FPGA Implementation of Modified ADPPL for Dual Clock Memory Design

Shivprasad.Mallappa.Channangi[1] Mr. Vinay Bagali[2]
[2]Assistant Professor
[1,2]Department of EEE
[1,2]BLDEA'S V.P.DR. P.G.H CET, Vijayapur, Karnataka, India

Abstract— In This Paper, We Propose A Novel Design Of Adpll Which Should Operate In Synchronous With Dual Clock Memory. The Main Blocks Of Adpll Such As Digital Controlled Oscillators Are Implemented With A New Method In Digital Domain For Better Performance. The Digital Controlled Oscillator Is Designed Using Modified Counters Which Reduce The Delay Compared To Traditional Methods. The Increment Decrement Counter Is Modified From Previous Work And Is Designed Using Mux. 200 Hz Is The Central Frequency And Operational Frequency Range Is 997.0089 Hz To 1008.64 Hz Which Is The Lock Range Of The Design. Further The Adpll Is Designed Using Vhdl Language, Synthesized Using Xilinx Ise 14.5 And Implemented Using Xilinx System Generator For Data Serialization Using Spartan 6 Lx45 Fpga Board.

Key words: DCO, ADPLL, FPGA, Loop filter, Phase Detector, Dual clock Memory

I. INTRODUCTION

The PLL is a self-correcting control system in which one signal chases another signal. PLL has four types i). Linear PLL ii).Digital Phase Locked Loop iii).All Digital Phase Locked Loop iv). Software PLL (SPLL).ADPLL takes input as only digital signals. Due to digital signal as input signal ADPLL offers

Various advantages over the different types of PLLs. Beginning of all digital phase-locked loops (ADPLL) started in 1980[1]. A new Digitally Controlled Oscillator (DCO) has been developed by researchers to obtain good phase and frequency error that was not implemented with 74HC297 IC[3], [2], [4].In 2006 double edge triggered D flip-flop as phase detector was proposed [5] .This design reduced 33% of power dissipation. In 2008 digital FM demodulator was proposed [6]. It was designed by VHDL. In 2009 frequency modulated modem was implemented on field programmable array (FPGA) [7]. In 2010 a field programmable array based linear ADPLL was proposed. This ADPLL used FPGA for implementation [8] .Recently an all-digital phase locked loop (ADPLL) having a fault detection of the input reference signal was modeled using Verilog hardware descriptive language (HDL) [9]

II. PROPOSED ADPLL ARCHITECTURE

The below figure 2.1 shows proposed ADPLL Architecture. The existing method of ID counter blocks were based on priority encoder or logical expression based Hardware. The proposed ID counter is based on multiplexer which reduces the delay of ID counter

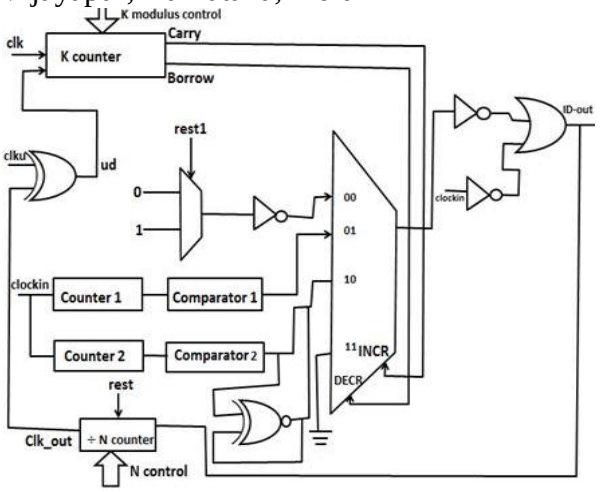

Fig. 1: Proposed ADPLL Architecture

A. Phase Detector

The phase detector is the first stage of ADPLL as shown in Fig 2.

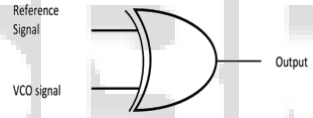

Fig. 2: EXOR gate

B. Loop Filter (K loop counter)

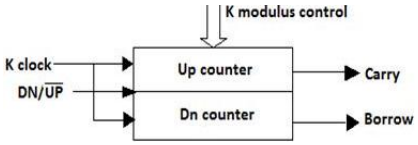

Fig. 3: Loop counter

The Loop counter consists of Up counter, down counter , carry and borrow are outputs.

C. Digital Controlled Oscillator

ADPLL contains another block as Digital Controlled Oscillator whose Block diagram is shown in below figure4

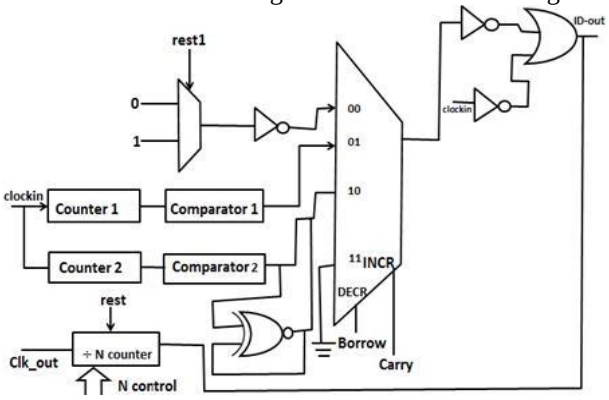

Fig. 4: Modified Digital Controlled Oscillator Block

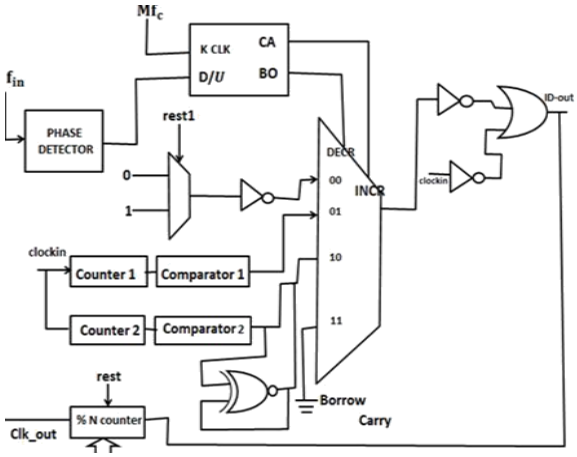

Fig. 5: Modified All Digital Phase Locked System

Parameters	ADPLL design
K	8
M	16
N	8
Central Frequency(f_0)	200 kHz

Table 1: parameter details of designed ADPLL

III. RESULTS AND ANALYSIS

A. Design Summary of ADPLL

ADPLL Project Status (09/28/2015 - 16:40:52)			
Project File:	ADPLL.xise	**Parser Errors:**	No Errors
Module Name:	ADPLL	**Implementation State:**	Synthesized
Target Device:	xc5vlx110t-2ff1738	• **Errors:**	No Errors
Product Version:	ISE 14.5	• **Warnings:**	8 Warnings (0 new)
Design Goal:	Balanced	• **Routing Results:**	
Design Strategy:	Xilinx Default (unlocked)	• **Timing Constraints:**	
Environment:	System Settings	• **Final Timing Score:**	

Device Utilization Summary (estimated values)			[-]
Logic Utilization	**Used**	**Available**	**Utilization**
Number of Slice Registers	42	69120	0%
Number of Slice LUTs	104	69120	0%
Number of fully used LUT-FF pairs	42	104	40%
Number of bonded IOBs	7	680	1%
Number of BUFG/BUFGCTRLs	2	32	6%

Detailed Reports					[-]
Report Name	**Status**	**Generated**	**Errors**	**Warnings**	**Infos**
Synthesis Report	Current	Mon Sep 28 16:54:20 2015	0	8 Warnings (0 new)	1 Info (0 new)
Translation Report					
Map Report					
Place and Route Report					
Power Report					
Post-PAR Static Timing Report					
Bitgen Report					

Secondary Reports		[-]
Report Name	**Status**	**Generated**
ISIM Simulator Log	Current	Wed Sep 30 19:20:24 2015

Date Generated: 10/02/2015 - 12:41:23

Fig. 6: Design summary of ADPLL

Parameters	Previous work	Present work
Combinational delay	12.997 ns	5.504 ns
IO utilization (bonded IOB)	4	6
Slice Logic Utilization (slice LUTs)	26	104
Slice Logic Distribution (FlipFlops)	7	62

Table 2: compares previous work and present work

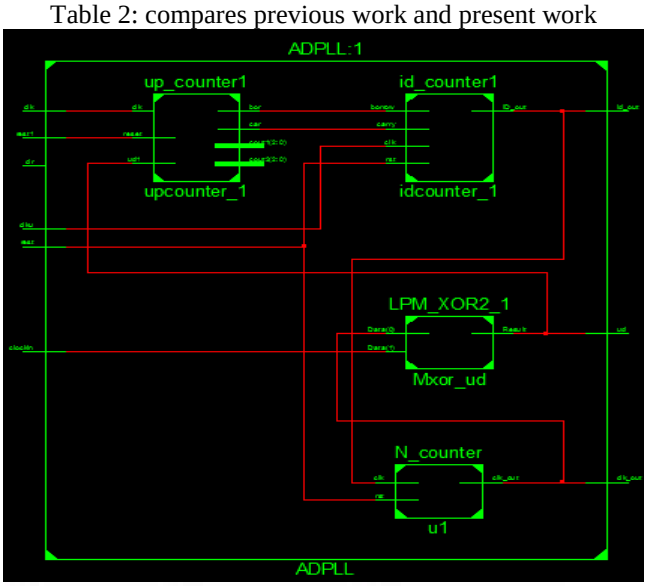

Fig. 7: RTL schematic of ADPLL

The above figure6 shows the RTL design of Proposed ADPLL

B. Simulation Results Of Proposed ADPLL

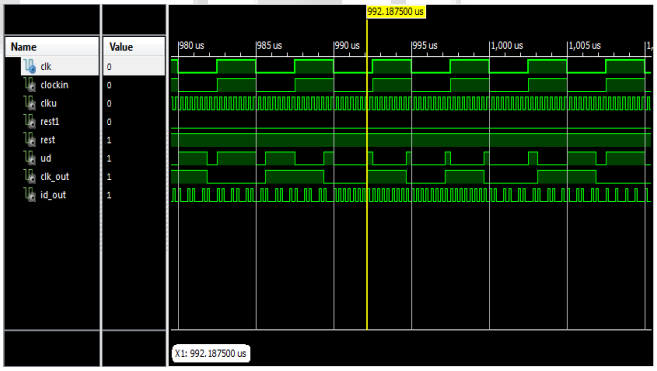

Fig. 8: simulation result of ADPLL

1) Measured ADPLL Parameters

a) LOCK RANGE: The Ability Of PLL To Lock For Certain Frequency Range

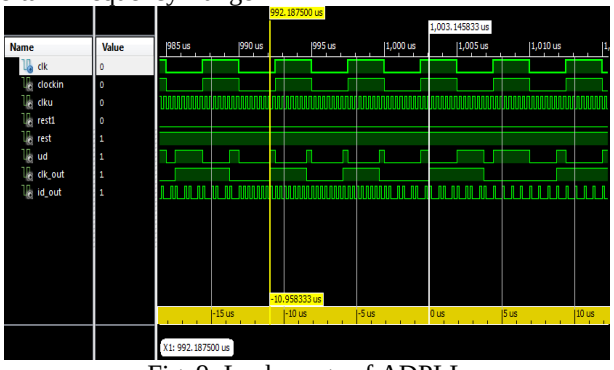

Fig. 9: Lock range of ADPLL

LOCK RANGE = 1003 µs – 992 µs = 11 µs

b) JITTER UNCERTAINITY: It Is Locking Uncertainty Expected Between Output And Input.

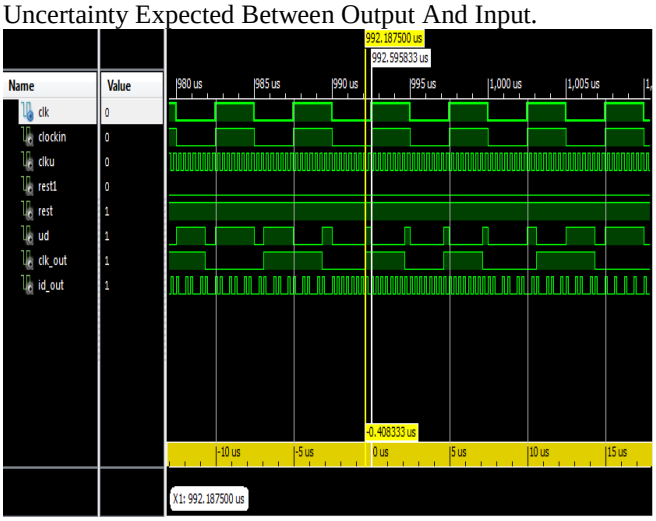

Fig. 10: Jitter Uncertainty of ADPLL

Jitter uncertainty = 992.595833 µs – 992.187µs = 0.408333 µs

c) Capture Range: The Ability of PLL to Lock – In From Unlock Condition

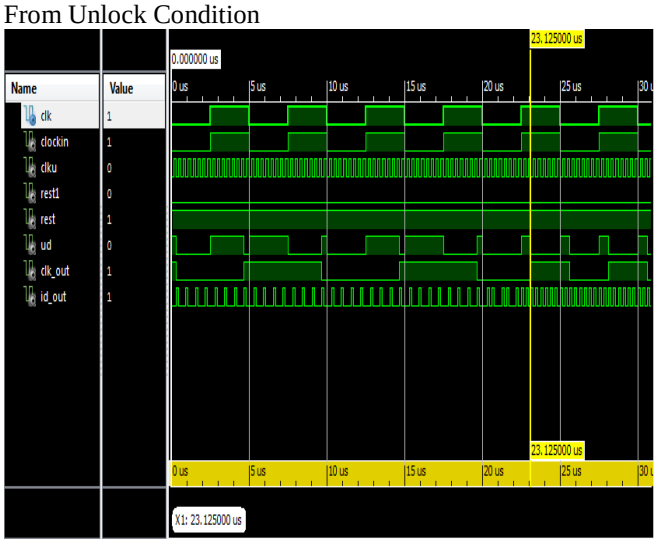

Fig. 11: Capture Range Of ADPLL

Capture range =23.125 µs – 0 µs = 23.125 µs

C. The System Generator Block for the ADPLL

The screenshot below describes the ADPLL functionality using the software model of the system generator

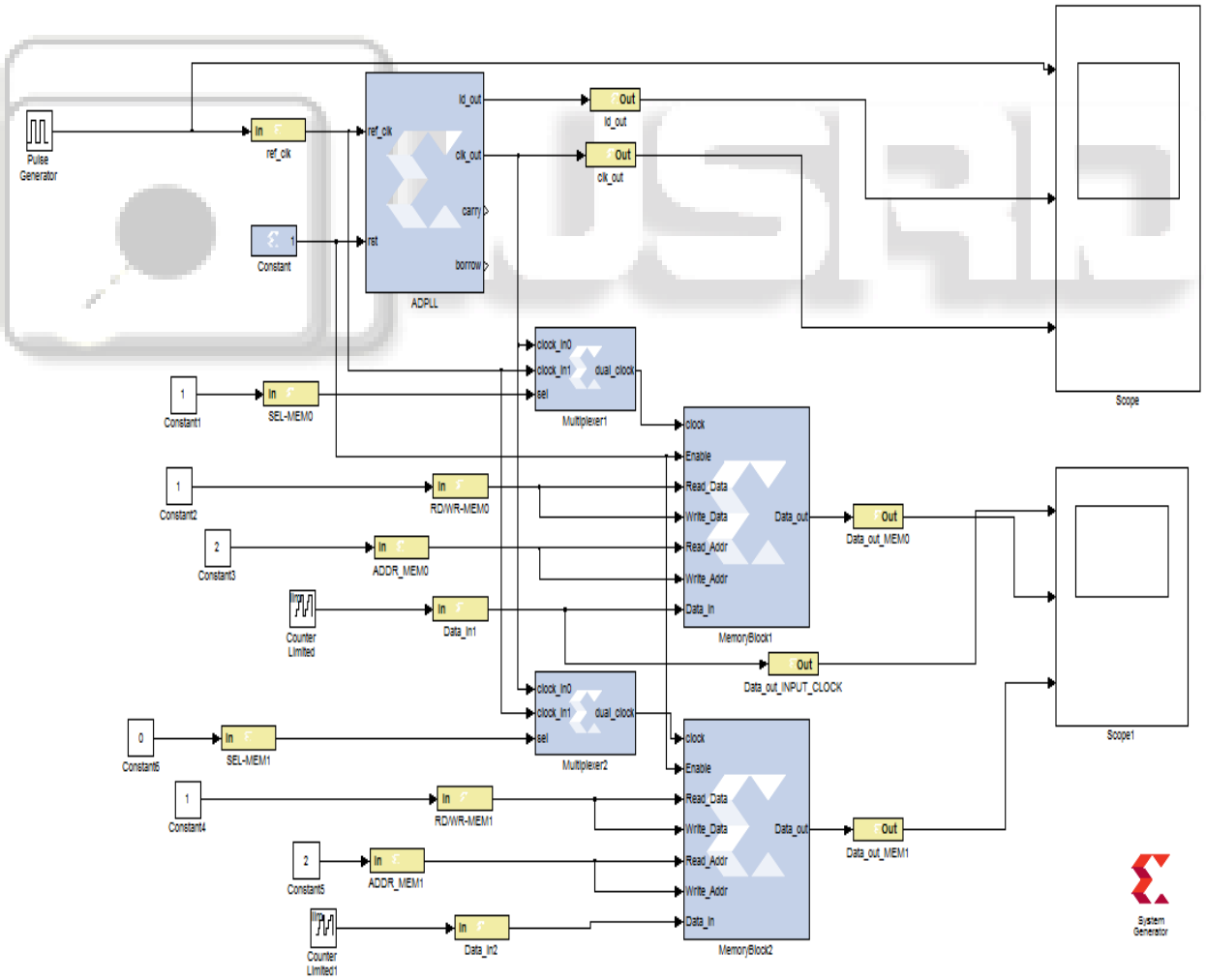

Fig. 12: software model of ADPLL from system generator

D. The Hardware Model for the Hardware-Co-Simulation

The screenshot2 given below is the hardware model obtained from the software model

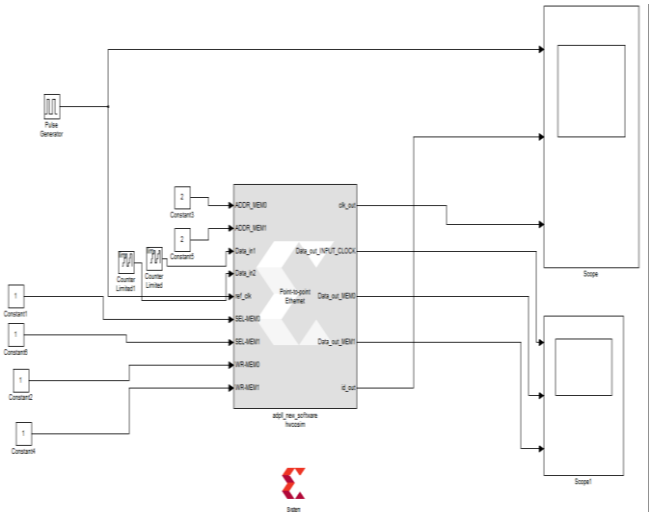

Fig. 13: Hardware model of ADPLL system generator

E. Simulation result through scope

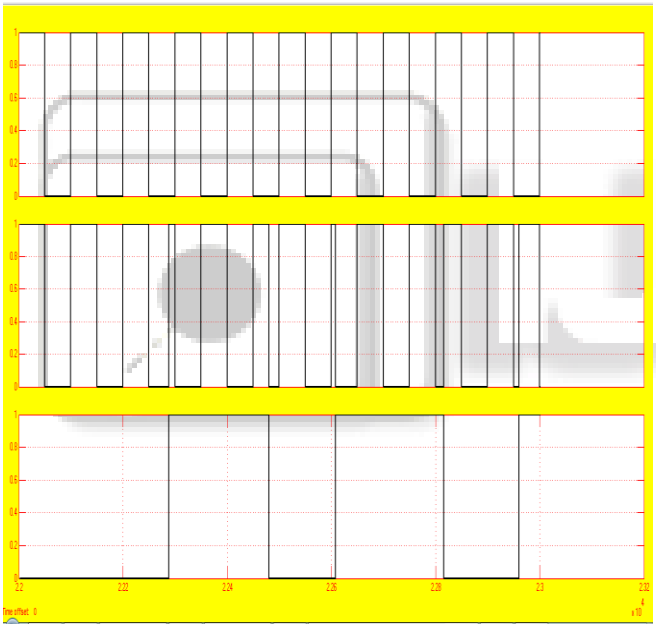

Fig. 14: output of ADPLL

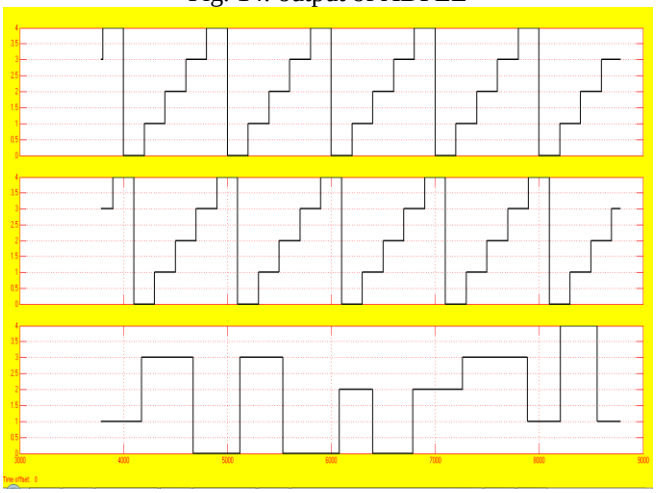

Fig. 15: Output of Dual Clock Memory

IV. CONCLUSION

In this work ADPLL is deliberated using VHDL and ISE CAD tool Xilinx 14.5 is used for simulation and system generator is used for implementation. It functions with Dual clock Memory. ADPLL is considered with central frequency of 200 kHz and operating frequency range of 997.0089 Hz to 1008.064516 Hz. Simulink tool is used for modeling and testing of the physical system. It has been concluded that the proposed method obtains better results if delay compared to existing methods.

REFERENCES

[1] In Lata, K, Kumar, M. "ADPLL design and implementation on FPGA",2013 international Conference on Intelligent Systems and Signal Processing (ISSP), 272 – 277, 1-2 March 2013.
[2] Manoj Kumar and Kusuma Lata, "ALL Digital Phase Locked Loop(ADPLL): A Survey", Proceedings of the 4th IEEE International Conference on Electronics Computer Technology (ICECT 2012),April 6-8 2012, Kanyakumari , India.
[3] Aniruddha Chandra, Lecture on "Phase Locked Loop", in winter school on VLSI Systems Design,ECE Department, NIT Durgapur, Jan. 2009.
[4] K.T.lbrahim, and A.E Salama, "Digital Of ADPLL for Good Phase and Frequency Trackig Performance", Nineteenth National Radio Science Conference,Alexandria,March 2002.
[5] Chang-Hong Shan, Zhong-ze Chen, "An All-Digital Phase Locked Loop Based on Double Edge Triggered Flip-flop", Proceedings of 8th IEEE International Conference on Solid State and Integrated Circuit Technology, (ICSICT-2006), Oct. 2006, Shanghai, China.
[6] R.E.Best, Phase Locked Loops Design Simulation and Applications, McGraw-Hill Professional, ch. 6, pp.205-246, 5th Edition, 2003.
[7] A. Babu and B. Daya, "All Digital Phase Locked Loop Design and Implementation," Project report, University of Florida, Gainesville, FL, 32608, USA.
[8] S. Vallabhaneni, S. Attri, N. Krishman, S. Sharma, and R. C. Chauhan. Design of an all- Design of an all Digital PLL core on FPGA[online] Available:http://klabs.org/mapld04/abstracts/sharma_a.doc
[9] http://web.ece.ucdavis.edu/~astill/dcfifo.html
[10] http://quartushelp.altera.com/13.1/mergedProjects/msgs/msgs.htm#msgs/winfer_ram_functionality_change_altsyncram_dual_clock.htm

