Low Power Analysis of DLX Processor Datapath using a Novel Clocking Scheme

R K Megalingam
Senior Lecturer, Department of ECE[1]
91476-2801280
rajeshm@am.amrita.edu

S Hassan, T Rao
Student, Department of ECE[1]
+91-9496329884
shekhil@ieee.org

A Mohan, V Perieye
Student, Department of ECE[1]
+91-9496360733
ashwinmohan@ieee.org

ABSTRACT

Low power VLSI circuit design is a core area for current research activities. Power reduction without compromising the performance is the vital concern for processor design. In this paper, we apply a new clocking scheme as in [1] that can be used to reduce the power consumption in a processor datapath. We have mainly focused on implementing the pipelined DLX processor datapath in HDL using two different clocking schemes as in [1] and analyzed the power consumption. We have adopted the method which uses dual edge triggered clock that can decrease the power consumption of a pipelined datapath considerably, without sacrificing the throughput of the CPU. Finally, we have given the experimental results which confirm the low power consumption of the DLX processor datapath.

Categories and Subject Descriptors

D.3.3 [**Register Transfer Level Implementation**]: Design – *Data-path Design.*

General Terms

Performance, Design

Keywords

Pipelining, Datapath, Power consumption, Dual Edge Triggered Clock, Design.

1. INTRODUCTION

CPU datapath, which forms the basic design of a microprocessor, has evolved in its designing criteria and this has inspired many research works in this field. Processor datapath determines the very basic flow of the instructions, and efficient throughput is achieved by processing these instructions through various stages in the design. Datapath designing faces many constraints, which includes the processing speed, the power consumed by the datapath, effectiveness and efficiency of the design and the overall throughput of the design. Nowadays, CPU datapath is designed considering these constraints as challenges and research work is done to elevate the present technology thereby increasing the overall performance of the processor. CPU datapath implements pipelining for improved performance and this enhances the overall performance of the microprocessor.

Pipelining is a technique by which multiple instructions can be overlapped in execution. In pipelining each instruction is completed in many stages. The overlap among instructions is known as instruction-level parallelism (ILP). Many hazards set back the efficient working of the pipeline. The major three types of hazards in pipelining are Structural hazards; Data hazards and Control hazards [2].Hazards in a pipeline make it necessary to stall the instructions in the pipeline. Various efficient methods have helped in reducing the stalls.

Datapath is the flow and transformation of data in processor. Generally it adverts to the internal bus of the processor with a collection of registers, control unit and many functional units. In this paper we have adopted DLX processor datapath [2] for low power analysis using a novel clocking scheme [1]. DLX processor is a good architectural model due to its simplicity and clarity. It is a simplified MIPS (Microprocessor without interlocked pipeline stages) 32-bit Load/Store architecture.

Power saving in a microprocessor is a significant area of present day research and lots of technologies has emerged to scale down power consumption. Clock is the major source of power dissipation in a microprocessor. In this paper we have converged on a new clocking strategy [1] to reduce power dissipation in a pipelined architecture.

2. POWER DISSIPATION

There are two types of power dissipation in the various blocks that are used in the datapath. They are static dissipation and dynamic dissipation.

2.1 Static Dissipation

Static dissipation is due to leakage current or other current drawn continuously from the power supply. The main cause of static dissipation is due to reverse bias leakage current between diffusion regions and substrate. The sub- threshold conduction also contributes to the static dissipation. Static power is consumed even when chip is quiescent.

The static power dissipation is given by the product of device leakage current and the supply voltage. It usually occurs in pseudo nMOS gates, where there is direct path between source and ground.

ICWET'10, February 26–27, 2010, Mumbai, Maharashtra, India.

[1] Amrita School of Engineering, Amrita Vishwa Vidyapeetham, Clappana P.O, Kollam, Kerala, India, PIN:690525.

2.2 Dynamic Dissipation

Dynamic power is due to switching transient current and due to charging and discharging of load capacitances [8]. In CMOS circuits a significant portion of power consumption is seen when the signal value toggles. During the transition of signal values, the power is drawn from the power supply. This transition power is referred to as Dynamic Power. If the number of transition increases in a CMOS circuit, then the amount of power consumed by the circuit will also increase.

In sequential circuits, it is often seen that the clock consumes more power as the toggle rate is high for the same. Current also charges and discharges the output capacitive load and this capacitive charging and discharging current is dominant in dynamic dissipation. Dynamic power P_d is given by

$$P_d=C_LV_{DD}^2f_p \qquad (1)$$

C_L-Load capacitance

f_p -Frequency of switching

This equation shows that the power consumption is directly proportional to the operating frequency. Hence, if we can reduce the operating frequency, it will reduce the power consumption by a significant amount. This will enhance the performance of the CPU to a greater extent.

3. DLX PROCESSOR DATAPATH

DLX processor implements RISC processor architecture [2]. This architecture consists of five stages namely Instruction Fetch(IF), Instruction Decode(ID), Execute(EX), Memory read or write(MEM) and Register write back(WB). Each instruction is divided into component stages. Every instruction (except conditional or unconditional branches) should pass through all the stages. We have implemented the DLX processor datapath that can also perform operand forwarding during data hazards and flush the pipeline during branch (taken) instructions.

Two types of memories - Instruction memory (for storing instructions) and Data memory (for storing data) are used in DLX processor datapath. The only memory instructions in RISC architecture are load and store. Thus memory access is very less in RISC pipeline when compared to conventional microprocessors [1]. Each stage will process an instruction in one clock cycle. At an average one instruction will be executed in each clock cycle.

The DLX processor datapath comprises of four pipeline registers - IF/ID, ID/EX, EX/MEM and MEM/WB. These registers are placed between two stages. Any data value required in the later stages must be propagated through the pipeline registers. The write back addresses, intermediate data values and control signals should pass through these registers [2].

4. RTL & SYNTHESIS

4.1 Instruction Format

The instruction is 32 bits wide. The implemented datapath can handle two types of instructions- R type instructions (ADD, SUB, MUL) and I-type instructions, ADDI- addition of a register data with an immediate value and Memory instructions(LOAD and STORE) and Conditional Branch. To make the pipelining less complex, fixed instruction formats are used. The instruction format for an R-Type instruction is shown in Figure 1. The first field specifies the opcode. The second and third field specifies the source register addresses. The fourth field specifies the data write back register address (Destination register address).

Opcode	Source register address 1	Source register address 2	Destination register address	
31 26	25 21	20 16	15 11	10 0

Figure 1. R-Type Instruction Format

The instruction format for I-type instruction is shown in Figure 2. The first field specifies the opcode. The second and third field specifies the register addresses. The second field is source for all I-type instructions. The third field in Figure 1 is destination register address for LOAD instruction but source register address for branch and immediate ADD, SUB instructions.

We have implemented the DLX processor datapath as two designs with different clocking schemes. In one design a Single Edge Triggered Clock (SETC) is used whereas in the other design we used Dual Edge Triggered Clock (DETC) as shown in Figure 5 [2].

4.2 Flush Logic

Flush logic has an important role when a branch is taken. Branch is performed based on the 'zero' register in EX/MEM pipeline register block of Figure 3. This register is a part of flush logic. When the register value is '1' the data path will be reset. During reset, the fetched value in i_mem block will be flushed out and simultaneously the instruction in the location corresponding to branch address will be fetched. This instruction will be written to imem_out in the next clock cycle. The flush logic will also reset the values in the ID/EX pipeline register block (The values read from the register and the control signals of Figure 3) [4].

4.3 Operand Forwarding

Operand forwarding is done to overcome data hazards. In our datapath, we have implemented operand forwarding logic in two stages- ALU and MEM. The destination address, RegWrite control signal and ALU output of the previous instruction from ID/MEM block and MEM/EX block is fed back to the operand forwarding block. Operand forwarding is done only when the previous instruction is an R-type instruction and any one of the source operand register addresses equals the destination register address of the previous two instructions. If these values are equal, then the ALU output value or dmem_out of the previous instruction is fed as operand to the ALU block.

Opcode	Register address 1	Register address 2	Offset value
31 26	25 21	20 16	15 0

Figure 2.I-Type Instruction Format

In the case of operand forwarding in ALU block, the forwarded ALU output value can be fed to any one of the two ALU operands depending on their source register address dependency (any of the operand source registers can be equal to the previous instruction destination register address). A multiplexer is used for selecting the operands, either the source register data or the previous ALU output or the dmem output. Thus operand forwarding helps to avoid data hazards in ALU block.

Operand forwarding is also done in d_mem block. It is done when a load instruction is immediately followed by a store instruction and the destination address of the LOAD instruction matches with the memory input data's register address. The destination address, RegWrite control signal and ALU output of previous instruction from ID/MEM block is fed back to the operand forwarding block.

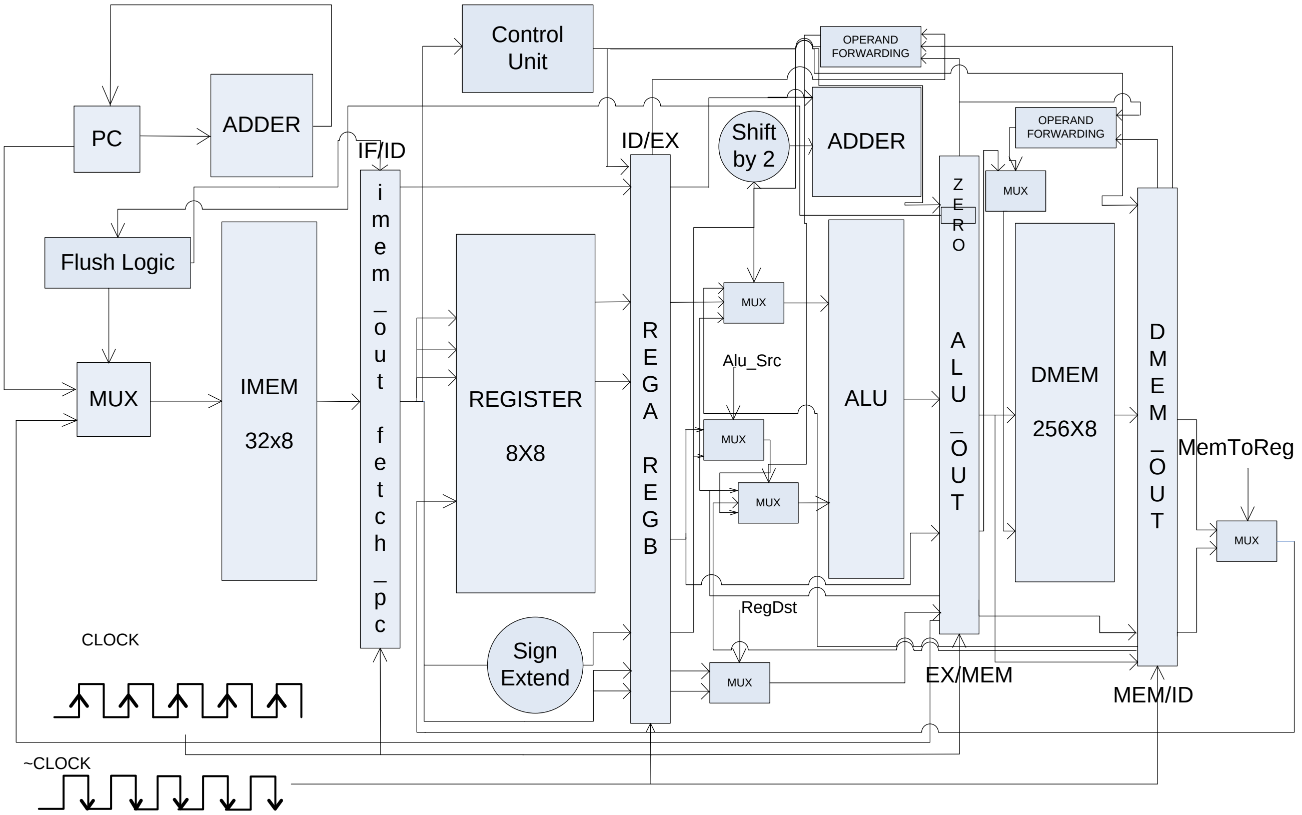

Figure 3.Implementation of DLX processor datapath using DETC

5. DUAL EDGED TRIGGERED CLOCK

Each segment in the pipelined architecture is followed by pipeline registers which stores the output as shown in Figure 5. This pipeline registers work on the clock, and this controls the overall throughput of the architecture. In the first design, the DLX processor datapath with SETC uses the Single Positive Edge Triggered Clock as shown in Figure 3. In the second design, the DLX processor datapath works with both the Positive Edge Triggered Clock and the Negative Edge Triggered Clock. Fig.4 shows the Positive Edge Triggered Clock and the Negative Edge Triggered Clock which is having a frequency that is half the frequency of the Positive Edge Triggered Clock that we used in the first design [6]. As an application of DETC, we made the IF/ID and the EX/MEM pipeline registers to work in Positive Edge and ID/EX and MEM/WB pipeline registers to work in Negative Edge of the clock. We can notice that the throughput of the segment is not affected as the proposed clocking scheme also gives us the identical triggering compared to the ordinary Positive Edge Triggered Clock. Since the frequency of the DETC is half compared to the ordinary clock frequency, power consumed by the architecture using DETC is reduced [1], [3]. As shown in the equation (1), power is proportional to the operating frequency. Since the frequency of the DETC is half compared to the ordinary clock, power consumed by this architecture is reduced. In the implemented architecture we have used both SETC and DETC. Both schemes have the same execution time but we achieved a reduction in power consumed by the architecture with DETC implementation [1]

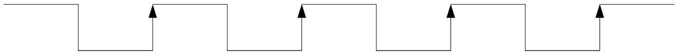

Figure 4.SETC

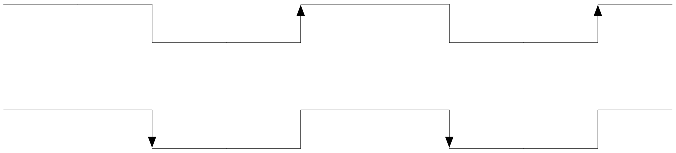

Figure 5. DETC where Triggering occurs in both Positive and Negative Edges

6. RESULTS & DISCUSSION

By implementing the DETC scheme, we have experimentally proved that the power consumption of the CPU can be reduced considerably. The DLX processor datapath was implemented and a comparison was carried out between the conventional datapath, which works with the SETC and proposed datapath, which works with the DETC. Both datapath were designed using Verilog^{TM} HDL. These designs were successfully simulated using ModelSim. Both the designs gave the identical timing measurements as expected and thus from this we could discern that the throughput of the design was not affected by changing the SETC scheme to DETC scheme. Both these designs were synthesized using Xilinx ISE 10.1. XPower Analyzer, which is an inbuilt power analyzing tool in Xilinx. This tool was used to analyze the power consumed by the DLX processor datapath with

SETC and the DETC. We compared the power consumed by the conventional design and the proposed design and plotted the graph for power consumed by the designs versus frequency of operation.

6.1 Timing Analysis

Simulation timing diagram for both the designs are shown in Figure 6 and Figure 7. The figures show the execution of four instructions. The instruction follows the standard MIPS instruction format. The instruction code snippet for which we have simulated the design using ModelSim is shown below:

ADD R2, R1, R3

SUB R5, R6, R3

ADD R10, R1, R5

MUL R9, R5, R8

SUB R7, R6, R10

From the figures it can be noted that both the designs give the same as expected.

Table 1. Power consumption for both the designs for different clock frequencies

Clock Frequency (MHz)		Power Consumed(W)	
SETC	DETC	DLX Datapath with SETC	DLX Datapath with DETC
100	50	0.219	0.214
200	100	0.242	0.224
400	200	0.268	0.245
600	300	0.300	0.266
800	400	0.332	0.286

6.2 Power Analysis

Figure 8 and 9 show the power consumption versus frequency for DLX processor datapath with SETC and DETC schemes. From this analysis it is clearly observed that the power consumption is reduced when we design the DLX processor datapath with DETC scheme. Figure 9 and Table 1 shows the power consumed by the DLX processor datapath with DETC for different clock frequencies. For different values of clock frequencies the power consumed by the DLX processor datapath is measured without toggling the input signals.

Figure 10 and Table 2 shows the power consumed by the DLX processor datapath using SETC with 100% input signal toggling. From Figure 11, it can be observed that the power consumed by the DLX processor datapath with DETC is nearly half that of the other design using SETC. For 100 MHz clock frequency, the power consumed by the DLX processor datapath with SETC is 0.464 W.

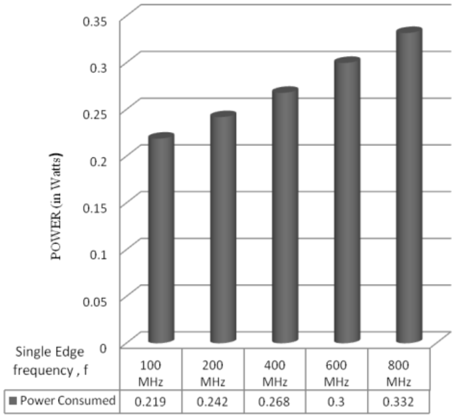

Figure 8.Graph showing the power consumed by the DLX processor datapath with SETC for different clock frequencies

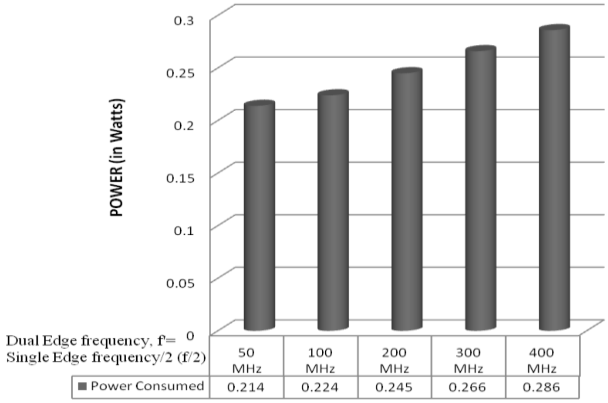

Figure 9. Graph showing the power consumed by the DLX processor datapath with DETC for different clock frequencies

Since the clock frequency is half in the case of DETC, the corresponding clock frequency is 50MHz and the power consumed by the DLX processor datapath is 0.339 W. In this case the power saving obtained is around 27%. In case of 600 MHz clock frequency for SETC, the power consumed by the datapath is 1.766 W. For DETC of 300 MHz clock frequency it is observed to

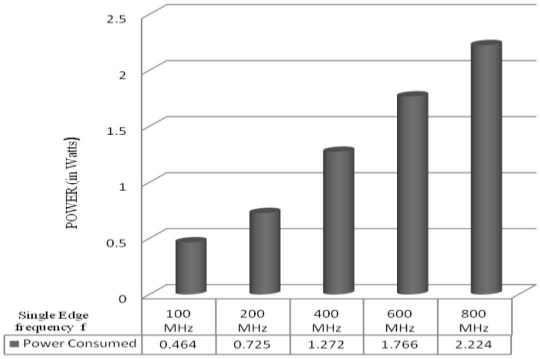

Figure 10.Graph showing the power consumed by the DLX processor datapath with SETC for 100% signal toggling

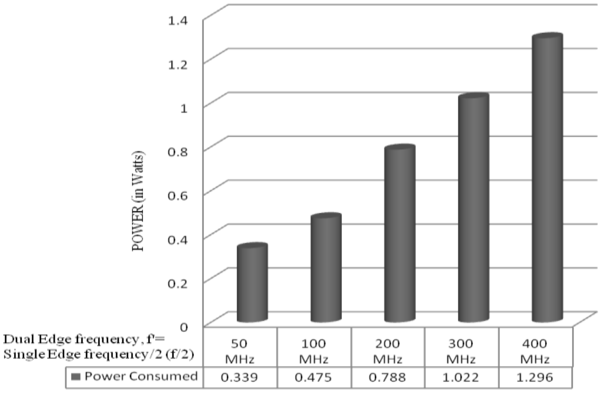

Figure 11.Graph showing the power consumed by the DLX processor datapath with DETC for 100% signal toggling

be 1.022 W. For this frequency, the power saving was observed to be 42.13%.From Table 2, it can be inferred that the power saving increases as the frequency of the clock increases.

We implemented the DLX processor datapath with SETC and DETC. We analyzed the power consumption for the different clock frequencies and 100% signal toggling for both the designs. It was observed that the power consumed by the DLX processor datapath with DETC was considerably reduced compared to the power consumed by the DLX processor datapath with SETC. It was further observed that the power saving for DLX processor datapath with DETC was up to a maximum of 43%.

Table 2.Power Consumption for both the designs for 100% signal toggling

Clock Frequency (MHz)		Power Consumed(W)	
SETC	DETC	DLX Datapath with SETC	DLX Datapath with DETC
100	50	0.464	0.339
200	100	0.725	0.475
400	200	1.272	0.788
600	300	1.766	1.022
800	400	2.224	1.296

7. CONCLUSION

We used two clocking schemes for DLX pipelined architecture and analyzed the power consumption for each scheme. The first scheme was a Single Positive Edge Triggered Clock and the second was using a Dual Edge Triggered Positive-Negative Clock. Both datapath designs were simulated in ModelSim and we observed that both schemes provided the same output timings as expected. These two designs were synthesized and power analysis was done in Xilinx ISE 10.1. Power consumption was observed for varying clock frequencies and signals toggling in both the designs and frequency-power graphs were plotted. It was observed that the power consumed in the DETC was less than the SETC. Theoretically power reduction of 50% was expected. Experimentally we were able to observe a power reduction of maximum 43%. We also noticed that the power reduction increases as the clock frequency increases. Thus the implementation of the DETC in the DLX architecture will decrease the power consumption without sacrificing throughput

8. ACKNOWLEDGMENTS

We gratefully acknowledge the Almighty GOD who gave us strength and health to successfully complete this venture. The authors wish to thank Amrita Vishwa Vidyapeetham, in particular the Digital library, for access to their research facilities.

9. REFERENCES

[1] Rajesh Kannan Megalingam, Shekhil Hassan, Vivek P, Ashwin Mohan, Tanmay Rao M, “ Power Consumption Reduction in CPU Datapath using Novel Clocking Scheme”, IEEE ISBN No: 978-1-4244-4520-2/09, ICCSIT 2009

[2] John.L.Hennessy, David.A.Patterson, “Computer Architecture – A quantitative approach” 4th edition.

[3] Rajesh Kannan Megalingam, Nived Krishnan, Arjun Ashok V, Arun Kumar M, “High Performance, Uncompromised performance Cache Design Using Dual Edge Triggered Clock”, IEEE ISBN No: 978-1-4244-4520-2/09, ICCSIT 2009

[4] Rajesh Kannan Megalingam, Venkat Krishnan B, Mithun M, Rahul Srikumar, Vineeth Sarma V, “Gating and Serialising the Datapath of CPU for Low Power Consumption”, Proceeding of 2009 International Conference on Parallel Processing (ICPP ’09)

[5] Ziad Youssfi, Michael Shanblatt, “A New Technique to Exploit Instruction-Level Parallelism for Reducing Microprocessor Power Consumption”, 0-7803-9592-1, 2006 IEEE International Conference on Electro/information Technology

[6] Vishwanadh Tirumalashetty and Hamid Mahmoodi, “Clock Gating and Negative Edge Triggering for Energy Recovery Clock”, 1-4244-0921-7/07.

[7] Norman Yeung, Barbara Zivkov, Gulbin Ezer Unified Datapath: An Innovative Approach to the Design of a Low-

Cost,Low-Power, High Performance Microprocessor , 1063-6390/94

[8] John P Uyemura, “Introduction to VLSI Circuits and Systems”.

[9] Al-Mohandes, I.H.; Rashed, A.M.; Ragaie, H.F.; Elsaid, M.K.; ,“Synthesis and physical design of DLX RISC processor”, 977-5031-62-1, Proceedings of the Sixteenth National Radio Science Conference, 1999. NRSC '99

[10] Fagin, B.; Chintrakulchai, P.;, Prototyping the DLX microprocessor, 0-8186-3520-7, International Workshop on Rapid System Prototyping, 1992.

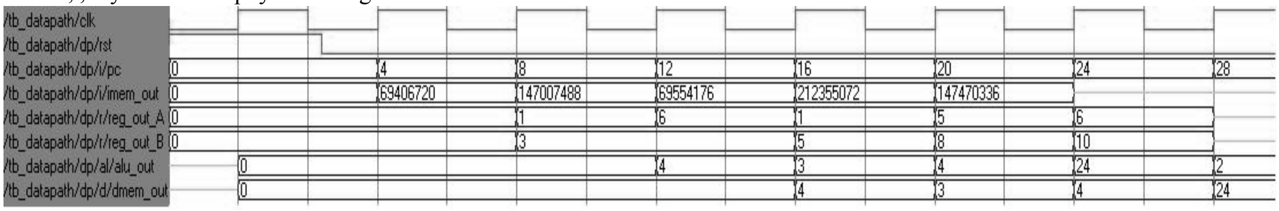
/tb_datapath/clk
/tb_datapath/dp/rst
/tb_datapath/dp/i/pc 0 4 8 12 16 20 24 28
/tb_datapath/dp/i/imem_out 0 69406720 147007488 69554176 212355072 147470336
/tb_datapath/dp/r/reg_out_A 0 1 6 1 5 6
/tb_datapath/dp/r/reg_out_B 0 3 5 8 10
/tb_datapath/dp/al/alu_out 0 4 3 4 24 2
/tb_datapath/dp/d/dmem_out 0 4 3 4 24

Figure 6.Simulation Timing Diagram of Design using SETC

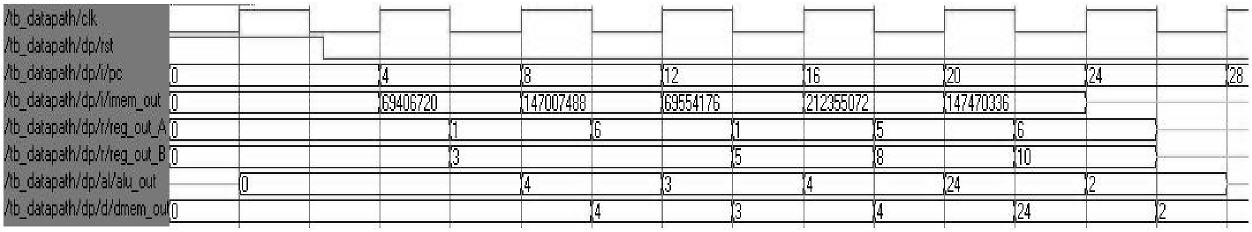
/tb_datapath/clk
/tb_datapath/dp/rst
/tb_datapath/dp/i/pc 0 4 8 12 16 20 24 28
/tb_datapath/dp/i/imem_out 0 69406720 147007488 69554176 212355072 147470336
/tb_datapath/dp/r/reg_out_A 0 1 6 1 5 6
/tb_datapath/dp/r/reg_out_B 0 3 5 8 10
/tb_datapath/dp/al/alu_out 0 4 3 4 24 2
/tb_datapath/dp/d/dmem_out 0 4 3 4 24 2

Figure 7.Simulation Timing Diagram of Design using SETC