# RISC Architecture based DLX Processor for Fast Convolution and Correlation

Sumalatha S
Department of ECE
Acharya Institute of technology
Bangalore-90, Karnataka.

Rajeswari
Department of ECE
Acharya Institute of technology
Bangalore-90, Karnataka.

Jayalaxmi.H
Department of ECE
Acharya Institute of technology
Bangalore-90, Karnataka.

## ABSTRACT
The need for convolution and correlation arises most frequently in all signal processing applications, which demands for optimization in processing speed. In this paper an efficient architecture for the implementation of Fast Correlation and Convolution using FPGAs through DLX 32-bit RISC processor is proposed. The proposed methodology mainly focuses on the design of 32-bit pipelined RISC processor based on the DLX architecture to perform fast convolution and correlation operations. The experimental results demonstrate that Field Programmable Gate Arrays FPGAs provide flexibility in architecture design and optimizes the processing speed in few nano seconds**.**

## General Terms
Convolution, Correlation, Architecture, Processors,

## Keywords
FPGA, DSP, Modelsim, correlation, convolution, Spartan3, Xilinx.

## 1. INTRODUCTION
DSPs are commonly used to speed up the computation of many signal and image processing algorithms. Though easy to program, DSPs have a fixed architecture that limits the kind of operations that can be performed[1]. DSPs therefore do not provide a system-on-a-chip solution which is a drawback when space and mobility are a concern. ASICs on the other hand provide the greatest amount of flexibility in designing the architecture but suffer from a long and tedious design process. Furthermore, the high cost of designing and fabricating an ASIC can make it prohibitive to use [2]. Reconfigurable devices such as FPGAs provide a middle ground. The design process is shorter and cheaper than for an ASIC and they provide much greater flexibility than DSPs making it possible to develop a variety of algorithms like convolution and correlation from start to end on the FPGA. Another important advantage of FPGAs is that they are reconfigurable and also a process can be completed in a span of nano-seconds so that the same chip can be used for different purposes.

### 1.1 DLX Processor
The DLX Microprocessor is a RISC Processor designed by John L. Hennessy and David A. Patterson, the principal designers of the MIPS and the Berkeley RISC designs (respectively), the two benchmark examples of RISC design [3,4]. The DLX is essentially simplified MIPS with simple 32-bit load/store architecture. The DLX design is widely used in university-level computer architecture courses[5] to help students to get knowledge about the RISC processor in terms of instruction encoding & decoding, pipelined operations, functions of each component within the processor, types of instruction set and operations etc. It is based on classic Harvard architecture of Separate instruction and data memories to allow simultaneous instruction fetching and data memory transactions. It supports 3 types of instruction formats (I, R & J) & uses immediate & displacement addressing modes. In non pipelined execution, each instruction takes 5-clock cycles.

### 1.2 Convolution And Correlation
Convolution is the most important and fundamental concept in signal processing and analysis. By using convolution, we can construct the output of system for any arbitrary input signal, if impulse response of that system is known. The convolution can be defined for functions on groups other than Euclidean space. In particular, the circular convolution can be defined for periodic functions (that is, functions on the circle), and the discrete convolution can be defined for functions on the set of integers.. The mathematical definition of convolution in discrete time domain [6] is given in eqn.(1)

$$y(n) = x(n) * h(n)$$
$$= \sum_{n=-\infty}^{\infty} x(n).y(n-k) \quad (1)$$

On the other hand correlation [6] is a measure of similarity of two waveforms as a function of a time-lag applied to one of them. This is also known as a sliding dot product or inner-product. It also has applications in pattern recognition. single particle analysis, electron tomography averaging, cryptanalysis, and neurophysiology. For discrete functions, the cross-correlation is defined as shown in eqn. (2)

$$Rxy = \sum_{n=-\infty}^{\infty} x(n).y(n-k) \quad (2)$$

k = 0,±1,±2,…..

where x(n) & y(n) are two real signal sequences, each of which has finite energy. The index k is the (time) shift parameter and the subscripts xy on the cross correlation sequence $R_{xy}(k)$ indicate the sequence being correlated. The order of the subscript with x preceding y, indicates the direction in which one sequence is shifted, relative to the other. i.e. x(n) is un-shifted and y(n) is shifted by k units in time, to the right for k positive and to the left for k negative.

## 2. IMPLEMENTATION

The proposed method mainly concentrates on the design of 32-bit pipelined RISC processor based on the DLX architecture to perform fast convolution and correlation operations, which is most required for almost all signal processing applications. The processor is designed to handle both integer and IEEE-754 floating point data formats making it effective for any application requiring floating point operations, namely graphics and scientific. Additionally, many modern multimedia applications requires the use of concurrent data calculations, which it can handle smoothly because of three pipeline stages. The processor design follows classic Harvard architecture of separate instruction and data memories to allow simultaneous instruction fetching and data memory transactions. The processor is designed by a powerful and flexible hardware descriptive language VHDL[7]. VHDL model of the DLX microprocessor consists of various functional units which were created as components. These components were coded as separate modules which were composed, simulated and then instantiated in the main entity. This microprocessor contains the following functional modules as shown in fig-1.

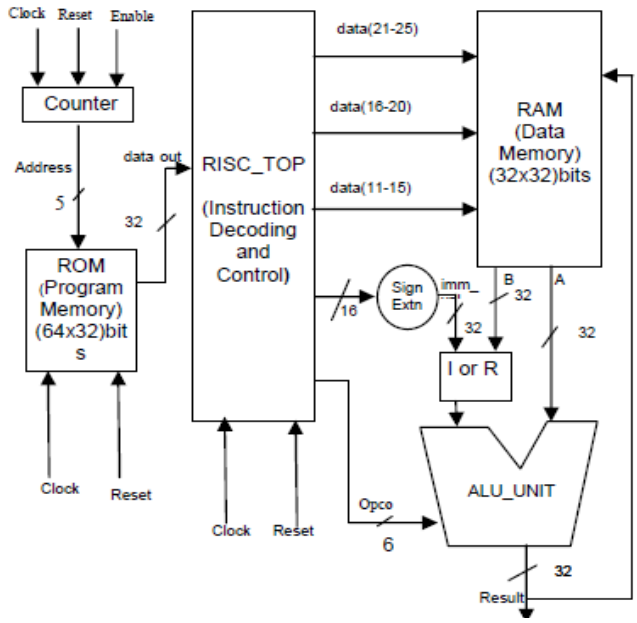


**Fig 1: VHDL model of DLX processor**

### 2.1 Implementation of DLX Processor

*Assembler***:** This module converts the assembly code written in text editor into equivalent binary instructions by following dlx instruction formats.

*Counter Unit***:** Functions as instruction fetching module. Once it fetches one instruction from ROM the program counter is automatically incremented to fetch further instructions from the instruction memory.

*Instruction decoding and control unit*: Instruction fetched is decoded based on I-type or R-type format. The controller then fetches operands from the RAM(data memory) if data is reside in register specified in register fields of the instruction otherwise 16- bit immediate data is sign extended to make 32-bit data and issues command to ALU as per the opcode decoded by decoder.

*ROM(Program Memory)***:** General purpose ROM memory module for instruction storage, in which instructions are downloaded from the in binary form by following the DLX instruction format.

*RAM (Data Memory)***:** Data memory is also called register file that contains all the 32-bit general purpose registers and 32-bit floating point registers of the microprocessor. The register file is not containing any reserved registers, such as the PC, the status register, or other special registers so that all register can be used for all operation.

*ALU:* Arithmetic and logic unit, performs particular operation on the given operators and operands. Various command operations that can be performed are Addition, Subtraction, AND, OR, XOR, logical comparisons, shift (left / right), divide, multiply Increment, decrement etc.

The complete methodology of DLX processing is described in the following flowchart shown in fig-2.

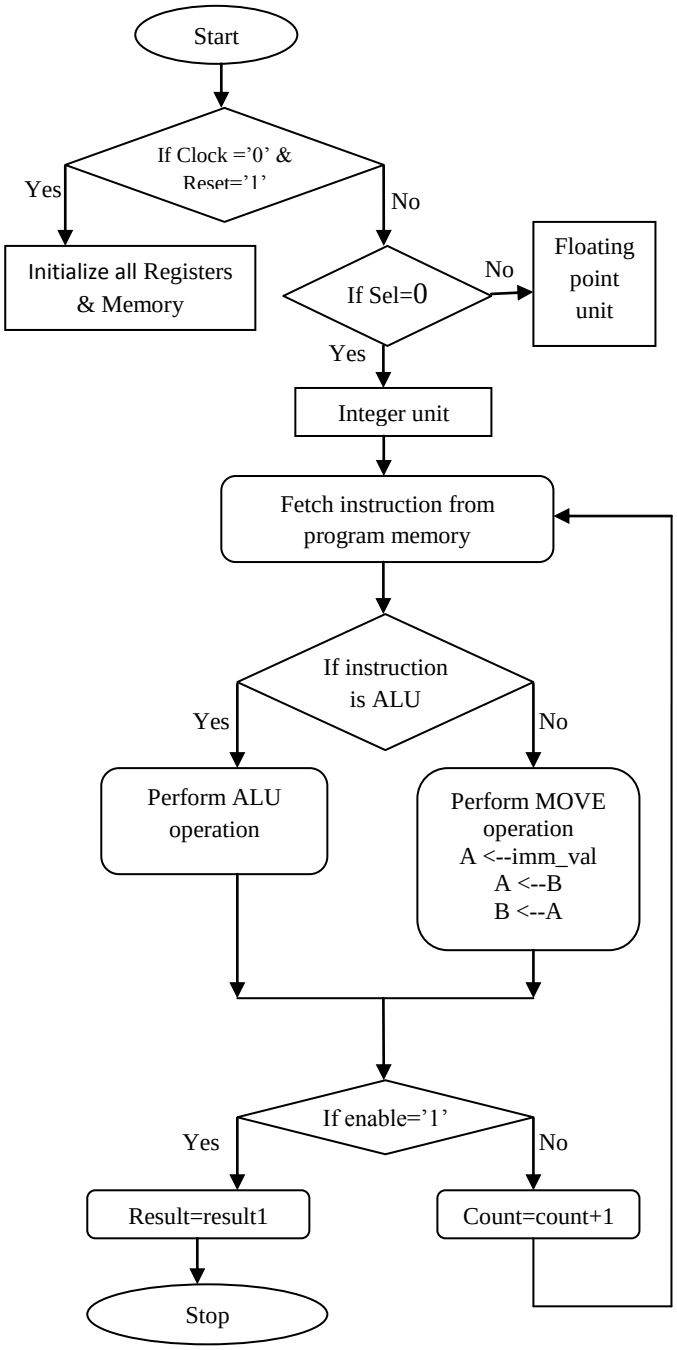


**Fig 2: Processing model of DLX processor**

## 2.2 Implementation of convolution

The response y(n) of LTI system as a function of the input signal x(n) and the unit sample response h(n) is called a convolution and is given by

$$Y(n) = \sum_{k=-\infty}^{\infty} x(k)h(n-k) \qquad (3)$$

The index in the summation k indicates both the input signal x(k) and impulse response h(n-k) are functions of k.The process of computing convolution between x(k) and h(n-k) involves following four steps.

Folding: fold h(k) about k=0 to obtain h(-k)

Shifting: shift h(-k) by n to the right if n is positive or to the left if n is negative to obtain h(n-k)

Multiplication: multiply x(k) by h(n-k) to obtain the product sequence.

Summation: sum all the values of the product sequence to obtain the value of the output y(n).

## 2.3 Implementation of Cross Correlation

The cross correlation is similar in nature to the convolution of two functions, whereas convolution involves reversing a signal and shifting it and multiplying by another signal, correlation involves shifting and multiplying. Thus the convolution of x(n) with y(-n) yields cross correlation $R_{xy}$ (k).

## 3. SIMULATION AND SYNTHSES RESULTS

Simulations were conducted to verify the feasibility of the proposed technique. The VHDL code for the DLX microprocessor was programmed in Xilinx-ISE 9.1i and was simulated successfully in Modelsim XE III 6.2C tool [8]. Individual blocks of the DLX Microprocessor Architecture was created as different modules and simulated separately and verified. for accurate results. Later these modules were instantiated in the Main Architecture. The main Architecture was then used to implement both convolution and correlation algorithms. The algorithms were written by following op-codes as well as instruction formats of DLX processor then it is processed by designed architecture and simulated successfully using Modelsim XE III 6.2C tool. Both algorithms were verified for various integer samples. The fig.3 and fig.4 shows simulation results for both convolution and correlation.

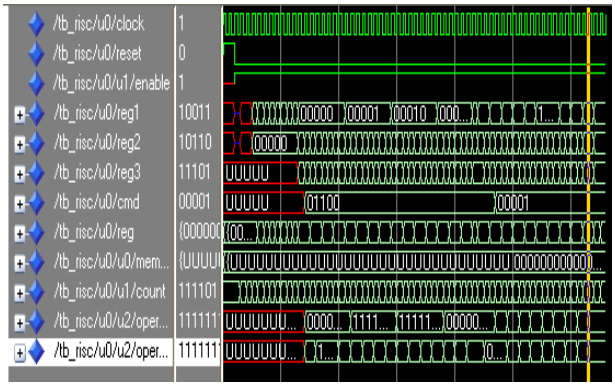


**Fig 3: Simulation result of convolution**

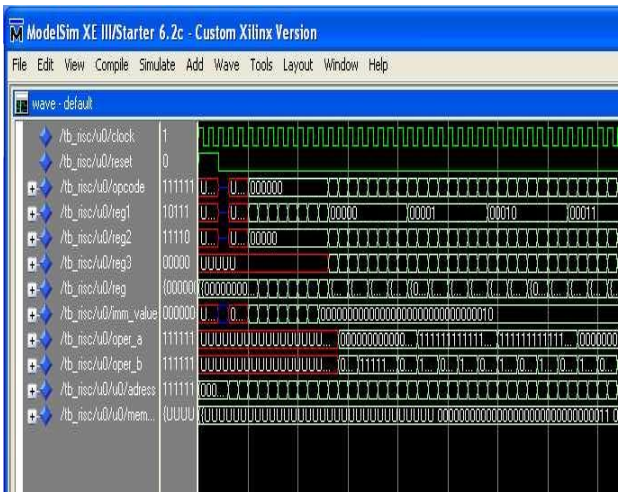


**Fig 4: simulation results of correlation**

The simulation results shows that the designed DLX architecture works as a pipelined processor with each instruction in pipelined execution takes 3-clock cycles. The processing speed or both convolution and correlation was found to be 13 nanoseconds. Finally the proposed design is implemented on Spartan-3 XC3s400 FPGA board [9] and verified the design by running both convolution and correlation programs for various integer samples, with verifying results for each case.

The RTL model of synthesized code is shown in fig-5 and also the number of devices used in the Spartan3 FPGA board is listed in Table-1.

**Fig 5: RTL model of the proposed method**

**Table-1: List of components used on XC3S400 Board**

| Devices | Available | Used | Percentage of device utilized |
|---|---|---|---|
| Flip Flops | 7168 | 100 | 1% |
| Slices | 3584 | 961 | 26% |
| 4 input LUTs | 7168 | 1759 | 24% |
| Bonded IOBs | 97 | 9 | 9% |
| BRAMs | 16 | 1 | 6% |
| MULT18X18s | 16 | 3 | 18% |
| GCLKs | 8 | 1 | 12% |

## 4. CONCLUSION

The primary objective of coding DLX processor using VHDL language was successfully completed. The designed architecture is examined by running both integer and floating point instruction. Finally the correlation and convolution algorithms were written and results are obtained by simulation as well as by realizing the code on Spartan-3 FPGA board. On future the performance of the proposed design is to be embedded for signal processing applications involving convolution and correlation, optimizing its speed of operation.

## 5. REFERENCES

[1] J mirald.'The future of high performance COTS signal processing hybrid FPGA/DSP architecture, the optimal solution in DSPFPGA.com resorce guide-2006

[2] Z.Guo.W Najjar F Vahid and K.Vissers. " A antitative analysis of the speedup factors FPGAs over processors, in ACM/SIGDA International Symposium on Field Programmable Gate arraya_FPGA Monterey C.A22-24 Febraury 2004 vol 12 pp 162-170

[3] Patterson D Hennessy,J."Computer Architecture A Quantitative approach" Morgan Kaufmann Publishers 1996, in Second edition.

[4] Patterson D Hennessy,J. "Computer Architecture A Quantitative approach" Morgan Kaufmann Publishers 1996, in Third edition.

[5] Ellard et al. "On the design of a new CPU architecture for pedagogical purposes", Proceedings of the 2002 workshop on Computer architecture education: Held in conjunction with the 29th International Symposium on Computer Architecture, May 2002

[6] John G Prokis, Dimitris G Monolakis "Digital Signal Processing-priniples,algorithms and applications" Prentice Hall of India Pvt Ltd ©2007 4th edition.

[7] P. J. Ashenden The Student's Guide to VHDL. San Francisco, CA:Morgan Kaufmann, 1998..

[8] http://www.eda.ncsu.edu/wiki/Tutorial:Modelsim_Tutorial

[9] www.xilinx.com/support/documentation/userguides/ug331.pdf