CHAPTER ONE

INTRODUCTION

CHAPTER TWO

METHODOLOGY

CHAPTER THREE

RESULT AND DISCUSSIONS

CHAPTER FOUR

CONCLUSION AND FUTURE WORKS

Ministry Of Higher Education
And Scientific Research
University of Diyala
College of Engineering
Electronic Department

High Speed Phase Accumulator Design Based on Fast Adders Using Verilog HDL Code

The Project Submitted to the Department of Electronic Engineering
University of Diyala in Fulfillment of the Requirements for the
Degree of Bachelor in Electronic Engineering

By

Huda Hadi Saleh
Dhuha Ahmed Ibrahim

Supervised by

Dr. Salah Hassan

2016

وزارة التعليم العالي والبحث العلمي
جامعة ديالى
قسم الالكترونيك

تصميم وتنفيذ المركم المرحلي عالي السرعة باستخدام لغة البرمجة الوصفية HDL

المشروع مقدم الى قسم الهندسة الالكترونية في جامعة ديالى كجزء من متطلبات نيل الحصول على شهادة البكلوريوس في الهندسة الالكترونية

هدى هادي صالح
ضحى أحمد إبراهيم

باشراف

الدكتور صلاح حسن

2016

بِسْمِ اللَّهِ الرَّحْمَٰنِ الرَّحِيمِ

﴿ يَرْفَعِ اللَّهُ الَّذِينَ آمَنُوا مِنكُمْ وَالَّذِينَ أُوتُوا الْعِلْمَ دَرَجَاتٍ وَاللَّهُ بِمَا تَعْمَلُونَ خَبِيرٌ ﴾

صدق الله العظيم

﴿سورة المجادلة ﴾(١١)

الإهداء

إلى ..
بستان الأمل والمحبّة الخضراء دوماً ...

(والدي الحنون)

إلى ..

من تحت أقدامها بُسطتْ لحنانها وعطائها سجادة الجنة ...

(أمي الحبيبة)

إلى ..
نبض المحبّة ودفء الوصال الفردوسي ...

(زوجي الحبيب)

إلى ..

سلسلة الصدق والوئام التي ما انقطعت قلادتها من جيد حياتي طيلة مكوثي على كرسي " إقرأ "

(أصدقائي وصديقاتي)

إلى ...

رسل العلم والمعرفة والثقافة الذين نقف لهم لنفهم التبجيلا ، مقدرون لدورهم التعليمي ورسالتهم النبيلة ...

(أساتذتي الافاضل)

أهدي إليكم جميعاً شلال النور الذي قادني إلى منابر العلا ،

لأرى الحياة بعين العلم سماءً احتوت كياني

ABSTRACT

This project presents design and simulation of phase accumulator (PA). The phase accumulator consists of adder and register to generate the phase output. Multiple types of adder such as ripple carry adder (RCA,), carry look-ahead adder (CLA) and Kogge-stone adder (KS) have been used in 4-, 6-, and 8-bit adder to design the specified phase accumulator, and to evaluates the faster adder. The comparison result of the suggested designs shows that the CLA adder relatively faster. The proposed projects have been coded Verilog hardware describe language (HDL), designed simulated in ALTERA Quartus II software.

CONTENTS

List of Figures

List of Tables

List of Abbreviations

PA	Phase Accumulator
DFF	D Flip flop
RCA	Ripple carry adder
CLA	Carry look-ahead adder
KS	Kogge-Stone adder
DDFS	Direct digital frequency synthesizer
RTL	Register transfer logic
FPGA	Field programmable gate array

CHAPTER ONE
INTRODUCTION

1.1 Phase Accumulator Overview

Phase accumulator (PA) is a generator that generates the digital phase of (0 to 2π) range values. The PA architecture consists of adder and register.

The adder is the key element of the PA, whereas the register is the storage element of the data. The Frequency Control Word (FCW) input bit adds to the adder at every clock pulse and internally combines with sum bits feedback of the second register to perform an accumulation. This operation demonstrates that the PA is a counter that linearly increases (saw-tooth) with each timing clock based on the N-bit input, to the 2^N-1, and then resets back to zero.

The block diagram of phase accumulator design is shown in Figure 1.1

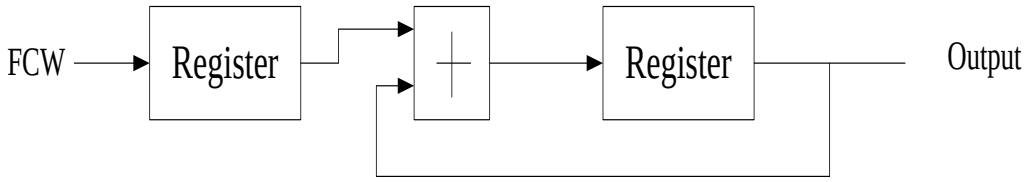

Figure 1.1 block diagram of phase accumulator

The phase accumulator can be illustrated as a phase wheel in the DDFS system. The generated phase increments have a repetitive angular phase rotating around the phase wheel in the range of 0 to 360 [1]. One rotation of the vector around the phase wheel, at a constant speed, produces one complete cycle (0 to 2π) of the output sine wave. The digital phase wheel is shown in Figure 1.2.

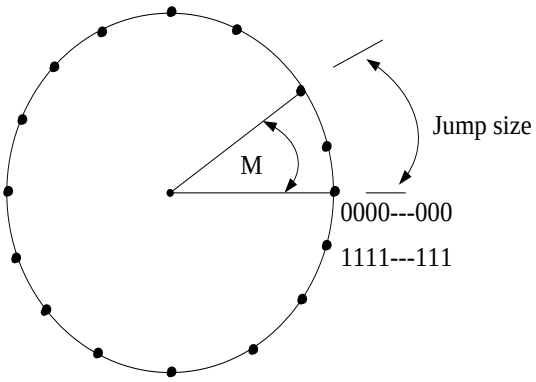

Figure 1.2 Digital Phase Wheel

The PA increments with each clock pulse correspond to the increment of the phase angle. The frequency control word (FCW) controls the incremental step in the PA.

The main constraints currently faced the researchers that work on this system is the speed and resolution [2]. On top of that, frequency resolution is another parameter that needs to be improved for better accuracy. This research addressed the requirement that able to to achieve high-speed phase accumulator throughput, fast adders, such as carry look-ahead (CLA), Kogge-Stone (KS) as well as the ripple carry adder (RCA) are used.

1.2 Phase Accumulator Design Based on Fast Adders

An adder is a key element of pipelined PA design. Most of the PA designs use ripple carry adder (RCA) (mono or groups of binary bits) for the PA. For example, 24 single bit RCA adder used in a PA with small number of registers [3], to achieve high–speed throughput.

Fast adders are desirable in the PA design to improve the PA speed. An example is the CLA, which is another type of adders that are widely used[4]. The fast CLA adder processing is offset by an increase in the components of the circuit.

Prefix adders are fast adders circuit for binary addition [5], and can be used in the PA design to increase the phase throughput. The concept of the prefix operates are as follows: Binary addition can be achieved by processing several bits at a time.

1.3 Phase Accumulator Application

The Phase accumulator application is use as a phase generator for direct digital frequency synthesizer (DDFS). The DDFS is a circuit system design to produce an analog sine waveform from a digital source. This system is defined as a technique that uses the frequency digital word to generate digital phase waveform and then convert it to analog sine wave.

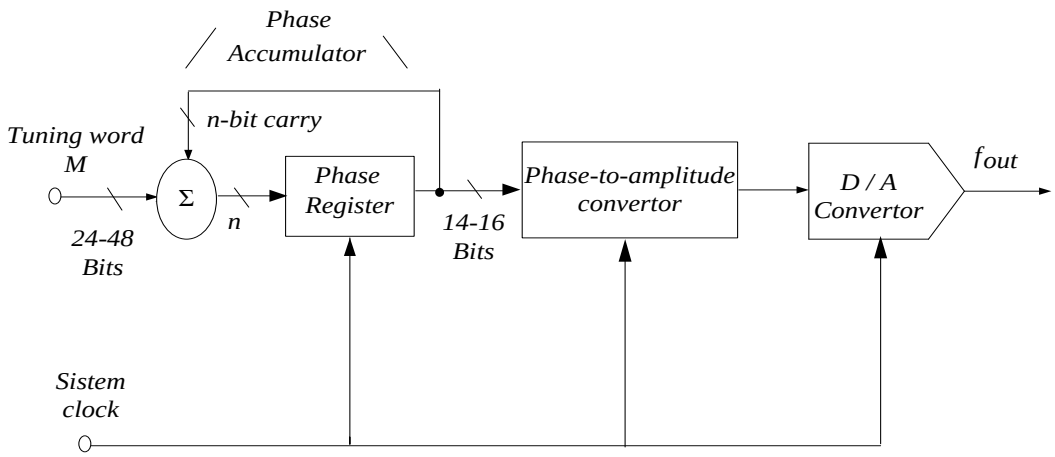

Figure 1.3 Block diagram of direct digital frequency synthesizer [1]

The phase accumulator may therefore be viewed as programmable digital a fractional divider that divides the reference sample frequency.

The fractional N frequency synthesizer is a modified version of the PLL based synthesizer [6], where the integer frequency divider is replaced by a fractional frequency divider as shown in Figure 1.4.

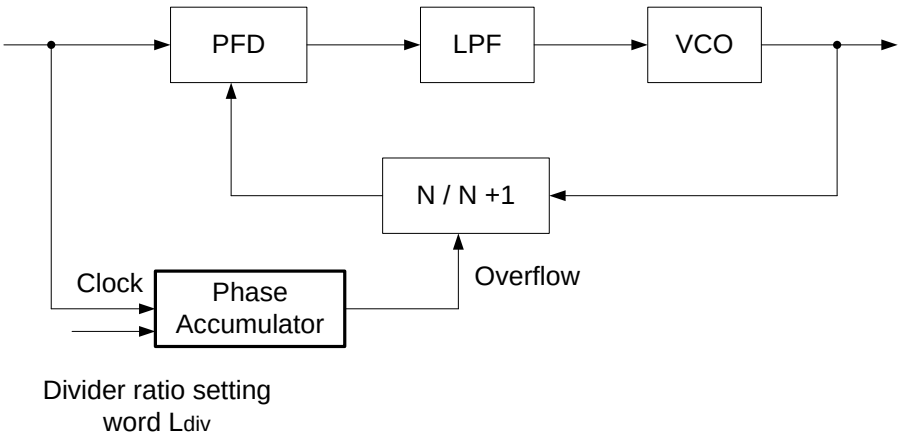

Figure 1.4 fractional N frequency synthesizers

1.4 Objectives

To achieve high-speed phase accumulator throughput, fast adders are used. Hence the adder is the main element in the design of phase accumulator, therefore choosing a fast adder for leads to high speed PA. Thus, based on the aforementioned issues, this study proposes a design that employs a PA design based on fast adders to achieve high-speed throughput output. In this project, a comparison was made for multiple types of adders to evaluate the high-speed phase accumulator which leads to improve the throughput output.

1.5 The Aim of project

The main objective of this thesis is to propose a high-speed throughput PA, to propose parallel prefix adders such as CLA adder and kogge-Stone adder (high speed adders) for phase accumulator design. The scope of this study involves:

1. Design a ripple carry adder (RCA), carry look-ahead (CLA) and Kogge-Stone adder (KS) adders.
2. Use Verilog HDL Code tools

3. Use hierarchical design techniques
4. Model and simulate the 4-, 6- and 8-bit pipelined phase accumulator using Verilog HDL

Chapter Two

Methodology

2.1 Introduction

In recent years, the interest in high-speed phase accumulator has been growing due the tremendous growth the applications of phase accumulator in direct digital synthesizer and other communication systems. The first consideration in the designing DDFS system is taking in account that the system is designed with the high-speed PA desired.

In this chapter, the high speed phase accumulator based on fast adders family (such carry look-ahead adder and Kogge-Stone adder), with multiple methods and various tuning word inputs have been designed.

2.2 Phase Accumulator Architecture

The phase accumulator is used to provide the phase output of the DDFS system. The resolution of the DDFS system as well as speed of the DDFS depends on the PA design. A detailed explanation of the proposed PA architecture will be discussed in this section.

2.2.1 Ripple-carry Adder (RCA)

An adder, also called summer, is a digital circuit that performs addition of numbers[7]. A ripple-carry adder is a logical circuit that uses multiple full adders to add N-bit numbers. Each full adder inputs a C_{in}, which is the C_{out} of the previous adder. The block circuit diagram of the 4-bit RCA is shown in Figure 2.1

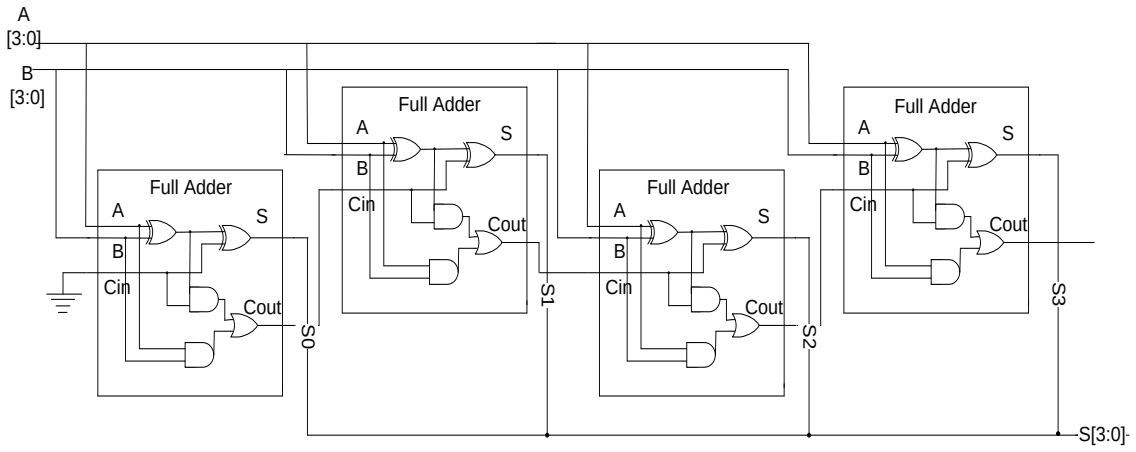

Figure 2.1 The block circuit diagram of the 4-bit RCA

2.2.2 Carry –Look-ahead Adder (CLA)

The adder is the element in PA. Therefore, a fast adder is needed to improve the performance of the accumulator. The carry-out function of the CLA used in each pipeline stage of PA design can be quickly determined with a value of 0 or 1 in each stage [8], and the result can be achieved in a shorter time. The carry-out functions of stage N are obtained from the following equation:

$$C_{N+1} = g_{N+1} + p_N.\ C_N \tag{2-1}$$

Where $g_N = x_N.y_N$ and $p_N = x_N + y_N$ are generate and propagate functions respectively, and *N* is the *FCW* bit.

The carry-out for one stage is expressed as follows:

$$C_1 = g_1 + p_0.\ C_0 \tag{2-2}$$

The carry-out for N bits is expressed as follows:

$$C_n = g_N + p_N.\ g_{N-1} + \ p_N.p_{N-1}g_{N-2} + \ldots\ldots\ g_0\ p_N.p_{N-1}\ p_{N-2}\ \ldots\ .\ p_0C_0 \tag{2-3}$$

The carry-out of the 8-bit CLA is obtained through the following equation:

$C_8 =$

$$G_7 + p_7 g_6 + p_7 p_6 g_5 + p_7 p_6 p_5 g_4 + p_7 p_6 g_5 p_4 g_3 + p_7 p_6 p_5 p_4 p_3 g_2 +$$

$$P_7 p_6 p_5 p_4 p_3 p_2 g_1 + p_7 p_6 p_5 p_4 p_3 p_2 p_1 g_0 + p_7 p_6 p_5 p_4 p_3 p_2 p_1 p_0 c_0$$

(2-4)

The block circuit diagram of the 8-bit RCA is shown in Figure 2.2

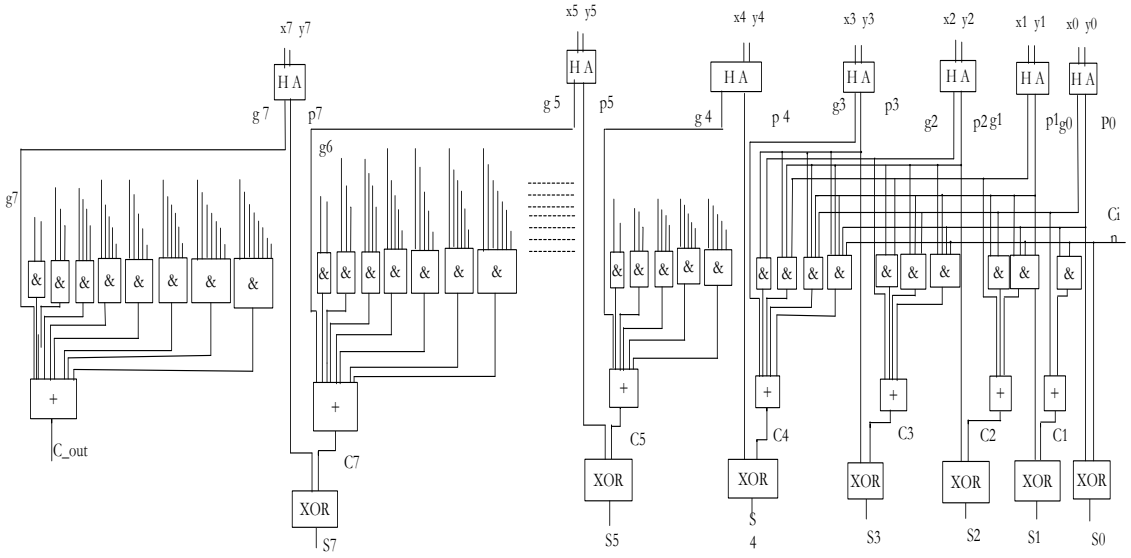

Figure 2.2 The block circuit diagram of 8-bit CLA

2.2.3 Kogge-Stone Adder

Kogge-Stone adder consists of groups of *p* and g cells, and the arithmetic operation is given in:

$$P_i = x_i + y_i \quad , \; g_i = x_i \,.\, y_i \tag{2-5}$$

The internal carriers Cn:1 of the 4-bits KS adder are given in the following Equations:

$$C_1 = g_0 + p_0 C_{in} \tag{2-6}$$

$$C_2 = (g_1 + p1.g_0) + p_1.p_0 c_{in} \tag{2-7}$$

$$C_3 = (g_2 + p_2.g_1\ ) + p_2\ .\ p_1c_1 \qquad (2\text{-}8)$$

The sum and carry out of the 4-bit KS adder are shown in the following equations

$$S_0\ = p_0 + c_{in} \qquad (2\text{-}9)$$

$$S_{N:1} = (g_{3\,+}\, p_{3\,.}\, g_2\,) + p_3\,.\,p_2\,C_2 \qquad (2\text{-}10)$$

The block circuit diagram of 4-bit Kogge-Stone adder in Figure 2.3 shown groups of *p* and g cells. These cells connect to achieve the summation outputs of the parallel prefix tree with minimum delay.

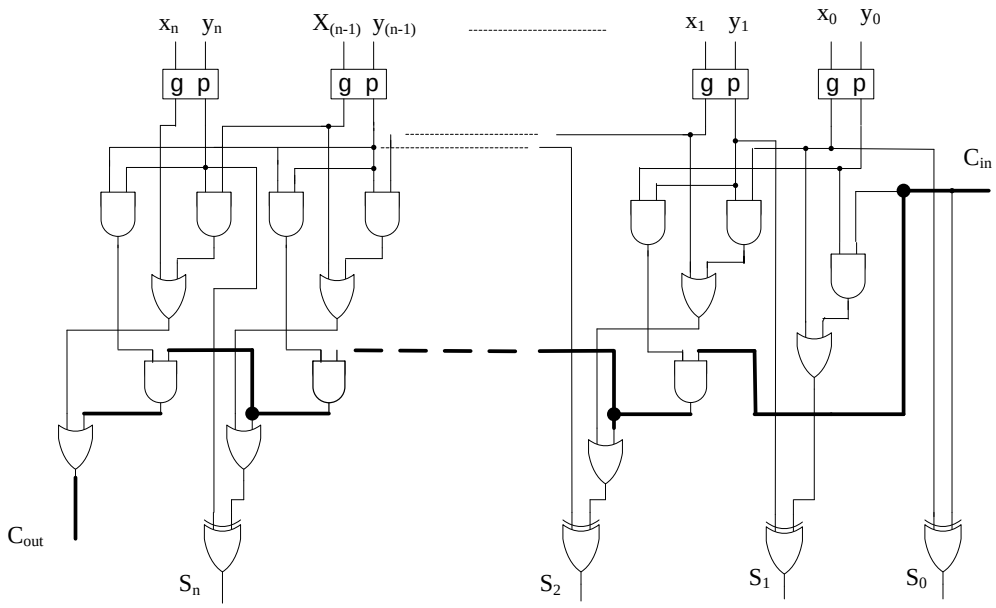

Figure 2.3 Sample of prefix adder architecture

2.3 N-Bit D Flip Flop Register

Shift registers are a type of sequential logic circuit, mainly for storage of digital data. They are a group of flip-flops connected in a chain so that the output from one flip-flop becomes the input of the next flip-flop.

A basic N-bit DFF register can be constructed using four D flip-flops, as shown in Figure 2.4.

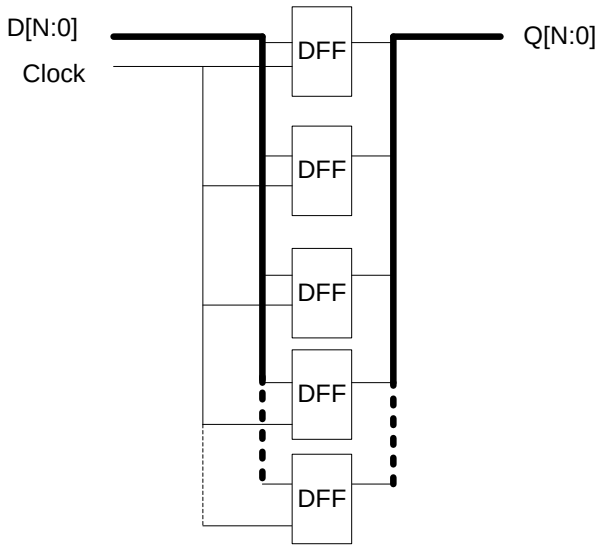

Figure 2.4 N-bit DFF register

CHAPTER THREE

RESULT AND DISCUSSION

3.1 Introduction

The conventional ripple carry adder (RCA), proposed CLA adder, and the KS adders have been used in the accumulator design. A comparison has been made between a conventional adder (RCA) and the other adders for 4-bits, 6-bits and 8-bits. The comparison result shows that the proposed CLA and KS adder which used in the designing of the proposed 4-, 6- and 8-bit PAs is relatively faster than the RCA and KS adder.

3.2 ALTERA QUARTUS II

Altera Quartus II is a programmable logic device design software produced by Altera. Quartus II enables analysis and synthesis of HDL designs, which enables the developer to compile their designs, perform timing analysis, examine RTL diagrams, simulate a design's reaction to different stimuli, and configure the target device with the programmer. Quartus includes an implementation of VHDL and Verilog for hardware description, visual editing of logic circuits, and vector waveform simulation.

Quartus II software features include:

1. SOPC Builder (System on a Programmable Chip Builder) , a tool in Quartus II software that eliminates manual system integration tasks by automatically generating interconnect logic and creating a testbench to verify functionality

2. Qsys, a system-integration tool that is the next generation of SOPC Builder. It uses an FPGA-optimized network-on-chip architecture that doubles the F_{MAX} performance vs. SOPC Builder.

3. SoCEDS, a set of development tools, utility programs, run-time software, and application examples to help you develop software for SoC FPGA embedded systems.

4. DSP Builder, a tool that creates a seamless bridge between the MATLAB/Simulink tool and Quartus II software, so FPGA designers have the algorithm development, simulation, and verification capabilities of MATLAB/Simulink system-level design tools

5. External memory interface toolkit, which identifies calibration issues and measures the margins for each DQS signal.

6. Generation of JAM/STAPL files for JTAG in-circuit device programmers.

3.3 Hardware Description Language (HDL)

The proposed 4-bit PA design consists of 4-bit ripple carry adder and 4-bit DFF. Each of the named blocks was codded Verilog HDL code.

The Verilog HDL codes of the 4-bit ripple carry adder, full adder, 4-bit DFF are listed in the following:

3.4 4-Bit RCA Adder Verilog HDL Code

The 4-bit RCA, CLA and KS adders have been codded Verilog HDL code based on. The 4-bit RCA adder and a full adder have been codded Verilog HDL code using QurtusII Software, as in following:

```
module Ripple_4
(sum,c_4,a,b,c0);
input [3:0]a,b;
input c0;
output[3:0]sum;
output c_4;
wire [3:0] g;
wire [3:0] p;
wire [4:0] c;
assign p[3:0]=a[3:0]^b[3:0];
assign g[3:0]=a[3:0]&b[3:0];
assign
c[1]=g[0]|(p[0]&c[0]),
c[2]=g[1]|(p[1]&c[1]),
c[3]=g[2]|(p[2]&c[2]),
c[4]=g[3]|(p[3]&c[3]);
assign
sum[3:0]=p[3:0]^c[3:0],
c_4=c[4];
endmodule
```

The 4-bit CLA adder and a full adder have been codded Verilog HDL code using QurtusII Software, as in following:

```
module CLA_4_bit
(sum,c_4,a,b,c0);
input [3:0]a,b;
input c0;
output[3:0]sum;
output c_4;

wire [3:0] g;
wire [3:0] p;
wire [4:0] c;
assign p[3:0]=a[3:0]^b[3:0];
assign g[3:0]=a[3:0]&b[3:0];
assign
c[1]=g[0]|(p[0]&c[0]),
c[2]=g[1]|(p[1]&g[0])|(p[1]&p[0]&c[0]),
c[3]=g[2]|(p[2]&g[1])|(p[2]&p[1]&g[0])|(p[2]&p[1]&p[0]&c[0]),
c[4]=g[3]|(p[3]&g[2])|(p[3]&p[2]&g[1])|(p[3]&p[2]&p[1]&g[0])|(p[3]&p[
2]&p[1]&p[0]&c[0]);
assign
```

```
sum[3:0]=p[3:0]^c[3:0],

c_4=c[4];

endmodule
```

The 4-bit KS adder and a full adder have been codded Verilog HDL code using QurtusII Software, as in following:

```
module k_s_4

( x ,y  , c_out ,c_in, sum );

// Inputs and outputs

input[3:0] x;

input[3:0] y;

input c_in ;

output  [3:0]sum ;

output  c_out;

// Internal wires

wire [3:0] sum;

wire c_out;

wire [3:0] g;

wire [3:0] p;

assign  g[3:0]=x[3:0] & y[3:0];   //carry generation

assign  p[3:0]=x[3:0] ^ y[3:0];   //carry propagation
```

```
// compute the carry for each stage

 assign  sum[0]= p[0] ^ c_in;

 assign  sum[1]= p[1] ^ (g[0] | ( p[0] & c_in )) ;

 assign  sum[2]= p[2] ^ (((p[1] & p[0]) & c_in ) | (g[1] | (p[1] & g[0] )));

 assign  sum[3]= p[3] ^ (((p[2] & p[1]) & (g[0] | ( p[0] & c_in ))) | (g[2] |
(p[2] & g[1] )));

 assign  c_out = ((g[3] | (p[3]& g[2])) |(( p[3] & p[2]) & (((p[1] & p[0]) &
c_in ) | (g[1] | (p[1] & g[0] )))));


endmodule
```

3.5 4-bit D Flip Flop Register

The Verilog code of 4-bit DFF registers as in following:

```
module D_4
(D,clK,Reset,Q );
input [3:0]D;
input clK ,Reset;
output [3:0]Q;
reg [3:0]Q;

always@( posedge clK or posedge Reset)
begin
if (Reset)
```

```
Q <= 1;
else
    Q <= D;
end
endmodule
```

The 4-bit ripple carry adder was elaborated and synthesized using ALTERA Qurtus II software, The register transfer logic (RTL) viewer of 4-bit ripple carry adder is shown in Figure 3.1.

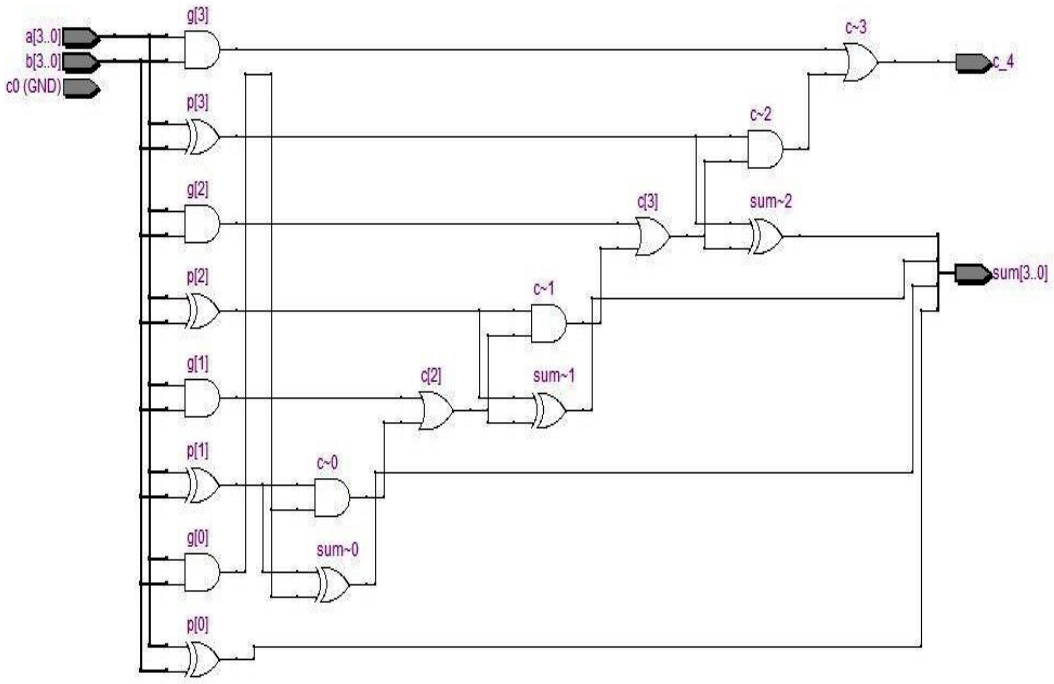

Figure 3.1 The RTL viewer of 4-bit ripple carry adder

3.6 4-bit Phase Accumulator Architecture

The proposed 4-bit PA design including the 4-bit DFFs and 4-bit adder was coded in Verilog HDL code. The Verilog HDL code of the complete 4-bit phase accumulator design was written as in following:

```
module PA_4_bit_CLA   (
   Reset,
   clk,
   D,
  Q
    );
input wire    Reset;
input wire    clk;
input wire    [3:0] D;
output wire  [3:0] Q;
wire   [3:0] SYNTHESIZED_WIRE_0;
wire   [3:0] SYNTHESIZED_WIRE_1;
wire   [3:0] SYNTHESIZED_WIRE_2;
assignQ = SYNTHESIZED_WIRE_2;

DFF_4        b2v_inst(
        .clK(clk),
        .Reset(Reset),
        .D(D),
        .Q(SYNTHESIZED_WIRE_1));
```

```
DFF_4        b2v_inst1(
        .clK(clk),
        .Reset(Reset),
        .D(SYNTHESIZED_WIRE_0),
        .Q(SYNTHESIZED_WIRE_2));

RIPPLE _4  b2v_inst3(

        .x(SYNTHESIZED_WIRE_1),
        .y(SYNTHESIZED_WIRE_2),

        .sum(SYNTHESIZED_WIRE_0));
endmodule
```

The Verilog HDL code of the suggested 4-bit PA design was successfully analysis and synthesizes using ALTERA Qurtus II software.

The The generated register transfer logic of the proposed 4-bit phase accumulator project is shown in Figure 3.2:

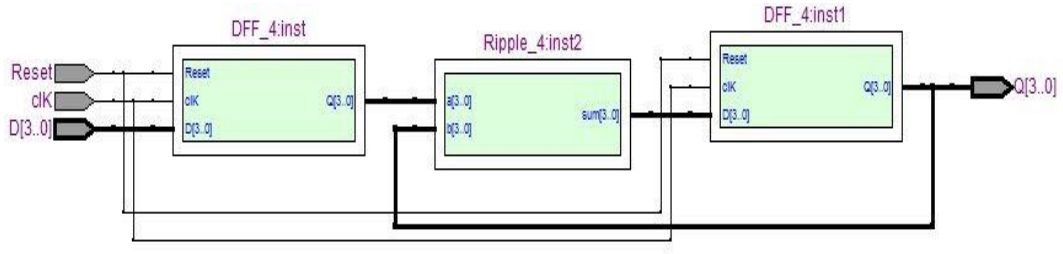

Figure 3.2 The RTL viewer of 4-bit phase accumulator

Same principal is used for the rest of the other 4-bit CLA and KS adder

3.7 6-bit CLA Adder Verilog HDL Code

The 6-bit CLA adder has been codded Verilog HDL code using QurtusII Software. The Verilog HDL code of the 6-bit CLA adder was written as in following:

```
module cla_6_PA(sum,c_6,a,b,c0);

input [5:0]a,b;

input c0;

output[5:0]sum;

output c_6;


wire [5:0] g;

wire [5:0] p;

wire [6:0] c;
```

```
assign p[5:0]=a[5:0]^b[5:0];

assign g[5:0]=a[5:0]&b[5:0];

assign

c[1]=g[0]|(p[0]&c[0]),

c[2]=g[1]|(p[1]&g[0])|(p[1]&p[0]&c[0]),

c[3]=g[2]|(p[2]&g[1])|(p[2]&p[1]&g[0])|(p[2]&p[1]&p[0]&c[0]),

c[4]=g[3]|(p[3]&g[2])|(p[3]&p[2]&g[1])|(p[3]&p[2]&p[1]&g[0])|(p[3]&p[
2]&p[1]&p[0]&c[0]),

c[5]=g[4]|(p[4]&g[3])|(p[4]&p[3]&g[2])|(p[4]&p[3]&p[2]&g[1])|(p[4]&p[
3]&p[2]&p[1]&g[0])|(p[4]&p[3]&p[2]&p[1]&p[0]&c[0]),

c[6]=g[5]|(p[5]&g[4])|(p[5]&p[4]&g[3])|(p[5]&p[4]&p[3]&g[2])|(p[5]&p[
4]&p[3]&p[2]&g[1])|(p[5]&p[4]&p[3]&p[2]&p[1]&g[0])|(p[5]&p[4]&p[3]
&p[2]&p[1]&p[0]&c[0]);


assign

sum[5:0]=p[5:0]^c[5:0],

c_6=c[6];

endmodule
```

The register transfer logic (RTL) viewer of the 6-bit CLA adder design is shown in Figure 3.3.

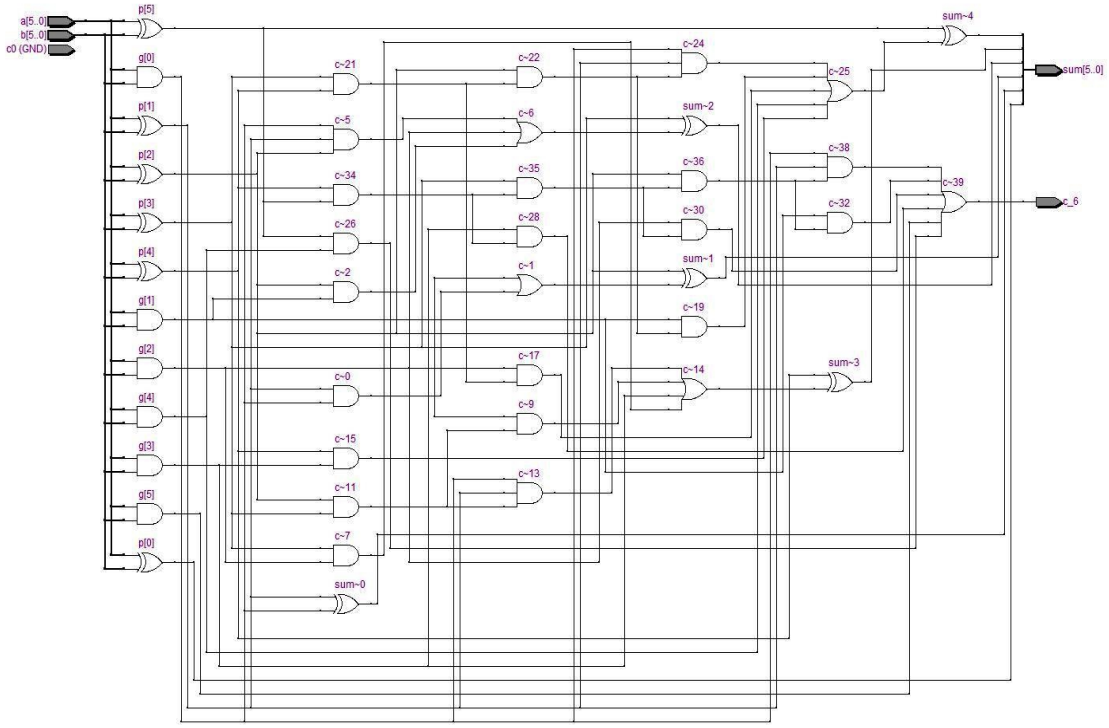

Figure 3.3 The RTL viewer of 6-bit CLA adder

Same principal is used for the rest of the other 6-bit RCA and KS adder, whereas the HDL code of 8-bit Kogge-Stone adder. The Verilog HDL code of the 8-bit Kogge-Stone adder was written as in following:

```
module koggo_stone_8
( x ,y  , c_out , c_in  , sum );
// Inputs and outputs
input[7:0] x;
input[7:0] y;
input c_in ;
output  [7:0]sum ;
```

```
output  c_out;

// Internal wires

wire [7:0] sum;

wire c_out;

wire M[5:0];

wire k[6:0];

wire C[7:1];

wire [7:1] g;

wire [7:0] p;

wire c;

assign  g[7:1]=x[7:1] & y[7:1];   //carry generation

assign  p[7:0]=x[7:0] ^ y[7:0];   //carry propagation

assign M[0] = p[2] & p[1] ;

assign M[1] = p[3] & p[2] ;

assign M[2] = p[4] & p[3] ;

assign M[3] = p[5] & p[4] ;

assign M[4] = p[6] & p[5] ;

assign M[5] = p[7] & p[6] ;

assign k[0] = g[1] |( p[1] & c );

assign k[1] = g[2] |( p[2] & g[1] );

assign k[2] = g[3] |( p[3] & g[2] );
```

```
assign k[3] = g[4] |( p[4] & g[3] );

assign k[4] = g[5] |( p[5] & g[4] );

assign k[5] = g[6] |( p[6] & g[5] );

assign k[6] = g[7] |( p[7] & g[6] );

assign C[1] = c ;

assign C[2] = k[0] ;

assign C[3] = k[1] | (M[0] & c) ;

assign C[4] = k[2] | (M[1] & k[0]) ;

assign C[5] = k[3] | (M[2] & C[3]) ;

assign C[6] = k[4] | (M[3] & C[4]) ;

assign C[7] = k[5] | (M[4] & C[5]) ;

 assign  sum[0]= p[0] ^  c_in;

 assign  sum[1]= p[1] ^  C[1] ;

 assign  sum[2]= p[2] ^  C[2];

 assign  sum[3]= p[3] ^  C[3];

 assign  sum[4]= p[4] ^  C[4];

 assign  sum[5]= p[5] ^  C[5];

 assign  sum[6]= p[6] ^  C[6];

 assign  sum[7]= p[7] ^  C[7];

     assign c_out =  k[6] | (M[5] & C[6]);
```

endmodule

The register transfer logic (RTL) viewer of the suggested 8-bit KS adder design is shown in Figure 3. 4.

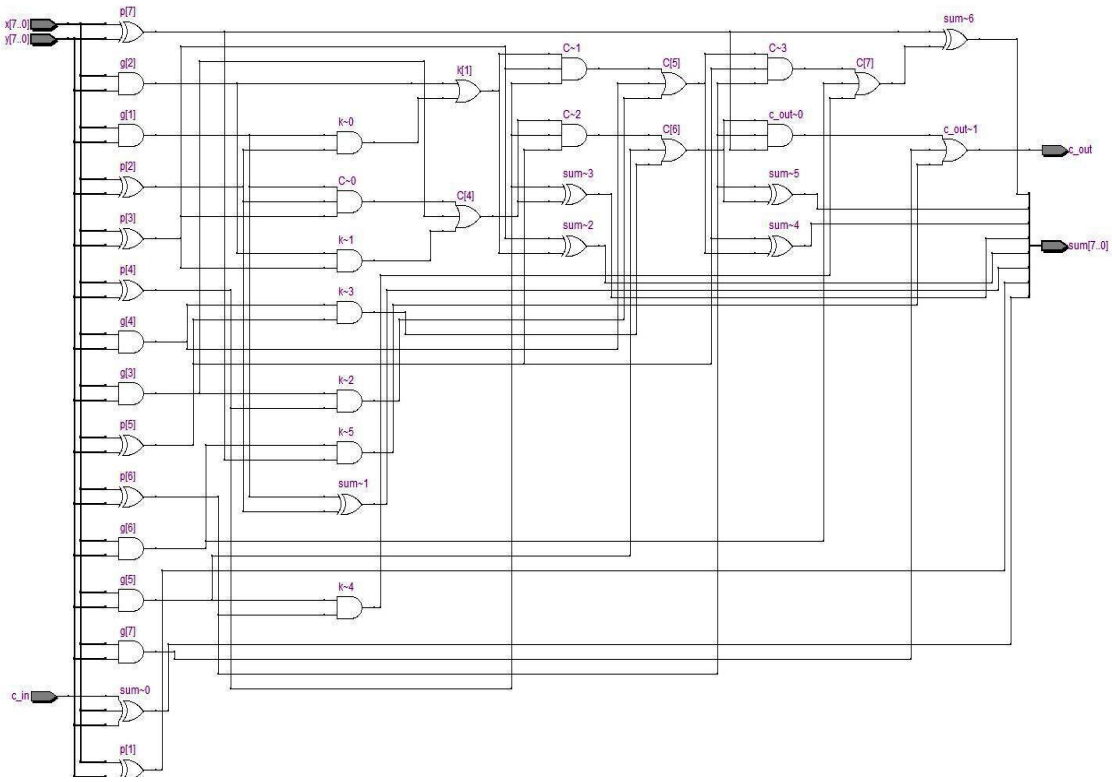

Figure 3.4 The RTL viewer of 8-bit KS adder

3.8 Phase Accumulator Comparison Based on Fast Adders

A comparison of different adder architectures was applied in a phase accumulator design using multiple tuning word input (4-bits, 6-bits, and 8-bits) to select the fast adder for high-speed phase accumulator. The comparison Table shown in Table 3.1

Adder	4-bit (MHZ)	6-bit (MHZ)	8-bit (MHZ)
RCA	619.5	474.1	368.2
CLA	736.9	560.5	421
KS	658.7	520	412.2

Table 3.1 the comparison of 4-, 6-, and 8-bit phase accumulator based on RCA, CLA and KS adder

The comparison results of the 4-, 6-, and 8-bit phase accumulators designed based on RCA, CLA and KS adder in Table 3.1. The Table result shows that the proposed CLA, adder was relatively faster than the other types of adders.

3.9 Simulation of the Proposed Phase Accumulator Design

The proposed 4-bit, 6-bit and 8-bit Pas design were synthesized, elaborated and compiled in ALTERA Quartus II software system to functionally verify the system design. The gate level simulation of the proposed 4-, 6-, and 8-bit phase accumulators' designs shows the PA output in Figures 3.6, 3.7 and 3.8 respectively.

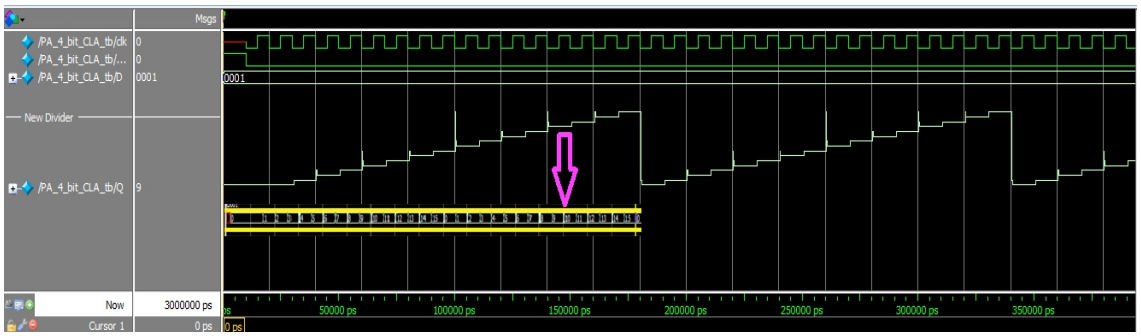

Figure 3.5 The gate level simulation of the 4-bit Phase Accumulator

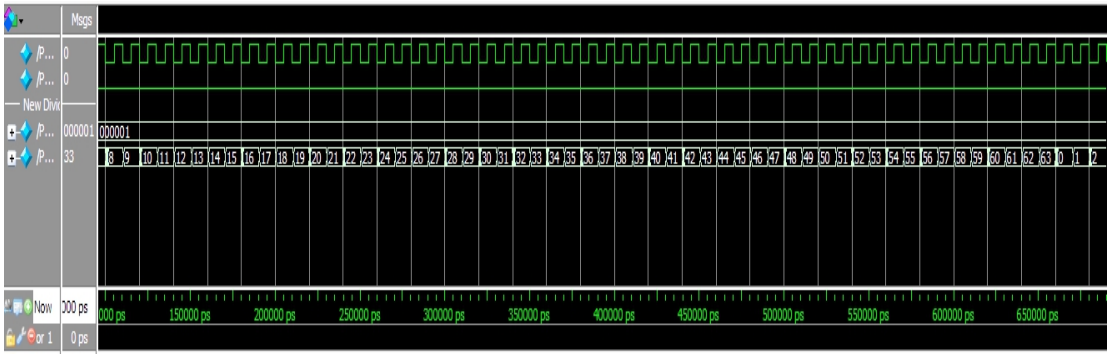

Figure 3.6 The gate level simulation of the 6-bit Phase Accumulator

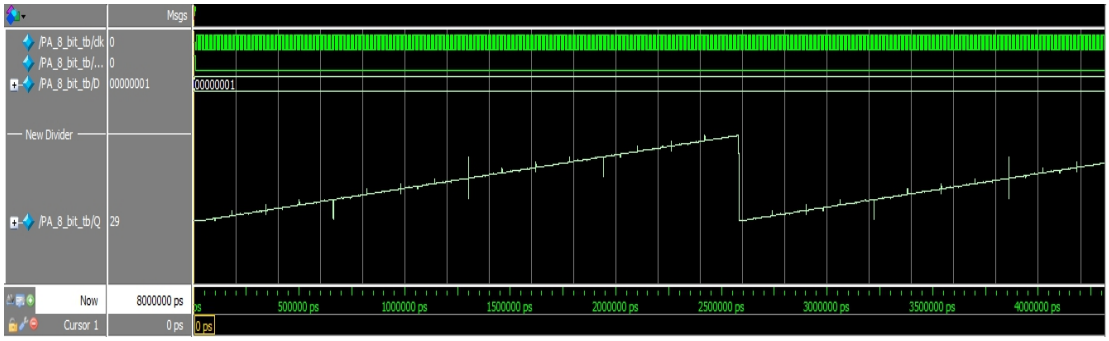

Figure 3.7 The gate level simulation of the 8-bit Phase Accumulator

CHAPTER FOUR
CONCLUSIONS AND FUTURE WORKS

4.1 Conclusion

Phase accumulator is a generator that generates the digital phase of (0 to 2π) range values. The PA architecture consists of adder and register.

This project proposed the architectures of phase accumulator based on RCA, CLA and KS adder. The projects have been designed using HDL code.

A comparison of different adder architectures was applied in the phase accumulator design using multiple tuning word input (4-bits, 6-bits, and 8-bits) to select the fast adder for high-speed pipelined PA. The result shows that the proposed CLA, adder has maximum frequency operation of 421, 560.5 and 736.9 MHz for 4-bits, 6-bits, and 8-bits PAs. Thus means that the CLA adder was relatively faster than the other types of adders.

The suggested PAs were coded in Verilog HDL and successfully simulated in ModelSim with ALTERA Quartus II software to prove the validity of the proposed design.

5.2 Future Works

It is suggested to implement the proposed phase accumulator design on the field programmable gate array (FPGA), to verify real time of the design project and compare the achieved result with the gate level simulation result, to prove the validity of the design.

REFERENCES

[1] D. B. Ken Gentile, Ted Harris (1999). *A Technical Tutorial on Digital Signal Synthesis, http://www.analog.com/UploadedFiles/Tutorials/450968421DDS_Tutorial_rev12-2-99.pdf*

[2] K. Yong Sin and K. Sung-Mo, "A High Speed Low-Power Accumulator for Direct Digital Frequency Synthesizer," in *Microwave Symposium Digest, 2006. IEEE MTT-S International*, 2006, pp. 502-505.

[3] G. Xueyang, F. F. Dai, J. D. Irwin, and R. C. Jaeger, "24-Bit 5.0 GHz Direct Digital Synthesizer RFIC With Direct Digital Modulations in 0.13 µm SiGe BiCMOS Technology," *Solid-State Circuits, IEEE Journal of*, vol. 45, pp. 944-954, 2010.

[4] S. H. Akhurwy, "AN AREA EFFICIENT READ ONLY MEMORY (ROM) DESIGN ARCHITECTURE FOR HIGH SPEED DIRECT DIGITAL FREQUENCY SYNTHESIZER," PhD, Department of Electrical, Electronics & System Engineering, UNIVERSITI KEBANGSAAN MALAYSIA, Malaysia, 2015.

[5] Y. G. K.Babulu, "Implementation and Performance Evaluation of Prefix Adders uing FPGAs," *IOSR Journal of VLSI and Signal Processing*, vol. 1, pp. PP 51-57, Sep-Oct. 2012 2012.

[6] A. M. N. Hassan, "The fractional N frequency synthesizer," *Journal of Engineering*, vol. 12, 2006. [7] U. o. California. (2009). *Adder* .

[8] Z. V. S. Brown, *Fundamental of digital logic with VHDL*, 2end edition ed.: Mc.Graw-Hill, 2005