ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

*(An ISO 3297: 2007 Certified Organization)*

**Vol. 5, Issue 1, January 2016**

# Mixed Signal Modeling and Physical Layout Design of a Simple FPGA Architecture

Nafisa Nawal[1], Richard Victor Biswas[1], Md. Sazzad Hossain[1], S.M.Ehsan Ekramul[1]

Undergraduate Student, Dept. of EEE, American International University -Bangladesh, Banani, Dhaka, Bangladesh[1]

**ABSTRACT**: FPGAs have become one of the dominant implementation technologies for digital circuits today. Even though FPGAs are digital in nature, due to the presence of clock management units (and in some cases ADCs and DACs), which are generally analog in nature, FPGA architectures are mixed signal circuits to be precise. We modeled logic cells and interconnect in Verilog and ADC, DAC and Phase Locked Loop (PLL) for clock management in Verilog-AMS (mixed signal version of Verilog) using the free Verilog-AMS simulator SMASH (from Dolphin Integration). Initially we wanted physical layout to be generated from AMS models. However, due to unavailability of suitable tools, we modeled and simulated these individual components (both digital and mixed signals blocks) using schematic tools such as DSCH3.5 and then generate physical layouts for these components from the free layout tool Microwind 3.5.

**KEYWORDS:** Mixed Signal, Verilog-AMS, FPGA.

## I.INTRODUCTION

Field-Programmable Gate-Arrays (FPGA) are well established for implementing programmable digital logic. However, traditional manufactures did not used to integrate analogue interfaces in their device and all interfaces between FPGA and analog signal were done by means of ADC and DAC. But recent FPGAs have implemented ADC and DAC on FPGAs for several reasons, such as, cost, board space and reduced number of components. Besides this, an FPGA has clock management unit that includes Phase Locked Loop (PLL) or Delay Locked Loop (DLL) which are analog in nature. For that reason, a modern FPGA can be considered as an analogue mixed signal circuit.

For modelling an analogue mixed signal circuit Verilog-AMS could be a great choice for designers. This is because AMS languages and simulators provide better integration among digital and analogue segments, reduce the number of bugs and improve debugging and simulation time.

First major components of FPGA i.e. ADC, DAC, CLB, Programmable Interconnection (PI), PLL, DLL were simulated using Verilog-AMS. After that, schematic diagram and layout of those components were created using DSCH and Microwind respectively.

## II.ARCHITECTURE OF AN FPGA

A typical FPGA contains thousands of logic blocks which are connected through a reconfigurable routing system. This routing system is used for wiring the logic blocks together to constitute user defined combination or sequential circuits. Programmable interconnection is employed to interface the logic blocks and routing architectures to the wide range of components in FPGA. Beside these, modern FPGAs have been developed through the addition of some special block such as Block RAM, multiplier, ADC, DAC. Moreover, clock management circuit such as DLL or PLL.

## III.ADC AND DAC MODELING

An analog to digital converter (ADC) is a device that converts a continuous time domain and continuous value signal to digital number. A flash ADC contains several numbers of resistors, comparators and a priority encoder. Resistors are used for changing the level of reference voltage and comparators are used to compare input voltage with different level of reference voltage. Binary weighted Digital to Analog Converter (DAC) comprises of an Op-amp and some resistors. Binary inputs are connected thorough the resistors and the entire input currents are summed together and amplified with the Op-Amp.

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

*(An ISO 3297: 2007 Certified Organization)*

**Vol. 5, Issue 1, January 2016**

Table 1: Verilog-AMS code of a 3-bit flash ADC and DAC

| Verilog-AMS code of a 3 bits flash ADC: | Verilog-AMS code of a Binary-Weight inputs DAC |
|---|---|
| always @(posedge clk)<br>begin<br>sample = V(in);<br>new_ref = ref*0.72;<br>for (i = 0; i <=7; i = i + 1)<br>begin<br>if(sample > new_ref)<br>begin<br>plot_out[i] <= #(td) 1;<br>sample = sample - new_ref<br>end<br>else plot_out[i] <= #(td) 0;<br>end<br>end | always @(clk) begin<br>if (clk == idir)<br>begin<br>aout=0.0;<br>weight = 2;<br>for (i=bits-1; i>=0; i=i-1)<br>begin<br>if (in[i])<br>aout = aout + fullscale / weight; weight = weight * 2;<br>end<br>result = #(td / 1.0E-9) aout;<br>end<br>end<br>analog begin<br>V(out) <+ result;<br>end |

## IV.CLOCK MANAGEMENT UNIT

In Field Programmable Gate Arrays (FPGA's), clock skew problems can be eliminated by clock management circuits such as Phase Locked Loops (PLL's) and Delay Locked Loops (DLL's). Shifting, de-skewing and frequency synthesis functions are performed by typical Clock management circuits.

### 1. Phase Locked Loops (PLL's):

The Phase Locked Loop (PLL) is a feedback system which locks output clock frequency with reference clock frequency by adding a voltage controlled oscillator and a phase detector. Main Parts of PLL includes Phase Frequency Detector (PFD), Charge Pump (CP), Low pass Filter of second order (LPF), Voltage controlled Oscillator (VCO) and frequency divider (FD).

Verilog-AMS code of these blocks inside a PLL is provided in table 2.The blocks were integrated and verified in simulation as shown in Fig.1.

Table 2: Verilog-AMS code of Phase-Frequency Detector & Charge-Pump, Voltage-Controlled Oscillator

| Verilog-Ams code for Phase-Frequency Detector & Charge-Pump | Verilog-Ams code for Voltage-Controlled Oscillator |
|---|---|
| Analog<br>begin<br>@(initial_step) state=0;<br>@(cross(V(ref)- vth, dir))<br>if (state > -1) state = state - 1;<br>@(cross(V(fb)- vth, dir))<br>if (state < 1) state = state + 1;<br>I(out) <+ transition( iout * (V(fb)-V(ref)), td, tt);<br>end | analog<br>begin<br>inst_freq = center_freq + vco_gain * V(vin);<br>$bound_step(1/(steps_per_period*inst_freq));<br>phase = idtmod (inst_freq, 0, 1);<br>V(vout)<+ amp * sin(2 * `M_PI * phase);<br>end |

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

*(An ISO 3297: 2007 Certified Organization)*

**Vol. 5, Issue 1, January 2016**

Table 2: Verilog-AMS code of Frequency Divider

| **Verilog-AMS code for Frequency Divider** |
|---|
| analog begin<br>@(initial_step) x= 0;<br>@(cross(V(clk) - vth, +1))<br>begin<br>if (V(j) >vth&& V(k) >vth) x = !x ;<br>else if (V(j) >vth&& V(k) <vth) x = 1;<br>else if (V(j) <vth&& V(k) >vth) x = 0;<br>else if (V(j) <vth&& V(k) <vth) x = x; end<br>V(q) <+ transition (v_high * x+v_low * !x, td,tr); end |

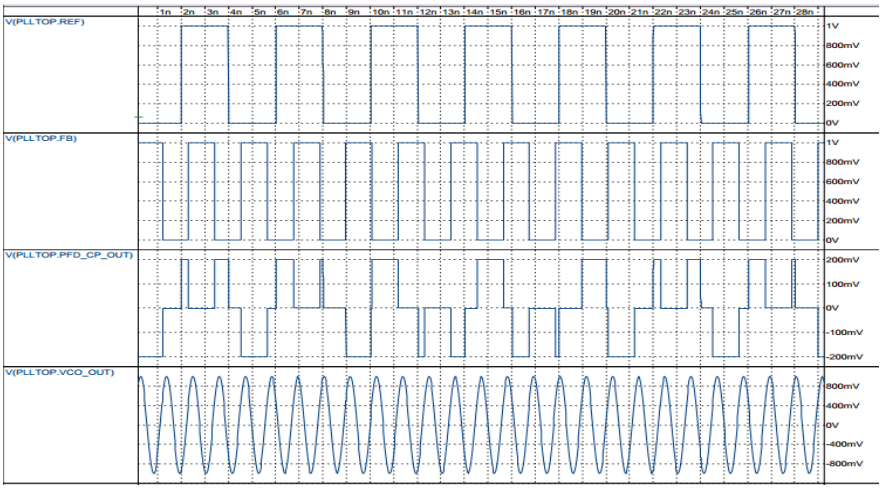


Fig. 1 Transient Analysis of PLL using Verilog-AMS in SMASH.

From the Waveforms of each element of PLL, it is evident that frequency of Reference Clock is lower than feedback divided clock. Also, 200mV can be denoted as "Up" and -200mV can be denoted as "DOWN" in PFD_CP_OUT waveform. At 1ns, reference clock leads feedback clock. Therefore, "DOWN" signal passed. Between 1ns and 2ns, 0V signal emerged because reference clock and feedback clock are in same phase. At this very moment, Locking is being ensured by PLL. At 2ns, feedback clock leads reference clock. Therefore, "UP" signal passed.VCO produces one analog signal and one digital signal. Digital signal is fed to the FD and analog signal is passed to RF modules. Pulse width and characteristic parameters of these two signals are different. Dividing the frequency of digital clock signal makes pulse width of this signal equal to the analog signal. When phase of reference clock and feedback clock is same, such phenomenon is seen.
Finally, approximately at 6.7ns feedback clock frequency and reference clock frequency are same and from this point locking phase starts.

## 2. Delay Locked Loops (DLLs)

The Counter-controlled DLL comprises of Phase Detector (PD), N-bit UP/DOWN Counter and Delay-line. According to the output error signal of phase detector, UP/DOWN counter changes its value. The counter sends count signal into delay line until the output clock synchronizes with input clock. The schematic diagram of a DLL using DSCH is given below.

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

*(An ISO 3297: 2007 Certified Organization)*

**Vol. 5, Issue 1, January 2016**

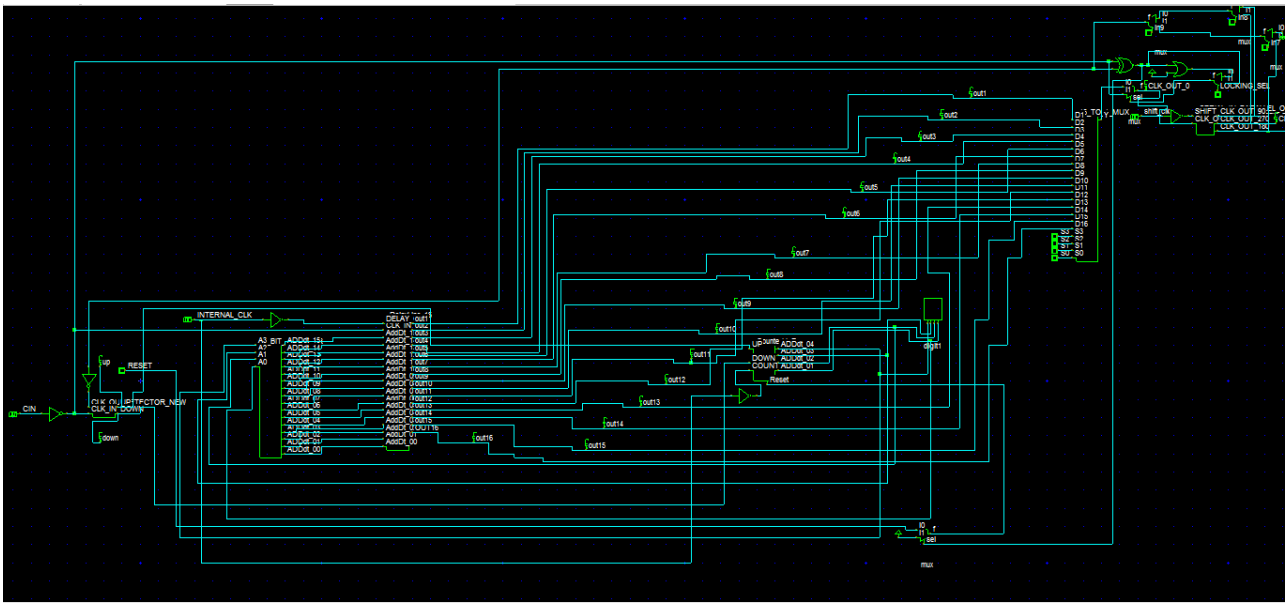


Fig. 2 Schematic diagram of a DLL

## V. CONFIGURABLE LOGIC BLOCK AND PROGRAMMABLE INTERCONNECT

A configurable logic block (CLB) is the building block of an FPGA that provides basic logic and storage functionality for target design. There are thousands to millions of CLBs integrated in an FPGA. In general, a CLB consists of one or more logical cells called basic logic element (BLE). A CLB can comprise of (BLE), or a cluster of interconnected BLEs. A simple BLE consists of a LUT, and a Flip-Flop.

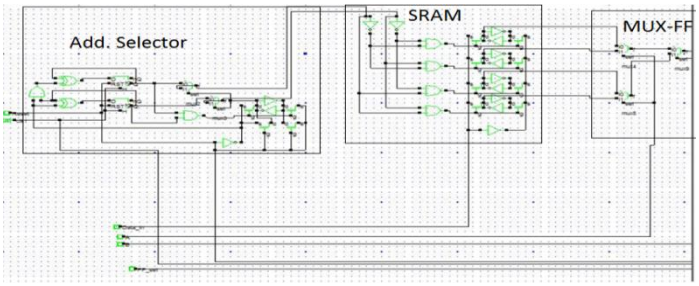


Fig. 3 Schematic diagram of a CLB

The following figure shows simulation for BLE-

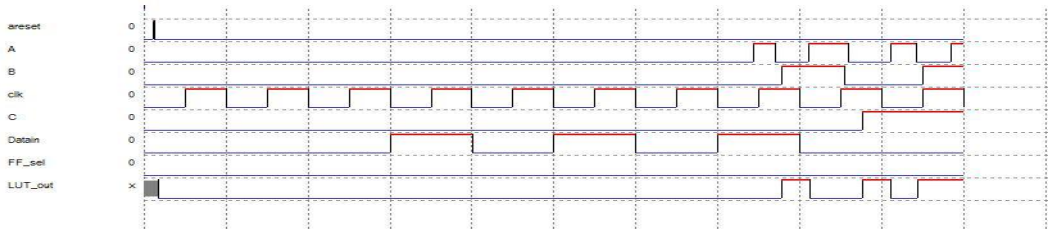


Fig. 4 Waveform of CLB schematic using DSCH

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

*(An ISO 3297: 2007 Certified Organization)*

**Vol. 5, Issue 1, January 2016**

There are several types of programmable architecture among those architectures Island- based architecture is one the most popular architecture. An Island Based FPGA routing network consists of horizontal and vertical routing tracks which are interconnected through routing switch. Logic blocks are connected to the routing network through connection boxes. These two are also used making connection between I/O block and internal components. In Fig.5 and Fig.6 schematic diagrams of a Programmable Routing Switch and Programmable Connection Block are shown.

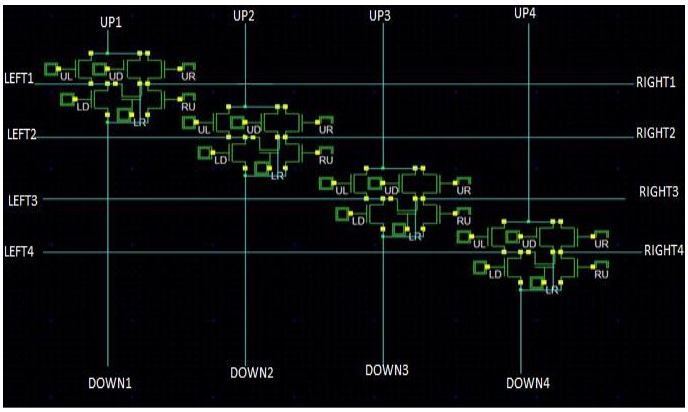


Fig. 5 Programmable Routing Switch

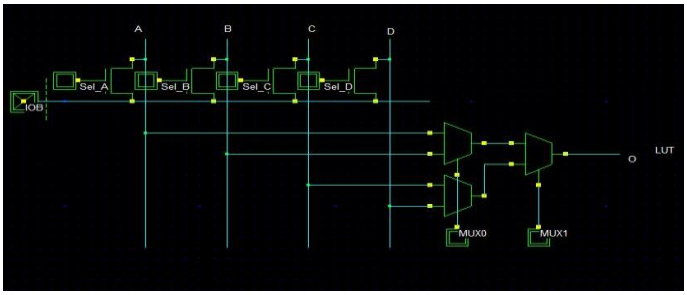


Fig. 6 Programmable Connection Block

A programmable routing switch consists of some small blocks. In table 3, Verilog-AMS code of a small block of a programmable routing switch has been shown.

Table 3:Verilog-AMS code of a small block of Routable Switch Block

| **Veriog-AMS code of a small block of Routable Switch Block.** |
|---|
| Module switch_block (UP, DOWN,LEFT,RIGHT,UD,UR,RD,LR,UL,LD);<br>input wire UD,UR,RD,LR,UL,LD;;<br>inout wire UP,DOWN,LEFT,RIGHT;<br>assign DOWN=(UD==1) ? UP : (UR==1) ? RIGHT : (LD==1) ? LEFT : 1'bz;<br>assign LEFT= (UL==1) ? UP : (LR==1) ? RIGHT : 1'bz;<br>assign RIGHT= UR==1) ? UP : 1'bz;<br>endmodule |

In table 4, Verilog-AMS code of a simple connection blocks has been demonstrated.

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

*(An ISO 3297: 2007 Certified Organization)*

**Vol. 5, Issue 1, January 2016**

Table 4: Verilog-AMS code of a connection block

| **Verilog-AMS code of a connection block** |
|---|
| module<br>connection_block(A,B,C,D,Sel_A,Sel_B,Sel_C,Sel_D,<br>Mux0,Mux1,I,O );<br>input wire Sel_A, Sel_B,Sel_C, Sel_D,I,Mux0,Mux1;<br>inout wire A,B,C,D;<br>output wire O;<br>assign A=(Sel_A)?I :1'bz;<br>assign B=(Sel_B)?I :1'bz;<br>assign C=(Sel_C)?I :1'bz;<br>assign D=(Sel_D)?I :1'bz;<br>assign O= ((~Mux0) & (~Mux1) ) ? A:<br>((~Mux0) & Mux1 ) ? B:<br>( Mux0 & (~Mux1) ) ? C:<br>( Mux0 & Mux1 ) ? D:1'bz;<br>endmodule |

The simulation of switch block and connection block have been shown in the following figures.

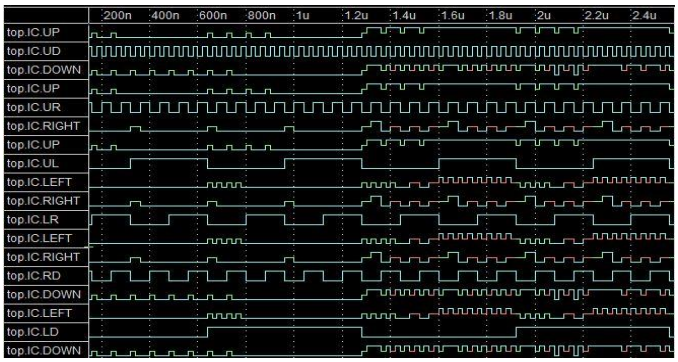


Fig. 7 Waveform of a small block of Routable Switch Block

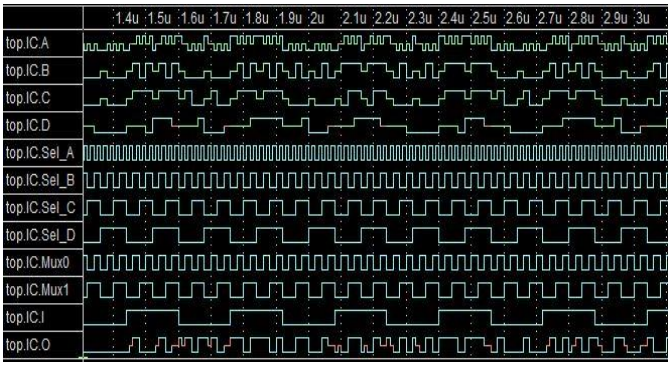


Fig. 8 Waveform of a Connection Block using Verilog-AMS

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

*(An ISO 3297: 2007 Certified Organization)*

**Vol. 5, Issue 1, January 2016**

## VI. SIMPLE FPGA MODEL

Two ADC were used to take continuous signal from analog source. Both of these signals were passed through two LUT's for adding those signals. Finally, outputs of those two LUT's were used as inputs of a DAC. For synchronizing clocks of the LUT's, a DLL was employed. As both of LUT's performed adding operation, data inputs of those LUT's were 0 1 1 0(output of a half adder). Model of the simple FPGA (excluding DLL) is shown below-

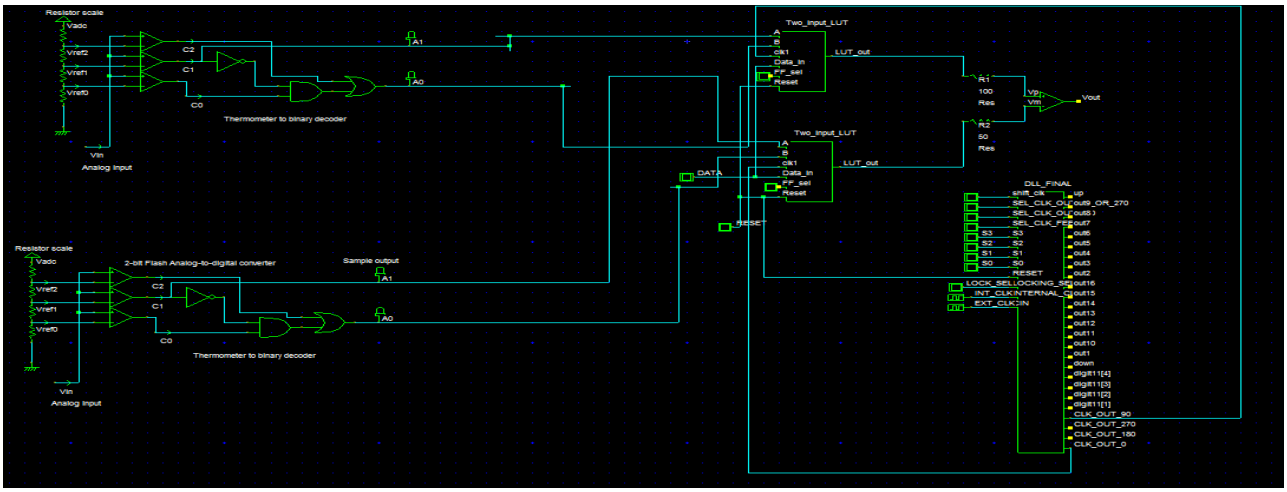


Fig. 9 Model of a simple FPGA

Verilog-AMS code of a simple FPGA using the modules of 'ADC', 'DAC' and 'LUT' is given below-

Table 5: Verilog-AMS code of a simple FPGA

```
`timescale 1ns / 1ps
`include "disciplines.vams"
module FPGA(fpgaout,X,Y,Z,reset,clk,data,ff_sel);
input X,Y,Z;
output  fpgaout;
input logic  reset,clk,data,ff_sel;
electrical X,Y,Z,fpgaout;
logic  [0:2]A;
logic  [0:2]B;
logic  [0:2]C;
logic  [0:2]D;
adc  adc1(.out(A), .in(X), .clk(clk)); // 'adc' is the module of Analog to digital converter.
adc  adc2(.out(B), .in(Y), .clk(clk));
adc  adc3(.out(B), .in(Z), .clk(clk));
lut LUT1(.clb_out(D[0]),.ff_sel(ff_sel),.areset(reset),.clk(clk),.in_put(A),.data_in(data));
// LUT is the module of a Look Up table.
lut LUT2(.clb_out(D[1]),.ff_sel(ff_sel),.areset(reset),.clk(clk),.in_put(A),.data_in(data));
lut LUT3(.clb_out(D[2]),.ff_sel(ff_sel),.areset(reset),.clk(clk),.in_put(A),.data_in(data));
dac dac1(.out(fpgaout), .in(D), .clk(clk)); // dac is the module of a digital to analog
converter.

Endmodule
```

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

*(An ISO 3297: 2007 Certified Organization)*

**Vol. 5, Issue 1, January 2016**

## VII.DISCUSSION AND CONCLUSION

As, DSCH does not support analog signal simulation, we were unable to evaluate the performance of our simple FPGA. Alternatively, we tried to simulate the whole FPGA in SMASH. But, we also could not do that because SMASH cannot simulate more than 15 signals at a time. Thus, future work of our thesis would be to accomplish the above mentioned task as effectively as possible with analog mixed signal simulator that does not have limitations. We successfully modeled and simulated ADC, DAC and PLL in Verilog-AMS. We also successfully modeled and simulated programmable BLE, DLL and Programmable Interconnect. Physical layouts were also generated for those components. We want to integrate everything for simulation and layout. This is a task we leave for future student researchers.

## REFERENCES

[1] Kenneth S. Kundert and Olaf Zinke, The Designer's Guide To Verilog-AMS,1st edition, New York, Boston ,Dordrecht, London, Moscow, Kluer Publishers,2004

[2] Stephen D. Brown,Robert J. Francis,Jonathan Rose,Zvonko G. Vranesic "Field-Programmable Gate Arrays." Springer Science+Business Media, LLC, chapter 3, 4, 5, edition 1992, ISBN 978-1-4615-3572-0.

[3] Russell G. Tessier, "Fast Place and Route Approaches for FPGAs". S.M., Massachusetts Institute of Technology,Department of Electrical Engineering and Computer Science, 1992 .

[4] Thomas L. Floyd, Digital Fundamentals, 11th Ed, ISBN-13: 978-0132737968

[5] Chia-Lin Chang, "A Wide-Range All-Digital Delay-Locked Loop in 65nm CMOS technology", M. Eng. thesis, National Chung Cheng University, Taiwan, 26-29 April 2010

[6] Nivedita,Ushaben Keshwala, "System Level Design of Timing and Frequency control circuit ",IOSR Journal of VLSI and Signal Processing (IOSR-JVSP) Volume4, Issue 4,Version. IE (Jul-Aug 2014)