International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 1 Issue 8, October - 2012

Comparison of FPGA based FIR filter using direct form and convolution based algorithm

Sumit Kumar Maity
Department of Electronics,
Jogesh Chandra Chaudhuri College,
30, Prince Anwar Shah Road,
Kolkata – 700 033, India

Madhusudan Maiti
Department of Electronics
and Communication Engineering,
Bengal Institute of Technology and Management,
Shantiniketan, Bolpur, Birbhum,India

Abstract

To achieve greater flexibility, higher performance (in terms of attenuation and selectivity), better time and environment stability in signal processing operations, we adopted the digital signal processing (DSP) techniques. In this paper we focus on a digital signal processing system (digital filter) that resort a distorted signal and separates a combined signal. This work concentrates at first on the development of a low pass finite impulse response (FIR) digital filter using MatLab FDA tool then we designed two different methodologies for the implementation of a low pass finite impulse response digital filter: FIR using direct form method and FIR using convolution algorithm. These two methodologies are implemented using hardware description language (VHDL) as a design entity, and their synthesis by xilinx synthesis tool on Spartan 3 family XC3S4000l-4fg900 FPGA kit has been done. The experimental result shows that proposed delay free model using convolution algorithm requires less execution time than the traditional structure of the filter.

Keywords : VHDL, DSP, FIR, IIR, FPGA

1. Introduction

In signal processing a system that performs mathematical operations to reduce or enhance certain aspects of a signal is considered as the Digital filter [1]. These are the important class of LTI systems that performs mathematical operations on a sampled, discrete-time signal to reduce or enhance certain aspects of that signal. Basic Fourier Transform theory states that the linear convolution of two sequences in the time domain is the same as multiplication of two corresponding spectral sequences in the frequency domain. Filtering is in essence the multiplication of the signal spectrum by the frequency domain impulse response of the filter [2, 3].

The general form of the digital filter difference equation is given in equation (1).

$$y(n) = -\sum_{k=1}^{N} a_k\, y(n-k) + \sum_{k=0}^{M-1} b_k\, x(n-k) \quad (1)$$

where y(n) is the current filter output, the y(n-k)‘s are previous filter outputs, the x(n-k)‘s are current or previous filter inputs, the ak‘s are the filter‘s feed forward coefficients corresponding to the zeros of the filter, the bk‘s are the filter‘s feedback coefficients corresponding to the poles of the filter, and N is the filter‘s order[4,5].

Depending upon the filter coefficient there are two type of digital filter if the coefficient are fixed then it is frequency selective filter and if the coefficients updated at each iteration in order to minimize the difference between the filter output and the desired signal then it is a adaptive digital filter. The frequency selective filters are of two type infinite impulse response (IIR) and finite impulse response (FIR)[6]. **IIR** filters have one or more non- zero feedback coefficients. That is, as a result of the feedback term, if the filter has one or more poles, once the filter has been excited with an impulse there is always an output. **FIR** filters have no non-zero feedback coefficient. That is, the filter has only zeros, and once it has been excited with an impulse, the output is present for only a finite (N) number of computational cycles[7].

In this paper the architecture of FIR filter is simulated using MATLAB and then it is implemented in FPGA

This FIR filter is verified by implementing a proposed structure of convolution technique using FPGA. The organization of the paper is as follows. In section II, an overview of the FIR filter, designed in MATLAB is presented. The direct form structure of FIR filter, implemented using FPGA is presented in section III. The proposed structure for the verification of FIR filter using convolution technique, implemented in FPGA is presented in section IV. Section V gives the conclusion.

2. Design of the FIR filter

A discrete-time filter produces a discrete-time output sequence for the discrete-time input sequence. In the Finite Impulsive Response (FIR) system, the impulse response sequence is of finite duration, i.e. it has a finite number of non-zero terms and hence the filter coefficients are also constant. The response of the FIR filter depends only on the present and past input samples (a causal system), thus making the system always stable [8, 9, 10]. FIR filter output samples can be computed using the following expression as given in equation (2).

$$y(n) = \sum_{k=0}^{N} h(k)\, x(n-k) \qquad (2)$$

Where X [k] is FIR filter input samples, h[k] are the coefficients of FIR filter frequency response and y[n] are FIR filter output samples. In this paper we have proposed a sixth order low pass FIR filter whose coefficients are shown in the table 1.

Table 1.

Transfer function coefficient	Coefficients
h(0)	0.01
h(1)	0.064
h(2)	0.443
h(3)	0.443
h(4)	0.064
h(5)	0.01

We have designed this low pass FIR filter using MatLab FDA tool with the specification as shown in the table 2. The magnitude and phase plot of the filter is shown in figure 1.

Table 2.

Filter parameters	Specification
Response type	Low pass
Order	6^{th}.
Stable	yes
Design method	Rectangular window
Cut off frequency (w_c)	0.25 (normalized)
Attenuation at cut off frequency	6 db

To design this filter we have used rectangular window method which is defined in equation (3) [11, 12].

$$\omega_R(n) \triangleq \begin{cases} 1, & -\frac{M-1}{2} \le n \le \frac{M-1}{2} \\ 0, & otherwise \end{cases} \qquad (3)$$

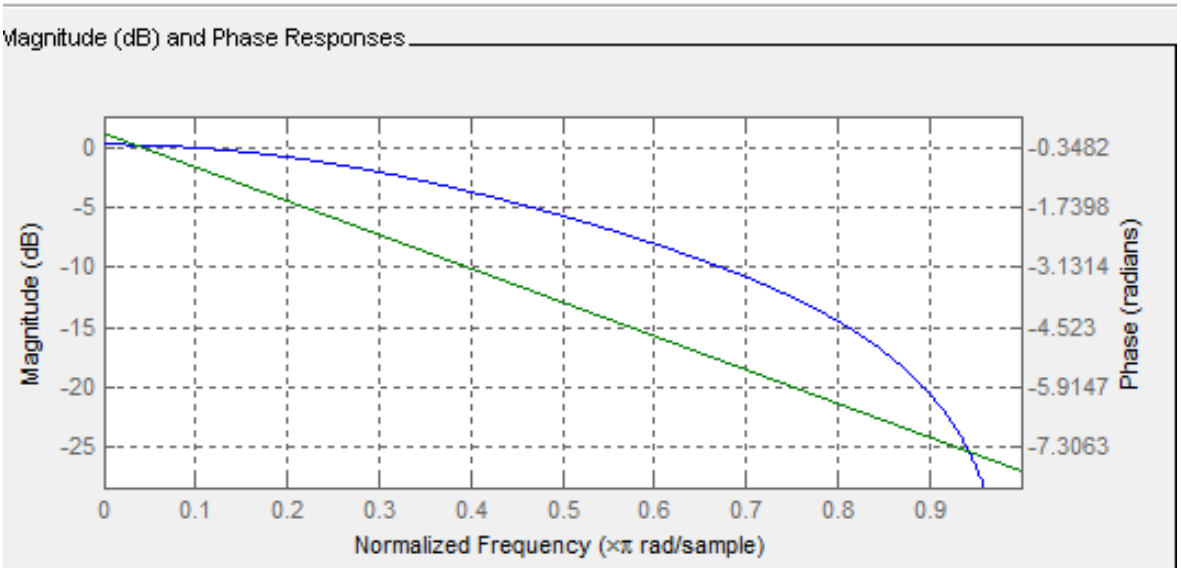


Figure 1. Magnitude and phase plot of the FIR filter.

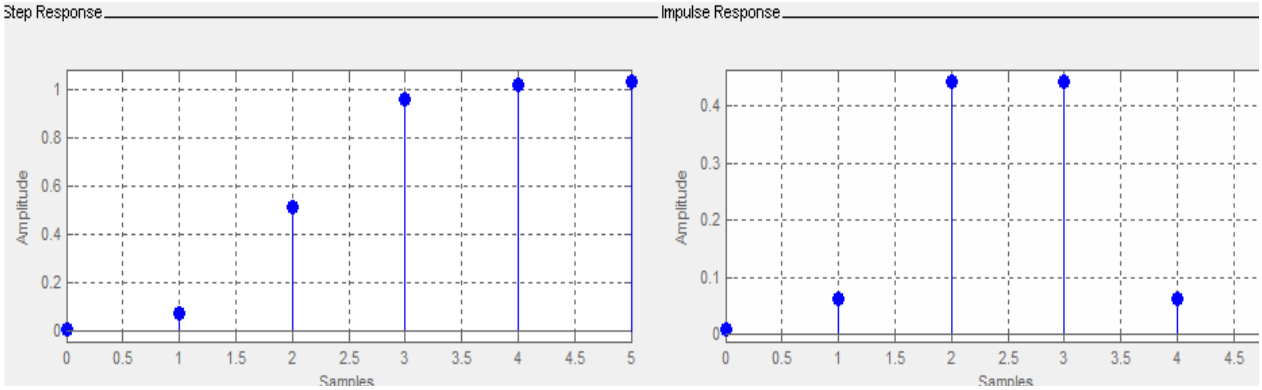


Figure 2. Step and Impulse response of the filter

Where 'M' is the window length in the samples. Here in our FIR filter the window length is six. The impulse response and the step response of the filter are shown in the figure 2.

3. Implementation of the FIR filter using direct form structure

Form the basic difference equation as given in equation (2), we can write the equation of the FIR filter as

$Y(n) = h(0)\,x(n) + h(1)\,x(n-1) + h(2)\,x(n-2) + h(3)\,x(n-3) + h(4)\,x(n-4) + h(5)\,x(n-5)$ (4) [13,14,15].

To implement this equation 4 we have used the direct form structure as shown in the figure 3. Here b (k) is the coefficient and z^{-1} is the delay flip flop[16,17]. To implement this filter in FPGA level we have to approximate the co efficient as shown in the table 3.

Table 3.

Transfer function co-efficient	Actual value	Approximate value	Binary equivalent value
h(0)	0.01	0.015625	0000000.00000100
h(1)	0.064	0.0625	0000000.00010000
h(2)	0.443	0.4453125	0000000.01110010
h(3)	0.443	0.4453125	0000000.01110010
h(4)	0.064	0.0625	0000000.00010000
h(5)	0.01	0.015625	0000000.00000100

Using these approximate values of the transfer function coefficient, we have implemented the filter in Spartan 3 family FPGA xc3s4000l-4fg900.To implement the direct form structure in FPGA we have used the logical elements like D flip flop as delay element adder for the addition operation and shift register[19,20,21]. The timing analysis using impulse response as the input and resource utilization for this structure of the FIR is shown in the figure 4 and figure 5.

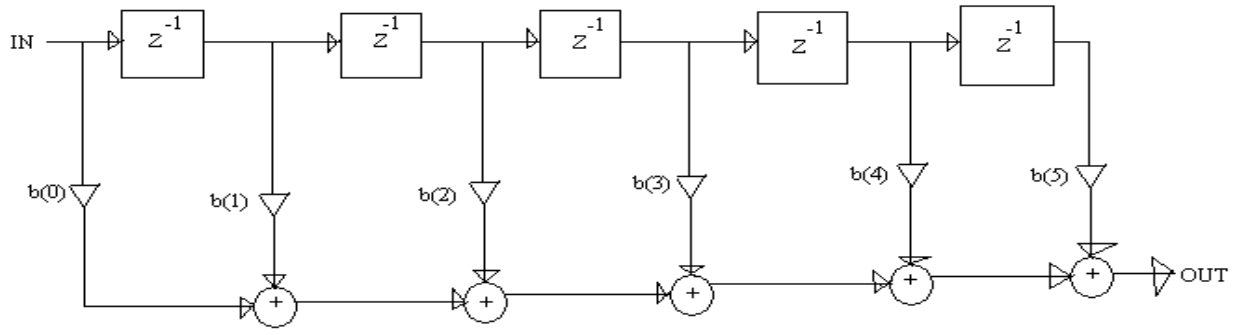


Figure 3. Direct form structure of the filter

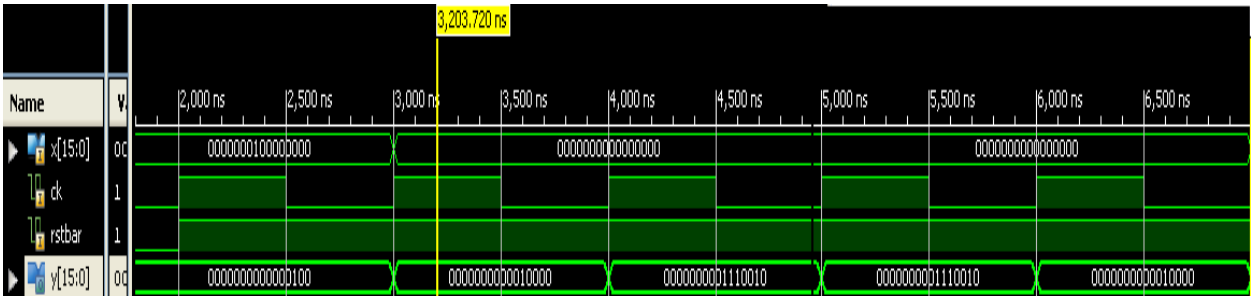


Figure 4. Simulation result of the direct form structure of the FIR filter

Device Utilization Summary			
Logic Utilization	**Used**	**Available**	**Utilization**
Number of Slice Flip Flops	64	55,296	1%
Number of 4 input LUTs	277	55,296	1%
Number of occupied Slices	199	27,648	1%
Number of Slices containing only related logic	199	199	100%
Number of Slices containing unrelated logic	0	199	0%
Total Number of 4 input LUTs	277	55,296	1%
Number of bonded IOBs	32	633	5%
Number of BUFGMUXs	1	8	12%
Average Fanout of Non-Clock Nets	2.96		

Figure 5. Device utilization summary of the direct form FIR filter

4. Implementation of the FIR filter using proposed structure

Here in this section we have proposed the structure of FIR filter using the convolution technique[22,23,24,25]. We know that the output of any LTI system is given by equation 4.

$$y[n] = x[n] * h[n] \quad (4)$$

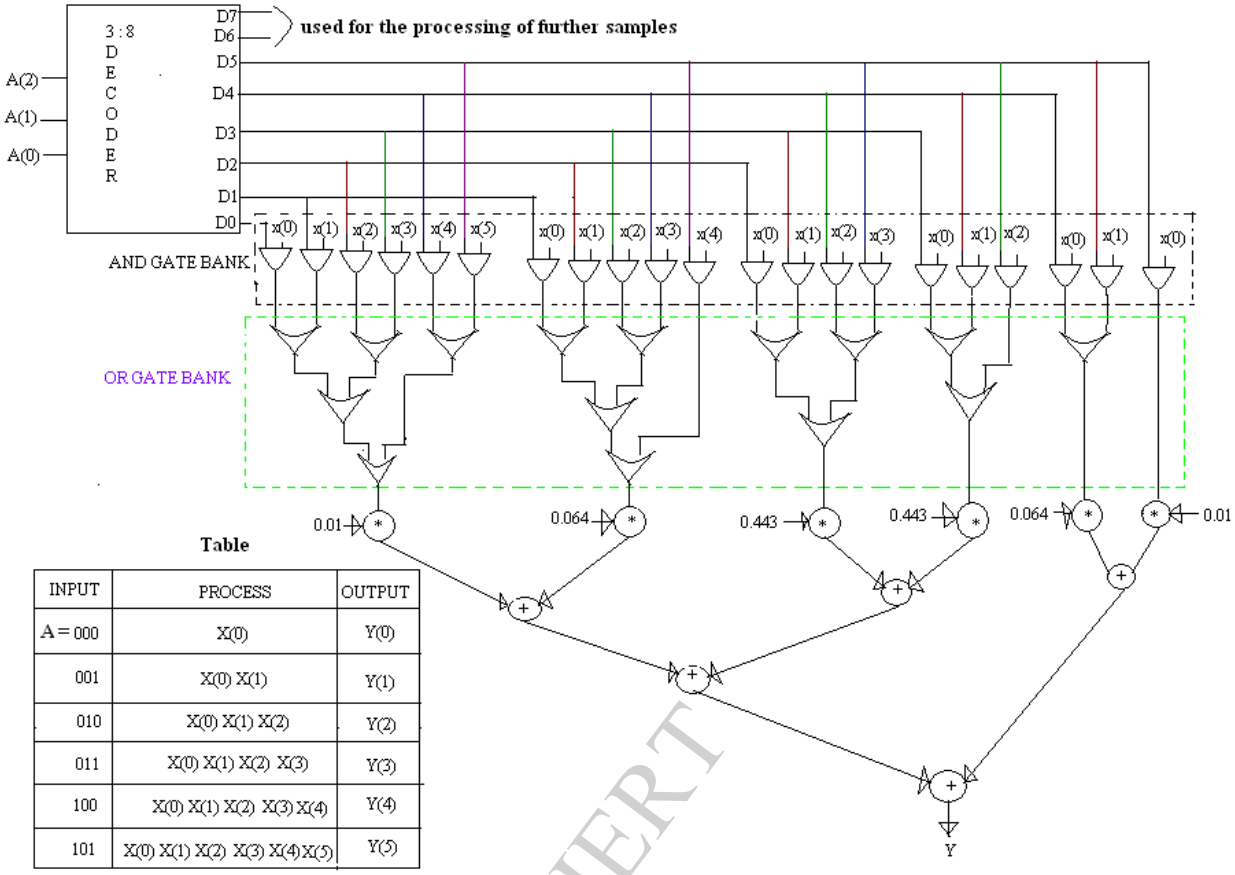
The system function of the FIR filter h[n] can be represented by the equation 5[26, 27].

$$h[n] = \sum_{k=0}^{M} b_k \, \delta[n-k] \quad (5)$$

Thus the output of the FIR filter y(n) can be defined using the equation 6[28,29,30].

$$y[n] = \sum_{k=0}^{M} b_k \, x[n-k] \quad (6)$$

Using these equation we can proposed the circuit of the FIR filter , using logical elements like banks of AND gate , OR gate , decoder , shift register and adder as shown in the figure(6).To operate this filter we have to use the table as given below. At first when the input of the decoder is set at "000" then D_0 output of the decoder will be enabled. This output will takes the input x(0) as shown in the figure (6) through the banks of AND and OR gates then the input x(0) will be multiplied with the first coefficient b_0.This process of multiplication will be performed by the shift register as shown in the figure(6), at last through the banks of adder we get our output y(0). Similarly if we set the input of the decoder "001" then D_1 output of the decoder will be enabled .This output D_1 will take input x(1) and x(0) through banks of AND and OR gates. In this way the process continues for all the samples. The timing analysis using impulse response as the input and resource utilization for the proposed structure of the FIR filter is shown in the figure 7and figure 8. The impulse response of this proposed structure is same as the direct form structure.



Table

INPUT	PROCESS	OUTPUT
A = 000	X(0)	Y(0)
001	X(0) X(1)	Y(1)
010	X(0) X(1) X(2)	Y(2)
011	X(0) X(1) X(2) X(3)	Y(3)
100	X(0) X(1) X(2) X(3) X(4)	Y(4)
101	X(0) X(1) X(2) X(3) X(4) X(5)	Y(5)

Figure 6. Circuit diagram of the proposed FIR filter

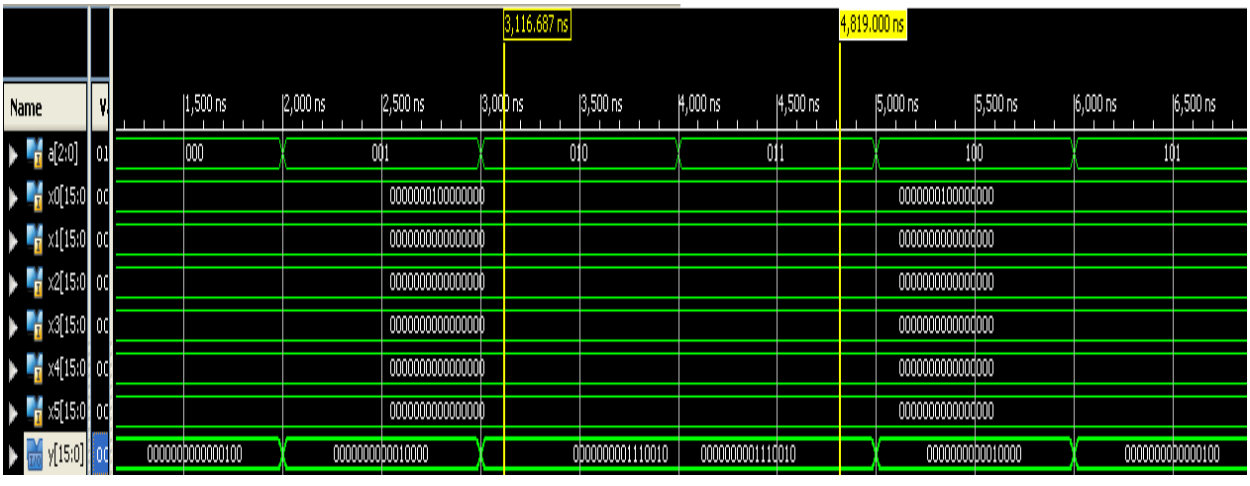


Figure 7. The timing analysis using impulse response

Device Utilization Summary				[-]
Logic Utilization	**Used**	**Available**	**Utilization**	**Note(s)**
Number of Slice Latches	6	55,296	1%	
Number of 4 input LUTs	443	55,296	1%	
Number of occupied Slices	247	27,648	1%	
Number of Slices containing only related logic	247	247	100%	
Number of Slices containing unrelated logic	0	247	0%	
Total Number of 4 input LUTs	443	55,296	1%	
Number of bonded IOBs	97	633	15%	
Average Fanout of Non-Clock Nets	2.97			

Figure 8. Device utilization summary of the proposed FIR filter

5. Conclusion

In this paper, we have presented a model that can implement a FIR filter using only combinational logic blocks. In this model the critical path delay is reduced and the power consumption for both structures is equal 0.268 watt. Thus using the delay free model we can reduce the computation time of the digital filter.

6. Reference

[1] Bishnupriya Bhattacharya and Shuvra S. Bhattacharyya ―Parameterized Data flow Modeling for DSP System‖, *IEEE TRANSACTIONS ON SIGNAL PROCESSING 49, NO. 10, OCTOBER 2001.*

[2] Edward A. Lee and David G. Messerschmitt ―Synchronous data flow*", Proceedings of the IEEE, vol. 75, no. 9, p 1235-1245, September, 1987.*

[3] R.E.Crochiere and A.V.Oppenheim,‖ Analysis of linear Digital Networks‖, *PROCEEDINGS OF THE IEEE, VOL. 63, NO. 4, APRIL 1975*

[4] Parhi, K.K « Algorithm Transformation Technique for concurrent processors », *PROCEEDINGS OF THE IEEE, VOL. 77, NO. 12, DECEMBER 1989.*

[5] Lori E. Lucke and Keshab K. Parhi ―Data flow Transformation for critical path time reduction in high level DSP synthesis*", IEEE TRANSACTIONS ON COMPUTER-AIDED DESIGN OF INTEGRATED CIRCUITS AND SYSTEMS, VOL. 12, NO. 7, IULY 1993.*

[6] Tracy C. Denk, Member, IEEE, and Keshab K. Parhi ‖ Exhautive Scheduling and retiming of digital signal processing system‖, *IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS—II: ANALOG AND DIGITAL SIGNAL PROCESSING, VOL. 45, NO. 7, JULY 1998.*

[7]. I. Karkowski and R.H.J.M. Otten‖Retiming Synchronous circuit with Imprecise delay‖ *32nd ACM/IEEE Design Automation Conference.*

[8]. Jia Wang and Hai Zhou ―A efficient Incremental algorithm for min-area retiming‖, *DAC 2008, June 8–13, 2008, Anaheim, California, USA*

[9] Ulrich Weinmann,Wolfgang Rosenstiel ―Technology mapping for sequential circuit based on retiming Technique‖ *Design Automation Conference, 1993, with EURO-VHDL '93. Proceedings EURO-DAC '93. European*

[10] Timothy W. O'Neil, Edwin H.-M. Sha, ―Retiming synchronous data flow graph to reduce execution time‖, *IEEE TRANSACTIONS ON SIGNAL PROCESSING, VOL. 49, NO. 10, OCTOBER 2001.*

[11] Parhi, K.K , D.G.Messerschcmitt,‖Pipeline interving and parallism in recursive digital filter part 1: pipelining using scattered look a head and decomposition", *IEEE TRANSACTIONS ON ACOUSTICS. SPEECH. AND SIGNAL PROCESSING, VOL. 31. NO. 7, JULY 1989.*

[12] Parhi, K.K , D.G.Messerschcmitt,‖ ―Pipeline interving and parallism in recursive digital filter part – II : Pipelining incremental block filtering‖ *IEEE TRANSACTIONS ON ACOUSTICS, SPEECH, AND SIGNAL PROCESSING VOL 37. NO. 7. JULY 1989*

[13] Lih-Gwo Jeng, Liang-Gee Chen, ―Rate optimal DSP synthesis by pipeline and minimum unfolding‖ *IEEE TRANSACIIONS ON VERY LARGE SCALE INTEGRATION (VLSI) SYSTEMS, VOL. 2, NO. 1, MARCH 1994.*

[14] Wolf, M.E., Lam, M.S. ―A loop transformation theory and an algorithm to maximize parallelism‖ *IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS, VOL. 2, NO. 4, OCTOBER 1991.*

[15] TimothyW. O'Neil, Edwin H.-M. Sha ―unfolding to minimize inter- iteration dependencies‖ *Proceeding of 16th . IASTED International conference Parallel and Distributed computing and systems,Nov 9-11,2004, Cambridge,MA,USA.*

[16] Duen-Jeng Wang and Yu Hen Hu ―Rate optimal scheduling of recursive DSP algorithm by unfolding‖, *IEEE*

TRANSACTIONS ON CIRCUITS AND SYSTEMS-I: FUNDAMENTAL THEORY AND APPLICATIONS, VOL. 41, NO. 10, OCTOBER 1994.

[17] Liang-Fang Chao Edwin Hsing-Mean Sha ―unfolding and retiming data flow DSP Program for RISC multiprocessor scheduling*" IEEE International Conference on Acoustics, Speech, and Signal Processing, 1992. ICASSP-92 Vol 5, 1992.*

[18] TimothyW. O'Neil Edwin H.-M. Sha ―Rate optimal graph transformation via Extended retiming and unfolding‖, *Proceedings of the IASTED International Conference Parallel and Distributed Computing and Systems (PDCS '99) Nov 3-6,1999 – Cambridge, MA, USA.*

[19] Qingfeng Zhuge, Chun Xue, Zili Shao, Meilin Liu,‖Design optimization and space minimization considering timing and code size via retiming and unfolding*" Microprocessors and Microsystems, Volume 30, Issue 4, Pages 173-183, 6 June 2006.*

[20] Timothy W. O'Neil , Edwin H.-M. Sha ―Combining Extended retiming and unfolding for rate optimal graph transformation‖ *Journal of VLSI Signal Processing Systems, Volume 39 Issue 3, March 2005.*

[21] T. W. O'Neil , and E. H.-M. Sha ―Optimal graph transformation using extended retiming with minimal unfolding‖ *in the Proceedings of the IASTED International Conference on Parallel and Distributed Computing Systems, Las Vegas, NV, November, 2000, pp. 128-133.*

[22] Liang-Fang Chao , Edwin Hsing-Mean Sha ―Scheduling data flow graph via retiming and unfolding‖ *IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS, VOL. 8, NO. 12, DECEMBER 1997.*

[23] Liang-Fang Chao, Edwin Hsing-Mean Sha ―A Efficient retiming and unfolding‖ IEEE International Conference on Acoustics, Speech, and Signal Processing, 1993. ICASSP-93.,

[24]]M. RENFORS,Y. NEUVO, ―The maximum sampling rate of digital filter under hardware speed constraints*" IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS, VOL. CAS-28, NO. 3, MARCH 1981*

[25] N. Wehn , J. Biesenack , Peter Duzy , T. Langmaier , M. Münch , Michael Pilsl , S. Rumler ―Scheduling of Behavioral VHDL by retiming Technique‖ *Proceedings of the conference on European design automation, EURO-DAC '94*

[26] Qingfeng Zhuge, Zili Shao, Bin Xiao, Edwin H.M. Sha ―Design space minimization with timing and code size optimization for Embedded DSP‖ *Proceedings of the 1st IEEE/ACM/IFIP international conference on Hardware/ software codesign and system synthesis, CODES+ISSS '03.*

[27] Soyata, T. Friedman, E.G. Mulligan, J.H., Jr. ―Incorporating interconnect , register and clock distribution delay into the retiming process*" IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems,Vol 16, pp 105 – 120, Jan 1997.*

[28] José Monteiro, Srinivas Devadas, Abhijit Ghosh‖Retiming sequential circuit for low power‖ *Proceedings of the IEEE/ACM international conference on Computer-aided design, ICCAD '93.*

[29] Santosh Chede , Kishore Kulat , Rajesh Thakare ―A significance of VLSI Technique for low power real Time system‖ *2010 International Journal of Computer Applications (0975 - 8887),Volume 1 – No. 22.*

[30] Mandeep Kaur, Vikas Sharma ―Analysis of various algorithm for low power consumption in embedded system using different architecture‖ *International Journal of Electronics Engineering, 2(1), 2010,*