2002 23rd INTERNATIONAL CONFERENCE ON MICROELECTRONICS

PROCEEDINGS

NIŠ

MIEL 2002

Volume 2

Niš, Yugoslavia
12-15 May 2002

organized by
Yugoslavia IEEE Section - ED/SSC Chapter

under the co-sponsorship of
IEEE Electron Devices Society

with the cooperation of
IEEE Solid-State Circuits Society

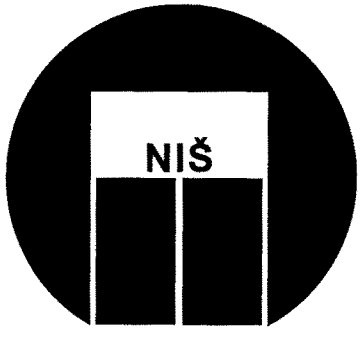
IEEE

IEEE Catalog No. 02TH8595
ISBN 0-7803-7235-2

2002 23rd INTERNATIONAL CONFERENCE ON MICROELECTRONICS

PROCEEDINGS

Volume 2



MIEL 2002

Niš, Yugoslavia
12-15 May 2002

organized by
Yugoslavia IEEE Section - ED/SSC Chapter

under the co-sponsorship of
IEEE Electron Devices Society

with the cooperation of
IEEE Solid-State Circuits Society

IEEE Catalog No. 02TH8595
ISBN 0-7803-7235-2

Synthesis of Folded Fully Pipelined Bit-Plane Architecture

I. Milentijević, I. Nikolić, V. Ćirić, O. Vojinović, and T. Tokić

Abstract - This paper describes the application of folding technique to the Bit-Plane systolic FIR filter Architecture (BPA). We present the transformation of original DFG (Data Flow Graph) that enables the application of folding technique and the synthesis of fully pipelined folded architecture. The array is restricted for the factor m, where m represents the coefficient length. The number of basic cells in target architecture is reduced to the number of basic cells in one plane of source architecture. Also, the total number of latches corresponds to the number of latches in one plane of the BPA. The hardware restriction is paid by decreasing of throughput for slightly more than m times.

I. Introduction

Finite Impulse Response (FIR) filters have been widely used for video rate digital filtering. Regular structure of FIR filter algorithm is suitable for implementations on systolic arrays [1]. Pipelined cellular arrays represent an appropriate implementation approach for arithmetic circuits where a high computational speed is required [2,3]. They are designed in the form of regularly repeated patterns of identical circuits. Thus, due to their geometrical regularity, they are suitable for VLSI implementations, either as stand-alone modules or as a part of a complex digital data path [4].

The operations of a FIR filtering can be arranged to a sequential summation of the products using the transposed direct form. However, the approach is limited by the speed of the multipliers. To overcome this deficiency it is common to pipeline multipliers and to combine multiplications and the accumulation of products into an array. This is the basic principle of Bit-Plane Architecture (BPA) [5, 6]. The BPA is highly regular architecture, which allows extensive pipelining, regular layout, high computational throughput, truncation of Least Significant Bits (LSBs) of intermediate results without any loss of accuracy, and programmability of coefficients. All these features point why the BPA is our candidate for application of folding technique.

It is well known that performances and cost of any digital circuit depend on circuit design style. Therefore, creating a given architecture, to establish optimal area-time-power tradeoff, a careful choice of circuit design style to use is necessary. In synthesizing DSP architectures, it is important to minimize the silicon area of the integrated circuits, which is achieved by reducing the number of functional units (such as multipliers and adders), registers, multiplexers, and interconnection wires. The folding transformation is used to systematically determine the control circuits in DSP architectures where multiple algorithm operations are time multiplexed to a single functional unit [7]. By executing multiple algorithm operations on a single functional unit, the number of functional units in the implementation is reduced, resulting in integrated circuit with low silicon area [8].

The aim of this paper is the application of folding technique to the BPA keeping all desirable features of BPA in the target architecture. However, the folding transformation can not be applied to the BPA in a straight foreword manner, because the algorithm is based on resorting of partial products, so that multiplication of coefficient and input data word is not recognized as an operation i.e. a node in Data Flow Graph (DFG). Therefore, the additional transformation of the original DFG for the BPA which prepares DFG for the application of folding technique should be found. One solution is presented in [9] where proposed transformation removes latches in carry and sum paths, so the pipelining inside the plane is not employed. It leads to the successful application of the folding technique, but the obtained target architecture is suitable only for small number of coefficients (plane length is equal to the number of coefficients). All these facts have motivated us to propose a new transformation of source DFG for BPA that enables the application of folding technique. As a result we obtain folded fully pipelined BPA for FIR filtering. In this paper we describe the complete synthesis path from the source DFG to the target architecture. The obtained architecture is presented by DFG, and functional block diagram.

II. Folding Technique

The folding technique is introduced by K.K. Parhi and described in [7, 8]. With aim to clarify the applying of folding technique to the BPA we give a brief review of folding transformation.

The synthesis of folded data path is explained in Fig. 1 a) and Fig. 1 b). Fig 1 a) shows an edge $U \rightarrow V$ with $w(e)$ delays, while Fig. 1 b) depicts the corresponding folded data path. The data begin at the functional unit H_u which has P_u pipelining stages, pass through

$$D_F(U \rightarrow V) = Nw(e) - P_u + v - u \qquad (1)$$

delays, and are switched into the functional unit H_v at the time instances $Nl+v$, where N is the number of operations folded to a single functional unit (folding factor), while u and v are the folding orders of nodes U and V that satisfy

I. Milentijević, I. Nikolić, O. Vojinović, and T. Tokić are with Faculty of Electronic Engineering, University of Niš, Beogradska 14, Yugoslavia, E-mail: milentijevic@elfak.ni.ac.yu

V. Ćirić is with A.D. Tigar, Pirot, Yugoslavia. E-mail: mciric@tigar.com

0-7803-7235-2/02/$10.00 © 2002 IEEE

$N-1 \geq u, v \geq 0$. A folding set, S, is defined as an ordered set of operations, which contains N entries, executed by the same functional unit. For a folded system to be realizable, $D_F(U \rightarrow V) \geq 0$ must hold for all of the edges in the DFG. Once valid folding sets have been assigned, retiming can be used to satisfy this property or determine that the folding sets are not feasible [7].

After this short description of folding technique and involving of suitable notation, let us to introduce the BPA as a source architecture.

III. Bit-plane architecture

Output words $\{y_i\}$ FIR filter are computed as

$$y_i = c_0 x_i + c_1 x_{i-1} + \ldots + c_{k-1} x_{i-k+1}, \quad (2)$$

where $c_0, c_1, \ldots, c_{k-1}$ are coefficients while $\{x_i\}$ are input words.

The bit–plane architecture (BPA) is semi-systolic architecture which provides regular connections with extensive pipelining and high computational throughput. The BPA is basic architecture for synthesis of folded architecture (FA), so we give a brief description of the BPA. In order to explain the BPA following notation is adopted:

m – coefficient word length,

k – number of coefficients $(c_0, c_1, \ldots, c_{k-1})$, and

c_i^j – bit of coefficient c_i (with weight 2^j).

The BPA is obtained by resorting of the partial products of different multipliers as it is shown in Fig.2.

With fine-grained pipelining, the splitted parts of the multiplications become input word times 1-b coefficient multiplications, the partial products. These are just logical AND function between the input word and coefficient bit. In the first bit–plane the least significant partial products of all coefficients are computed and accumulated (Fig. 2.). The output of the first bit–plane is shifted by one weight and then the second lowest significant partial products are processed in the second bit plane and so on [5,6]. Starting bit-plane processing with the LSB's first, enables to truncate one LSB of the intermediate output signal after each bit–plane without any loss of accuracy in the more significant weights. We choose this architecture as a basis for the synthesis of the fully pipelined folded FIR filter architecture.

IV. Synthesis of folded architecture

The folding transformation can not be applied to the BPA directly (Fig. 2.), because the algorithm is based on resorting of partial products, so that multiplications of coefficients and input words are not recognized as operations, i.e. nodes in DFG (Fig. 2.). Each multiplication node in the DFG, shown in Fig. 2., represents one raw of basic cells (full adder and *and* gate) in the array. Thus, one multiplication is distributed through the whole array. Also, additions are performed in rows of basic cells and accumulation is provided by carry–save arithmetic along the array.

The solution proposed in [9] declares all operations in one plane as one operation "plane" and provides time multiplexing to a single functional unit. Transformed DFG is shown Fig.3. [9]. Delays between planes are removed, as well as delays in the addition path. It allows simultaneous operation of plane units, but requires additional latches inside the plane. Transformed DFG is not suitable for implementation because of broadcasting line for input data words, but it is well prepared for folding. However, it leads to the folded BPA which is suitable for the filtering with small number of coefficients, because the transformation, proposed in [9], TDFG1 (Fig. 3), removes latches in carry and sum paths. Thus, the pipelining inside the plane is not employed and the critical path depends on the plane length, i.e. number of coefficients. It motivated us to find another solution for folding of the BPA which will enable the synthesis of fully pipelined folded BPA (FPFBPA).

Now, the crucial question is how to form the folding sets. The TDFG1 enables the existence of only one folding set $S=\{1,2,3,4\}$ with four "plane" operations. Let us back to the Fig. 3. Suppose that we have obtained TDFG1, but that we had not formed folding sets yet. If we remove multiplications by ½ from addition path we should involve multiplications by *2* in the input data path. It is shown in Fig. 4. The next step is the resorting of partial products collecting all coefficient bits from each coefficient, separately (Fig. 5.). Now, we have different "plane" from the "plane" in the BPA. This resorting requires delays in input data path. Fig. 5. depicts this step. The DFG from Fig. 5. is not prepared for folding, yet. The obstacle is the existence of latches in input data path. The last step, before folding, shown in Fig. 6. assumes the removing of delays from input data path and their involving into the addition path. This is followed by reverse ordering of coefficients. Finally, we have well prepared DFG, TDFG2, for application of folding technique. The folding sets are formed as it is shown in Fig. 6. The operations which will be folded are denoted with dashed lines. For the BPA with $k=3$ and $m=4$ there are three folding sets S_1, S_2 and S_3 each of them containing four operations.

One operation from the TDFG2 (Fig. 6.) includes the set of AND gates and full adders from the source BPA. The folding factor is equal to the coefficient length, $N=m=4$. Thus, the results of folding equations (1) satisfy the condition $D_F(U \rightarrow V) \geq 0$ and it proves that TDFG2 is well prepared for folding. Obtained fully pipelined folded bit-plane architecture, FPFBPA, is presented in Fig. 7., while Fig. 8. describes functional block diagram of the FPFBPA.

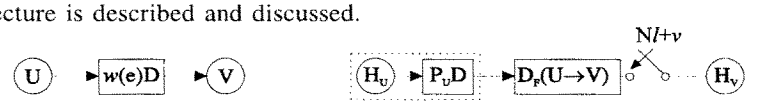
The FPFBPA is described as parameterized FIR filtering core in VHDL and its functionality is proved thorough the logic simulation.

V. Conclusion

The proposed transformation (Fig. 4., Fig. 5. and Fig 6.), enables the synthesis of fully pipelined folded FIR filter architecture, FPFBPA. There are k folding sets, i.e. the number of folding sets is equal to the number of coefficients. Each folding set contains m operations. The array is restricted for the factor m. The number of basic cells is reduced to the number of basic cells in one plane of source architecture. Also, the total number of latches corresponds to the number of latches in one plane of the BPA. The extensive pipelining in FPFBPA is paid by

involving two multiplexers per each basic cell. The critical path is extended for one additional multiplexer, so the basic clock frequency is slightly decreased. Thus, the throughput is decreased for slightly more than m times in respect to the BPA. The derived architecture is described and discussed.

The application of folding technique to the bit-plane architecture allowed the achieving of throughput requirements for FIR filtering on integrated circuits with low silicon area.



a) An edge $U\rightarrow V$ with $w(e)$ delays b) The corresponding folded data path.

Fig. 1. The synthesis of folded data path

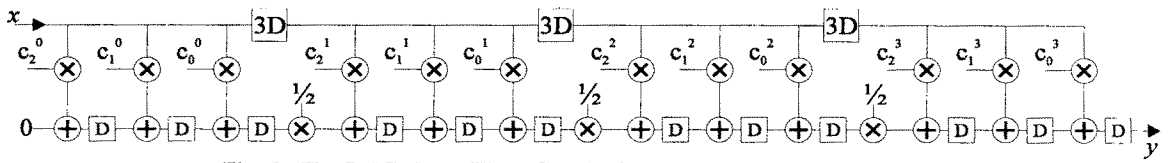


Fig. 2. The DFG (Data Flow Graph) for the BPA with $k=3$ and $m=4$

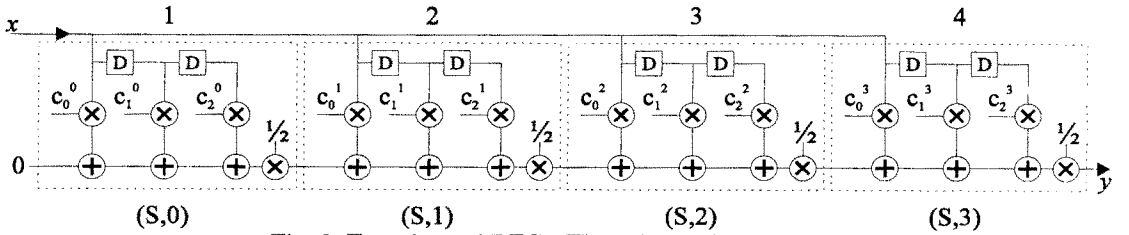


Fig. 3. Transformed DFG - TDFG1 with $k=3$ and $n=4$

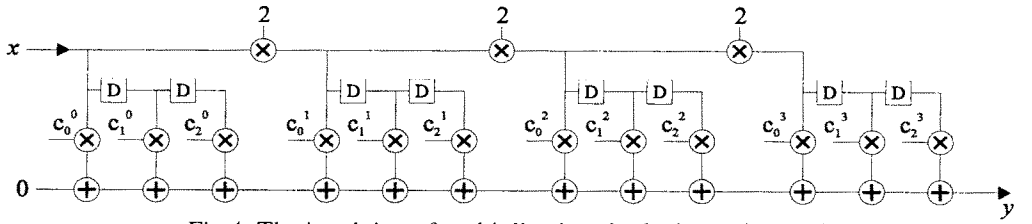


Fig 4. The involving of multiplications in the input data path

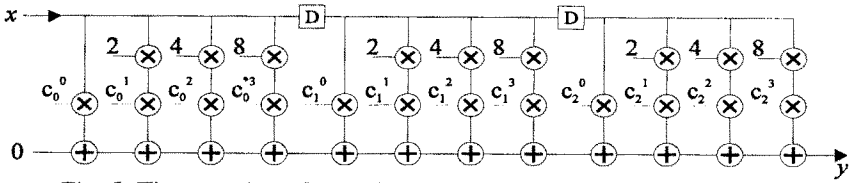


Fig. 5. The resorting of partial products collecting all coefficient bits

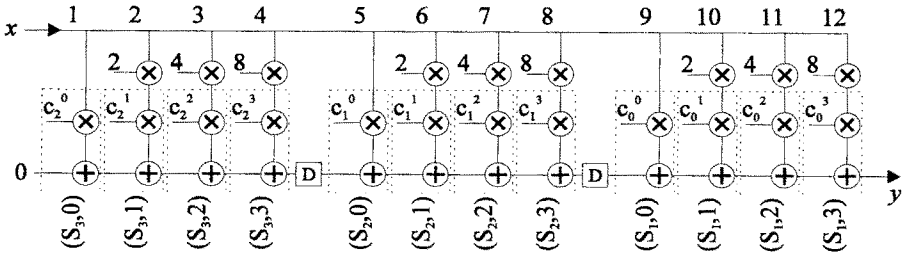


Fig. 6. Transformed DFG – TDFG2 with $k=3$ and $n=4$

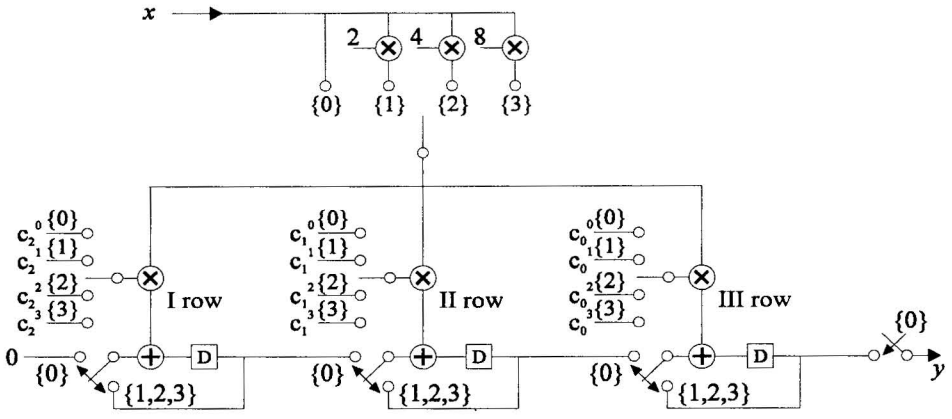


Fig. 7. Folded architecture FPFBPA with folding sets S_1, S_2 and S_3

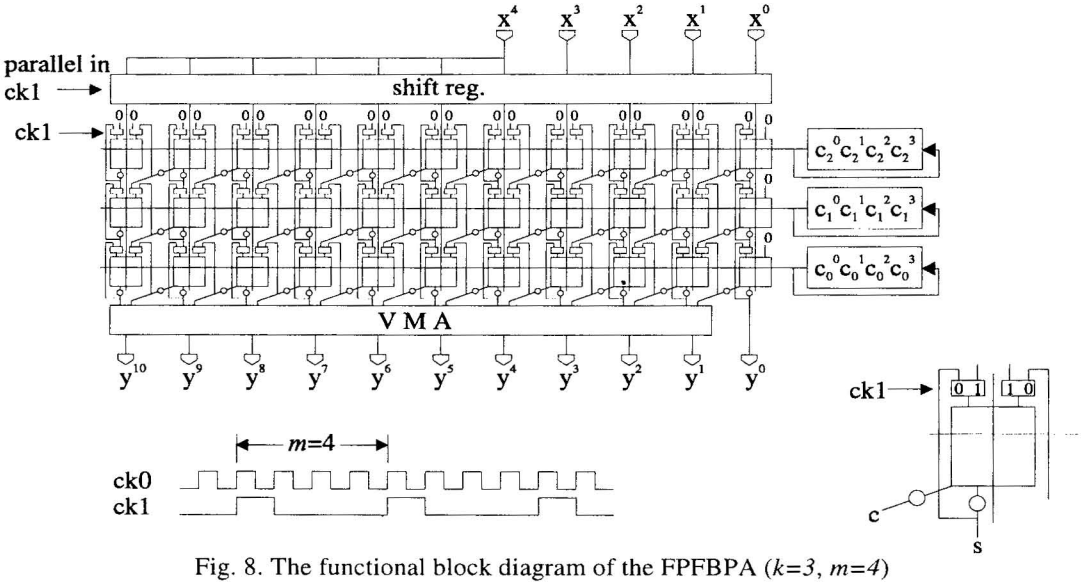


Fig. 8. The functional block diagram of the FPFBPA ($k=3$, $m=4$)

References

[1] Y-C. Lin, F-C, "Classes of Systolic Arrays for Digital Filtering", *Int. J. Electronics*, Vol. 70, No. 4, 1991, pp. 729-737.

[2] I. Milentijević, M. S. Stojčev, D. Maksimović, "Configurable Digit - Serial Convolver of Type F", *Microelectronics Journal*, Vol. 27. No. 6, Sep. 1996, pp. 559-566.

[3] I. Milentijević, I. Milovanović, E. Milovanović, M. Tošić, M. Stojčev, "Two - Level Pipelined Systolic Arrays for Matrix - Vector Multiplication", *Journal of Systems Architecture, The EROMICRO Journal*, Vol. 44, No. 5, Feb. 1998, pp. 383 -387.

[4] P. Corsonello, S. Perri , and G. Cocorullo., "Area-Time-Power Tradeoff in Cellular Arrays VLSI Implementations", *IEEE Transaction on Very Large Scale Integration (VLSI) Systems*, Vol. 8, No. 5, Oct. 2000, pp. 614-624.

[5] T. Noll, "Semi-systolic Maximum Rate Transversal Filters with Programmable coefficients", *Workshop of Systolic Architectures*, Oxford, 1986, pp. 103-112.

[6] D. Reuver, H. Klar, "A Configurable Convolution Chip with Programmable Coefficients", *IEEE Journal of Solid State Circuits*, Vol. 27, No. 7, July 1992, pp. 1121 -1123.

[7] K. K. Parhi, *VLSI Digital Signal Processing Systems (Design and Implementation)*, John Wiley & Sons, In., New York, 2000.

[8] T. C. Denk, K. K. Parhi, Synthesis of Folded Pipelined Architectures for Multirate DSP Algorithms, *IEEE Transaction on Very Large Scale Integration (VLSI) Systems*, Vol. 6, No. 4, Dec. 1998, pp. 595-607.

[9] I. Milentijević, T. Tokić, I. Nikolić, O. Vojinović and V. Ćirić, Synthesis of Folded FIR Filter Architecture With Reordered Partial Products, *Proceedings of a Workshop on Computational Intelligence and Informational Technologies*, June 20-21, Niš, Yugoslavia, 2001, pp. 155-160