

Architecture of High performance Reconfigurable DSP processor

Manisha Ghosh, Saurav Mandal

Abstract— This paper proposed a reconfigurable DSP architecture and algorithms, such as Discrete Wavelet Transform (DWT) and Fast Fourier Transform where basic building blocks are high performance adder, subtractors, multipliers etc. The reconfigurable architecture reduces area and become cost-effective. In the proposed DWT architecture the input data are separated as even and odd numbers of data as well as both data are input parallelly. This causes faster DWT operation then conventional architecture. In conventional architecture N-point DWT is computed in N cycles where as in proposed architecture N-sample sequence is computed in only N/2 cycles. Therefore this architecture is at least twice as fast as the conventional architecture. This paper proposed a parallel pipeline based reconfigurable Fast Fourier Transform architecture which increases the speed of computation. Finally, we had verified the mat lab simulation result of the proposed FFT architecture by using mentor graphics tool. The proposed three architectures had been synthesized using XILINX ISE 9.1i version and the results of the same had also presented..

Index Terms— Discrete Wavelet Transform, Matrix Multiplication, Fast Fourier Transform, Modulo Unit, Switch Matrix, Random Access Memory,

1 INTRODUCTION

Signal processing function like Discrete Wavelet Transform [1][2][3][4][7], Fast Fourier Transform [5][6][8] are computationally intensive. If these functions are directly mapped into the hardware, it can be shown that the basic building blocks required for these function are same and most of these functions partial parallelism [9] or temporal parallelism [10] or both. Utilizing parallelism Von-Neumann or Harvard architecture oriented processor has become increasingly inefficient [11]. The advantage of executing computing intensive functions at hardware speed resulted in the emergence of Application Specific Integrated circuits (ASICs). Even though ASICs offer highest possible performance at lowest silicon cost, they suffer for inflexibility [11]. Besides, if a particular application needs a large number of functions to be executed in real-time then a large number of ASIC chips will be required, which is not cost effective. Field Programmable Gate Arrays (FPGA) is a high performance programmable hardware and this has emerged with an additional feature - dynamically reconfigurable at run-time, which makes them attractive in many signal processing applications. These deficiencies motivated the design a new reconfigurable DSP architecture with the following features:-

- 1) Major building blocks which are common to most of DSP functions.
- 2) A particular DSP function is configured by dynamically interconnecting these hardware blocks.

3) DWT architecture can compute N-sample sequence in N/2 cycle and hence conceive a parallel base architecture which can operate twice as faster.

2. REVIEW OF DISCRETE WAVELET TRANSFORM

A basic implementation of a 1-D DWT [12][13][14] has been done by using the Daubachies wavelet coefficients. Two different output bands are produced by applying two FIR filters on data input samples. A low pass filter produced low frequency data by using $h(x)$ coefficient and high pass filter produced high frequency data by using high pass filter coefficient $g(x)$. The conventional DWT [12][13][14] consisted of N- tap low pass filter and N-tap high pass filter. Input data are $a(0) a(1) a(2) a(3) \dots \dots a(N)$.

During one cycle filtering is operated and the filtered values are stored for next octave. Since one input data are filtered during one cycle, the computation period is N.

3 DETAIL OF THE ARCHITECTURE.

3.1 Details of the architecture

In this architecture basic building blocks are Modulo Unit (MU), which is shown in Figure 1. The MU consists of Functional blocks, Register Files, Buffers and Configurable Networks (CN). Functional blocks include multipliers (M0-M2), adder subtractors (A/S). Connection between the basic building blocks of Modulo unit is established by the configurable networks. IDR and ODR are the two array registers. The architecture allows parallel loading of data in register files IDR and ODR. Another memory data coefficient memory (DCM) is used to load the constant and the constant again used in operation. Depending on the mode signal of the CNs, the functional blocks can work in different modes (FFT, DWT).

- Manisha Ghosh , Assistant Professor, NSHM Knowledge Campus Durgapur, India, PH-07076201097. E-mail: manishaghosh.uit@gmail.com
- Saurav Mandal, Assistant Professor, DIATM, India, PH-09804770559. E-mail: saurav.technology@yahoo.com

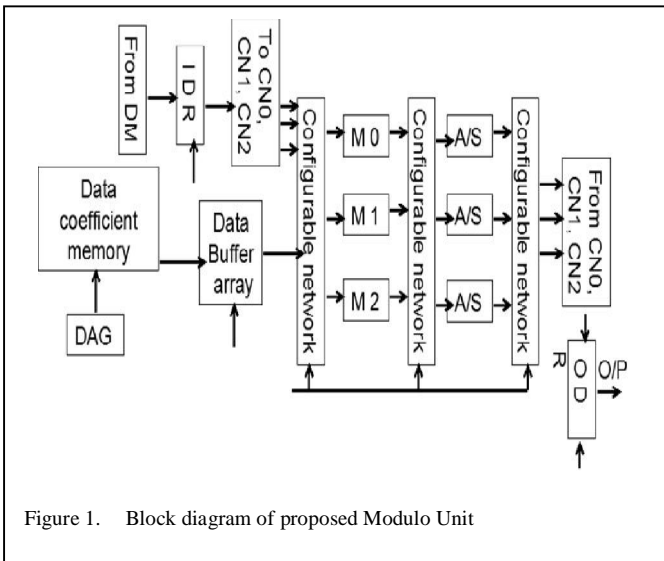


Figure 1. Block diagram of proposed Modulo Unit

Basic structure of CNs shown in figure 2. It can operate upto n modes. In this architecture n=2 bit mode control signal is required. Operation of configurable network for n=2 mode shown in Table 1. Beside data it is also route the control signals. Control signal is required to enable functional blocks like adder, subtractor, multipliers. A special connection is required for connecting IDR to CN0, CN1, CN2. This link will be used basically faster operation where concurrent addition, multiplication and subtraction is required. Depending upon the mode of signals CNs send the data to adder, multiplier blocks and output is available from ODM.

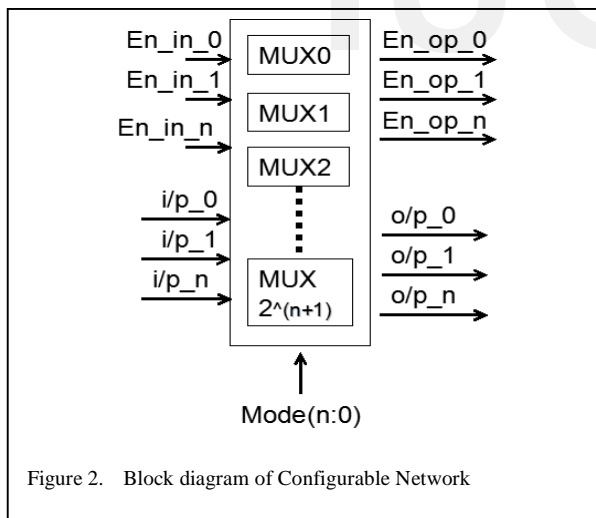


Figure 2. Block diagram of Configurable Network

the instruction from outside of the processor and store them in queue. CU fetch the instruction one by one from the ID. CU has

Mode (n=1:0)	Operation
00	Multiplication
01	Addition
10	Subtraction
11	Butterfly

generated control signals to program address generator, which configure the configurable memory. The CU also gives instruction to data address generator (DAG). DAG generate the location of the data in DR. DR has two segment one is DR0 for storing the intermediate results and DR1 for storing the input from outside of the processor. MUXes select lines are controlled by the control unit (CU). PAG have generated two addresses one for reading control data and another for configuring the CM. In CM when execution is completed, it will set the control flag signal in 1. Otherwise CU shows busy signal with its previous instruction. Data decoder has decoded the data from CM and generates control signals which is reduce the width of control memory. The architecture is so flexible that number of concurrent micro operation perform to a single control word.

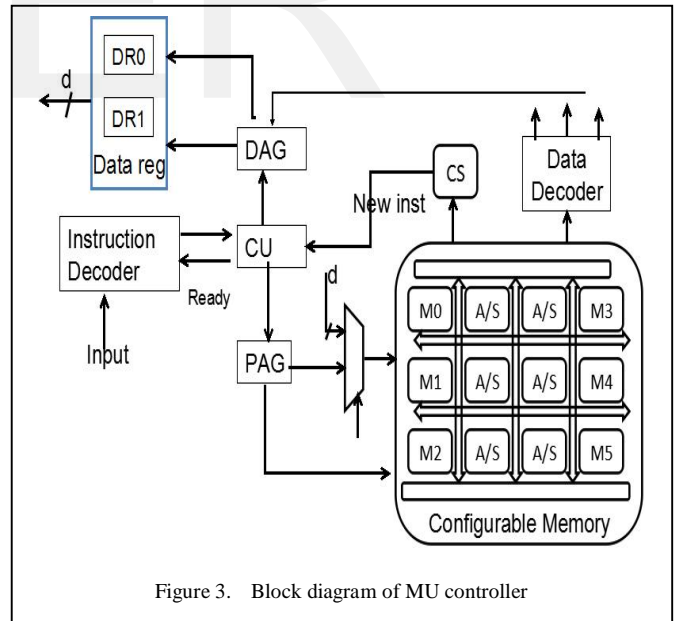


Figure 3. Block diagram of MU controller

TABLE 1
DIFFERENT MODE OF CONFIGURABLE NETWORK

Modulo control unit shown in figure 3. It is consist of a instruction decoder(ID), data decoder(DD), data address generator(DAG), program address generator(PAG), control flag signal(CS), control unit(CU), configurable memory (CM) and two data register (DR0, DR1). Instruction decoder decode

Convolution needs multiplication and addition. The architecture have performed DWT consists of multiplier, adder and some registers. The filter have separated in two parts, one has even index coefficient and other has odd index coefficient. For simplicity we called it even filter and odd filter respectively. If input data samples are even then it filtered by

even filter and odd number of input data samples are filtered by odd filter. In level 1 low pass filtered approx image output have stored in register R for next level computation.

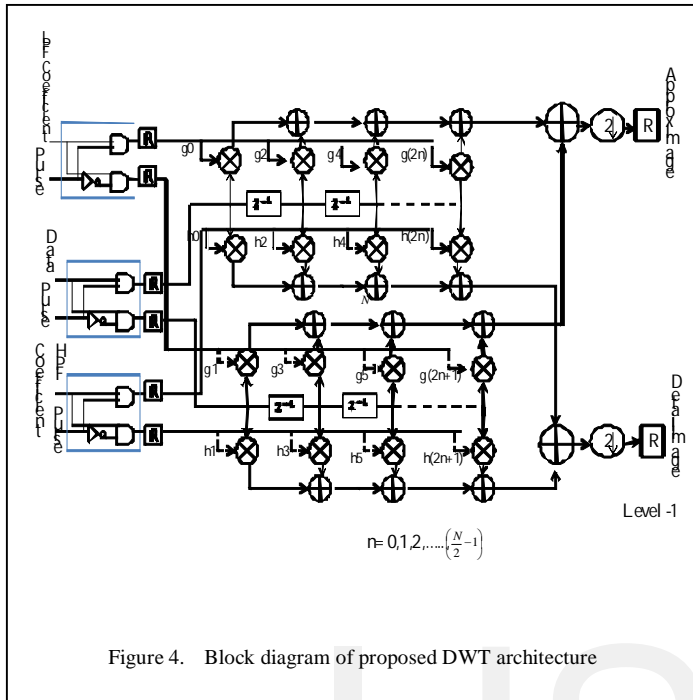


Figure 4. Block diagram of proposed DWT architecture

The input data are divided into two parts and inputted them in parallel. Let the input data will be:
 $a(0), a(2), a(4), a(6), \dots$ even part
 $a(1), a(3), a(5), a(7), \dots$ odd part
 During one cycle, filtering is operated and the filtered values are stored for next level operation. Two input data are filtered during one cycle and the computation period becomes $N/2$. The proposed FFT architecture consisted of Modulo Unit (MU). Number of MU required with respect to FFT computational points. The mathematical representation of decimation-in-time algorithm as follows:

$$X(K) = \sum_{n=0}^{N-1} W^{kn} X(n)$$

$$= \sum_{n=\text{even}} X(n) W^{kn} + \sum_{n=\text{odd}} X(n) W^{kn}$$

$$= \sum_{m=0}^{N/2-1} X(2m) W^{k2m} + \sum_{m=0}^{N/2-1} X(2m+1) W^{k(2m+1)}$$

$W^2_N = W_{N/2}$ with this substitution can be expressed as---

$$X(K) = \sum_{m=0}^{N/2-1} f1(m) W^{km}_{N/2} + W^k \sum_{m=0}^{N/2-1} f2(m) W^{km}_{N/2}$$

$$X(k) = F1(k) + W^k_N F2(k), k=0,1,\dots,N/2-1$$

$$X(k+N/2) = F1(k) - W^k_N F2(k), k=0,1,\dots,N/2-1$$

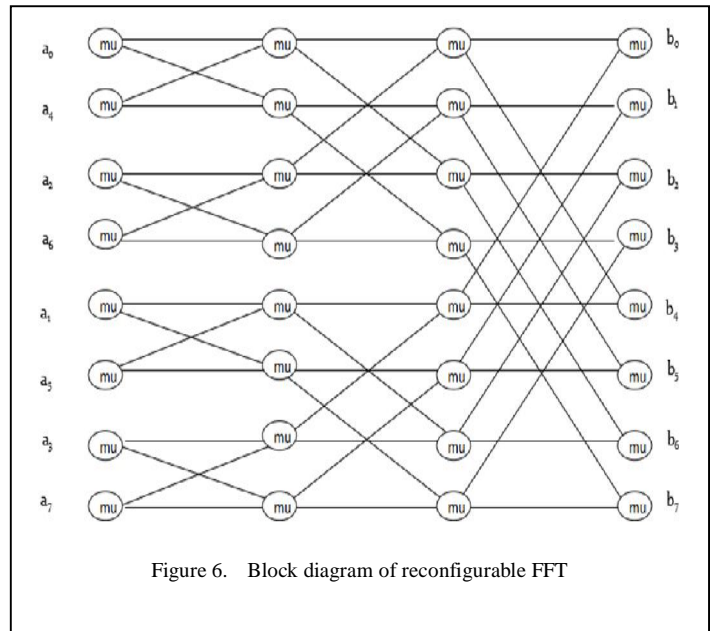


Figure 6. Block diagram of reconfigurable FFT

The decimation of the data sequence can be repeated again and again until the resulting sequences are reduced to one point sequence. For $N=2^J$, this decimation can be performed $J=\log_2 N$ times. Thus the total complex multiplication is reduced to $N/2 \log_2 N$. Number of complex addition is $N \log_2 N$. While one MU array is being reconfigured the other array is computing one step of FFT. Reconfigurable interconnection networks have controlled the MU for computing FFT.

5 EXPERIMENTATION AND RESULTS

The multiplier unit of processing element described in this paper has been implemented. This being heart of the entire system. The multiplier unit of the system has been design to multiply two number. The code for the circuit is written in verilog using XILINX ISE 9.1i version. The multiplier unit of the system is implimented and tested on FPGA board for proper working. Wide range of input combination is given and the output has been verified.

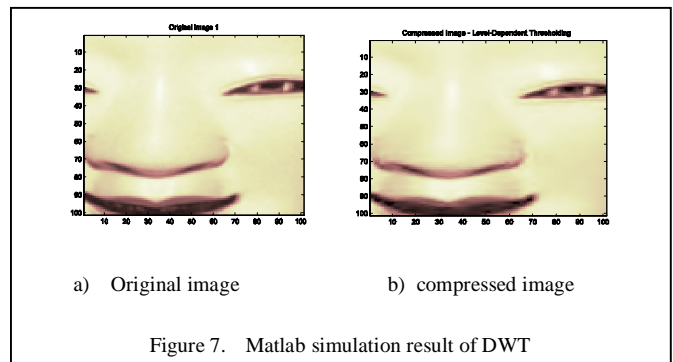


Figure 7. Matlab simulation result of DWT

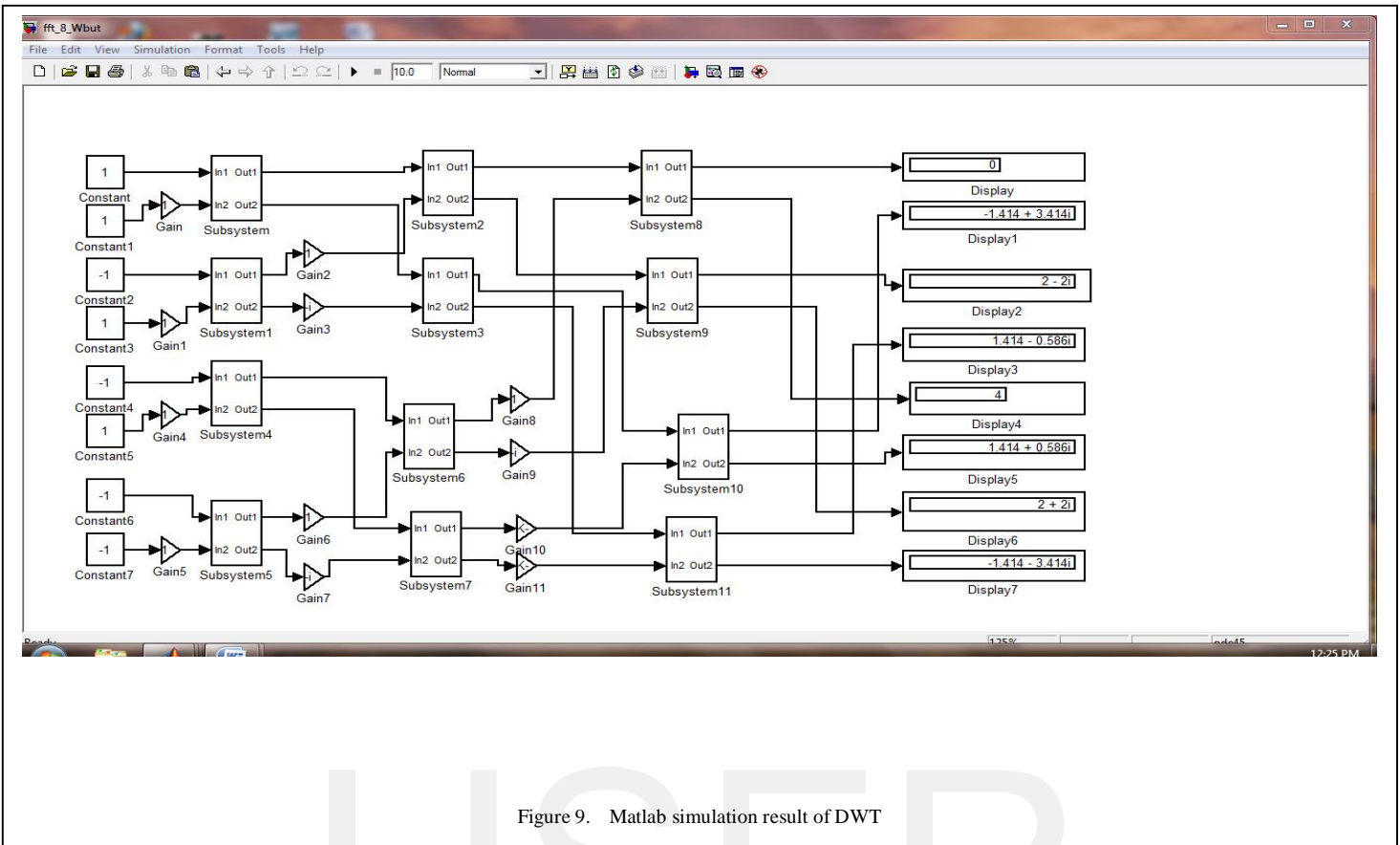


Figure 9. Matlab simulation result of DWT

6 COMPARATIVE STUDY OF HARDWARE UTILIZATION ON DWT ARCHITECTURE BY XILINX ISE 9.1I

6.1 Synthesis Result of DWT pipeline architecture

Device utilization summary:

Cell usages : 212
 Number of Slices: 58 out of 10752
 Number of Slice Flip Flops: 91 out of 21504
 Number of 4 input LUTs: 76 out of 21504
 Number used as logic: 72
 Number used as Shift registers: 1
 Number of IOs: 26
 Number of bonded IOBs: 26 out of 448

Timing Summary:

Speed Grade: -10

Minimum period: 2.192ns
 Minimum input arrival time before clock: 3.494ns
 Maximum output required time after clock: 3.677ns

Speed: 385.840MHz

/Synthesis result of proposed DWT parallel pipeline architecture

Device utilization summary:

Cell usage: 143
 Number of Slices: 29 out of 10752
 Number of Slice Flip Flops: 2 out of 21504
 Number of 4 input LUTs: 47 out of 21504
 Number used as logic: 38
 Number of IOs: 26
 Number of bonded IOBs: 26 out of 448
 Number of DSP48s: 1 out of 48

Timing Summary:

Speed Grade: -10

Minimum period: 0.698ns
 Minimum input arrival time before clock: 2.10ns

Speed: 1252.662MHz

Power summary: P(mW)

Total estimated power consumption: 448

6.2 Time Delay Calculation of DWT, Matrix Multiplication and FFT

FFT:
Total Time 12.681ns (11.870ns logic, 0.811ns route)
(94.1% logic, 5.9% route)

Matrix Multiplication:
Total Time 9.413ns (9.348ns logic, 0.545ns route)
(94.5% logic, 5.5% route)

DWT:
Total Time 14.893ns (12.208ns logic, 2.685ns route)
(82.0% logic, 18.0% route)

7 CONCLUSIONS

In this paper we proposed a reconfigurable architecture for DSP applications, such as DWT, FFT etc. In the proposed DWT architecture N-sample sequence computed in only N/2 clock cycles. This result allows high speed of computation by means of parallel pipelining. The basic computation block like multiplier, adder, subtractor are fixed and DSP function can be implemented only change the configuration of CMs. Numbers of LUTs and switches are reduced. So, the hardware complexity of the architecture and power consumption drastically reduced. The architecture provide a balance performance between ASIC and FPGAs by enhancing the speed and silicon utilization factor closed to ASIC..

ACKNOWLEDGMENT

I would like to thank Dr. Amitabha Sinha for given the innovative idea.

REFERENCES

- [1] S.Shrestha, K.Wahid, "Hybrid DWT-DCT algorithm for biomedical image and video compression applications", IEEE conference on Information sciences signal processing and their applications(ISSPA), 2010,Kulala Lumpur,pages 280-283
- [2] Po-Chih Tseng, Chao-Tsung Huang, and Liang-Gee Chen,"RECONFIGURABLE DISCRETE WAVELET TRANSFORM ARCHITECTURE FOR ADVANCED MULTIMEDIA SYSTEMS"IEEE Trans,2003.
- [3] C. Yu, C. H. I Hsieh, and S. J. Chen, "Design and Implementation of a Highly Efficient VLSI Architecture for Discrete Wavelet Transforms," Proceedings of IEEE Int. Conf on Custom Integrated Circ., 1997, pp. 237-240
- [4] M. Vishwanath, R. M. Owens, and M. J. Irwin, "VLSI Architectures for the Discrete Wavelet Transform," IEEE Trans. on Circ. and Syst. 11, vol. 42, 1995, pp.305-316.
- [5] John G. Proakis, Dimitris G. Manolakis, "Digital Signal Processing Principles, Algorithms Applications", Third Edition, PrenticeHall of India Private Limited, New Delhi, 2005 (©1996 by Prentice Hall, Inc. Upper Saddle River, New Jersey 07458, U.S.A ISBN 81-203-1129-9).
- [6] T.Ayhan,W.Dehaene,M.Verhelst, "A 128:2048/1536 point FFT

hardware implementation with output pruning", IEEE Signal Processing Conference (EUSIPCO), 2014 Proceedings of the 22nd European, 1-5 September 2014, pages 266-270.

- [7] K.K.Parthi, T.Nishitani "VLSI Architectures for Discrete Wavelet Transforms" , IEEE Trans. 1993, Volume:1 Issue:2, pages 191-202.
- [8] E.H. Wold and A.M. Despain. 'Pipeline and Parallel FFT Processors for VLSI Implementations', IEEE Transactions on Computers, vol. C-33, 1984.
- [9] Si-J. Chang, M. Ho Lee, and J. Park, "A high speed VLSI architecture of discrete wavelet transform for MPEG-4," IEEEProc. Of the Int. Conf. On Cons. El. 97, N.Y. 1997
- [10] K. Mayer-Patel and L.A. Rowe, "Exploiting temporal parallelism for software-only video effects processing", Proc. of the sixth ACM international conference on Multimedia", Bristol, U.K., pp.161-169, September, 1998.
- [11] Sharbari Benerjee, Amitabha Sinha , "Performance Analysis of Different DSP Algorithms on Microcontroller and FPGA" Proc. Int'l conf(an IEEE conference)on Advances in Computation tools for Engineering Application,2009
- [12] Andreas Antoniou" Digital signal processing, signal system and filter" Copyright © 2006 by The McGraw-Hill Companies, ISBN 0-07-145424-1).
- [13] J. Fridman, and S. Manolakis, "Discrete Wavelet Transform: Data Dependence Analysis and Synthesis of Distributed Memory and Control Array Architectures," IEEE Trans. On Sig. Proc., vol. 45, 1997, pp. 1291-1308
- [14] A. Grzeszczak, M. K. Mandal, S. Panchanatan, "VLSI Implementation of Discrete Wavelet Transform," IEEE Trans. on VLSI Systems, vol. 4, n. 4, 1996, pp. 421-433.