saini, "Implementation of Barker Code and Linear Frequency Modulation Pulse Compression Techniques in Matlab", International Journal of Emerging Technology and Advanced Engineering, Volume 4, Issue 4, April 2014(105-111).

[8]. Zoran Golubi cic, Slobodan Simic c, Aleksa J. Zejak, "Design and FPGA implementation of digital pulse compression for band-pass Radar signals", Journal of Electrical Engineering, VOL. 64, NO. 3, 2013, 191–195.

[9]. John C. Bancroft CREWES "Introduction to matched filters" Research Report. Volume 14 (2002).

[10]. Steve Winder "Analog and Digital Filter Design, second edition", Elsevier Science (USA),450. 2002.

International Multidisciplinary Research Journal Reviews (IMRJR)
A Peer-reviewed journal
Volume 2, Issue 7, July 2025
DOI 10.17148/IMRJR.2025.020702

BIOGRAPHY

Dr. Kamal Aboutabikh holds a PhD in communication engineering in 1988 from the USSR, university of communication in Leningrad, holds a degree assistant professor in 2009 from Aleppo university.
Lecturer at Department of Biomedical Engineering, Al Andalus University for Medical Sciences-Syria, Tishreen University-Syria, Corduba Private University- Syria, Kassala University-Sudan and Ittihad Private University- Syria.
Publish a lot of research in the field of digital communication and digital signal processing in the universities of Syria and in the European and Indian journals.
Working in the field of programming FPGA by using VHDL and design of Digital Filters.