DSD 2005

Eighth EUROMICRO Conference on DIGITAL SYSTEM DESIGN

Architectures, Methods and Tools

Porto, Portugal
30 August – 3 September 2005

Editor: Christophe Wolinski

Proceedings

8th EUROMICRO CONFERENCE ON DIGITAL SYSTEM DESIGN

Porto, Portugal, August 30 – September 3, 2005

Universidade do Porto
Faculdade de Engenharia
FEUP

IEEE COMPUTER SOCIETY
http://computer.org
Los Alamitos, California
Washington • Brussels • Tokyo

IEEE Computer Society Order Number P2433
ISBN 0-7695-2433-8
Library of Congress Number 2005930138

Additional copies may be ordered from:

IEEE Computer Society
Customer Service Center
10662 Los Vaqueros Circle
P.O. Box 3014
Los Alamitos, CA 90720-1314
Tel: + 1 800 272 6657
Fax: + 1 714 821 4641
http://computer.org/cspress
csbooks@computer.org

IEEE Service Center
445 Hoes Lane
P.O. Box 1331
Piscataway, NJ 08855-1331
Tel: + 1 732 981 0060
Fax: + 1 732 981 9667
http://shop.ieee.org/store/
customer-service@ieee.org

IEEE Computer Society
Asia/Pacific Office
Watanabe Bldg., 1-4-2
Minami-Aoyama
Minato-ku, Tokyo 107-0062
JAPAN
Tel: + 81 3 3408 3118
Fax: + 81 3 3408 3553
tokyo.ofc@computer.org

Individual paper REPRINTS may be ordered at: reprints@computer.org

Editorial production by Bob Werner

Cover art production by Joe Daigle/Studio Productions

Printed in the United States of America by The Printing House

IEEE Computer Society
Conference Publishing Services
http://www.computer.org/proceedings/

Coefficient Bit Reordering Method for Configurable FIR Filtering on Folded Bit-plane Array

Vladimir Ciric, Ivan Milentijevic
Faculty of Electronic Engineering, University of Nis, Serbia and Montenegro
{vciric, milentijevic}@elfak.ni.ac.yu

Abstract

The goal of this paper is development of coefficient bit reordering method for configurable FIR filtering that will enable correct mapping of operations onto functional units of folded bit-plane FIR filtering array, regardless to coefficient number and length. The reordering method is derived in mathematical form and used to synthesize a configurable hardware module that feeds folded array with coefficient bits in proper order. On-the-fly reconfiguration of filtering array is achieved by reconfiguration of hardware module that implements reordering algorithm. Possibilities for throughput increasing by reducing filtering parameters are explored. The derived module is able to handle feeding of folded bit-plane array with different number of coefficients and coefficient length, and it is able to increase the throughput of folded system in cases where filtering with reduced number of taps or coefficient length is performed.

1. Introduction

Cellular-phone technology is changing rapidly. As wireless technologies mature, service providers differentiate themselves by offering new features, such as multimedia capabilities. Providing each feature typically requires a separate chip, or essence, multiple circuitry systems physically joined on a peace of silicon [1]. The additional circuitry adds cost, takes up space, increases power usage in mobile devices, and increase product-design time. This problem can be solved using adaptive approach. With this approach, software can redraw a chip's physical circuitry on the fly, letting a single processor to perform multiple functions [1]-[4]. Circuitry typically can be changed on-the-fly, as software instructions tell circuitry control logic to alternate the computation. Reconfiguration can occur within as little as a few clock cycles [1].

In synthesizing DSP architectures, it is important to minimize the silicon area of integrated circuits, which is achieved by reducing the number of functional units (FUs), 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 FU [3]-[4]. By executing multiple algorithm operations on a single FU, the number of FUs in the implementation is reduced, resulting in integrated circuit with low silicon area [2].

Folded FIR filter is an architecture obtained by application of folding technique onto semi-systolic bit-plane array, which resulted with folded bit-plane array where the number of FUs is not obligatory equal to number of filter coefficients. Thus, the architecture is well prepared to support the operation with changeable number of coefficients and coefficient length [6]. For establishing number of coefficients and coefficient length as user-defined parameters we use the fact that operations mapping onto FUs of folded array depends on filtering parameters. The goal of this paper is development of coefficient bit reordering method for configurable FIR filtering that will enable a correct mapping of operations onto FUs of folded bit-plane FIR filtering array, regardless to filtering parameters. The reordering method will be derived in mathematical form, in accordance to dependencies of operations mapping onto the FUs of folded system. On-the-fly configuration of filtering array will be achieved by configuration of hardware module that implements reordering algorithm. Possibilities for throughput increasing in cases where filtering with reduced number of taps or coefficient length is performed will be explored. With aim to clarify the coefficient bit reordering method we give a brief review of configurable FIR filtering on folded bit-plane array.

2. Configurable FIR Filtering on Folded Bit-plane Array

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

$$y_i = c_0 x_i + c_1 x_{i-1} + \cdots + c_{k-1} x_{i-k+1} \qquad (1)$$

where $c_0, c_1, \ldots, c_{k-1}$ are coefficients while $\{x_i\}$ are input words. Computation (1) can be realized in different manners. When high performances are required systolic arrays are frequently used.

Folding [2]-[4], or time-multiplexing is a technique for efficient resource sharing for area-constrained behavioral synthesis from a data-flow graph (DFG). The basic terms

0-7695-2433-8/05 $20.00 © 2005 IEEE

of folding technique are folding set, folding factor and folding order. Folding set (S) is defined as an ordered set of operations, which contains N operations, time-multiplexed on the same FU. The number of operations executed by the same FU is called folding factor (N). Folding order (v) of operation H_v is a time instance in which FU of folded system executes the operation [2].

The following notation is adopted: k_C – number of coefficients, m_C – coefficient length, c_i^j – bit of coefficient c_i (with weight 2^j), N – folding factor, k – number of folding sets (FUs of folded system), L – total number of operations that has to be performed in order to compute (1), p – position of operation within the DFG ($0 \le p \le L-1$).

The folded bit-plane architecture (FBPA) is semi-systolic architecture (Fig. 1), obtained by application of folding technique onto semi-systolic bit-plane FIR filter architecture (BPA) [6], that consists of k rows of basic cells.

Data flow of configurable folded FIR filter for case k=3, N=4, k_C=2 and m_C=6 is given in Fig. 2. Each row of basic cells in Fig. 1 stands for one FU (i.e. folding set) that performs multiplication of input word with coefficient bit c_i^j, and addition of computed partial product to previously computed sum of partial products. Partial product, computed in FU S_i (i=0,1,…,k-2), enters the FU S_{i+1} with delay of one clock cycle (Fig. 2), together with the input word shifted for one position to the left (multiplication by 2^1), Fig. 1 and Fig. 2. The computation of new output word starts in each N-th time instance in FU S_0 (Fig. 2). Complete output word is available at FU S_{k-1} after $L=k_C \cdot m_C = k \cdot N$ time instances (Fig. 2).

Coefficient bits should enter the array in accordance to data flow from Fig. 2. Ordering of coefficient bits is based on mathematical dependencies that are inherent to the folding dependencies between operations and FUs. Therefore, in order to synthesize the Coefficient Bit Reordering Module (CBRM, Fig. 1), mathematical method for coefficient bit reordering and mapping onto the FUs in correct time instances has to be developed.

3. Coefficient Bit Reordering Method

In order to develop a mathematical method for mapping the coefficient bit c_i^j onto folded set S_s (s=0,1,…,k-1) in time instance r (r=0,1,…,N-1), DFG of unfolded bit-plane FIR filter in transpose form is given in general form in Fig. 3 [5]. Assignment of folding sets ($S_s|r$) for the application of folding technique to the BPA is described as:

$$s = (p-1) \bmod k, \text{ and } r = (p-1) \bmod N, \qquad (2)$$

where s is index of folding set, r represents folding order (Fig. 3). The folding set assignment (2) provides k folding sets (FUs) where each folding set contains N operations.

Assume that operation p ($1 \le p \le L$), shown in Fig. 3, performs multiplication of input data words by coefficient bit c_i^j. According to (2), operation p belongs to folding set (i.e. FU) $s=(p-1) \bmod k$, with folding order $r=(p-1) \bmod N$. In other words, folded architecture multiplies input data word by coefficient c_i^j within folding set S_s ($0 \le s \le k-1$) in time instances $\{N \cdot l + r\}$ ($0 \le r \le N-1$; l=0,1,2,…), Fig. 3. Position of operation that uses coefficient bit c_i^j, can be described as:

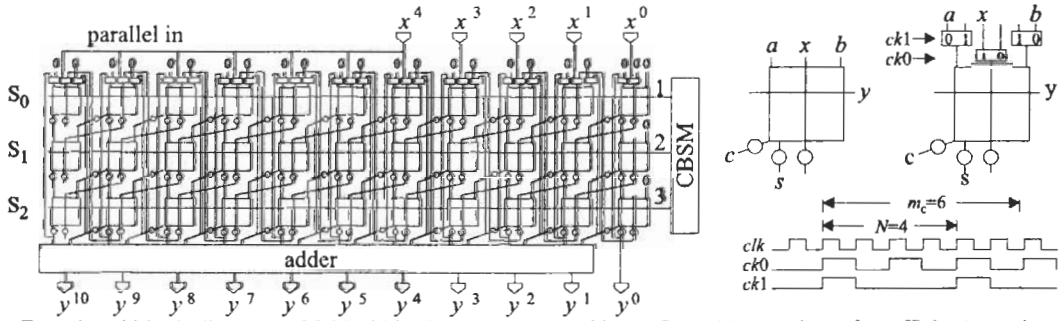

Fig. 1. Functional block diagram of folded bit-plane FIR filter with configurable number of coefficients and coefficient length for k=3 and N=4

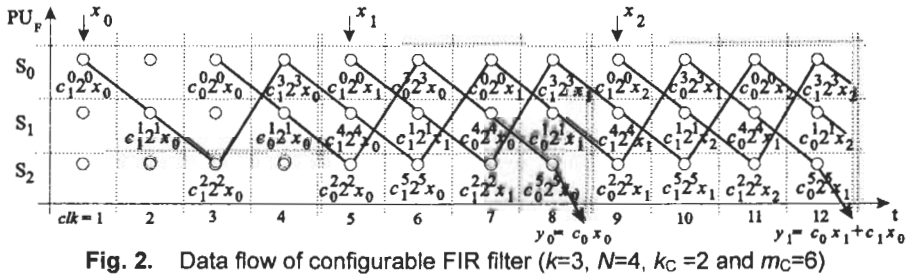

Fig. 2. Data flow of configurable FIR filter (k=3, N=4, k_C =2 and m_C=6)

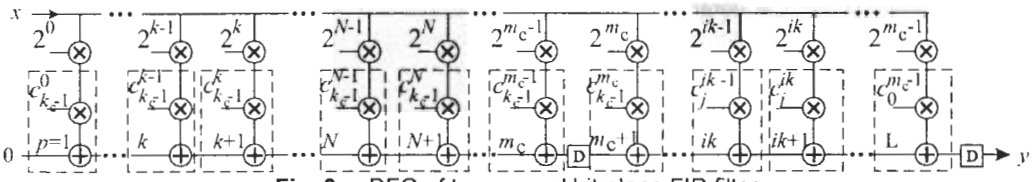

Fig. 3. DFG of transposed bit-plane FIR filter

$$p = m_C\left(k_C - (i+1)\right) + j + 1. \tag{3}$$

The dependency between index of folding set, s ($0 \le s \le k-1$), and folding order r of coefficient bit c_i^j, using (2) and (3), is obtained as:

$$\begin{aligned} s &= \left(m_C \cdot \left(k_C - (i+1)\right) + j + 1\right) \bmod k \\ r &= \left(m_C \cdot \left(k_C - (i+1)\right) + j + 1\right) \bmod N. \end{aligned} \tag{4}$$

Expression (4) identifies the index of folding set (s) that performs multiplication by coefficient c_i^j in time instances $N \cdot l + r$ ($0 \le r \le N-1$; $l = 0,1,2,\ldots$).

Inverse dependencies, denoted as $i = f(s,r)$ and $j = g(s,r)$, can be obtained by mapping the position of operation p to matrix $A_{k \times N}$ in accordance with folding set assignment (2). Each column in matrix $A_{k \times N}$ (Fig. 4a) represents one folding set S_S ($0 \le s \le k-1$) and each row stands for time instances r ($0 \le r \le N-1$).

$$A_{3\times 4} = \begin{vmatrix} 1 & 5 & 9 \\ 10 & 2 & 6 \\ 7 & 11 & 3 \\ 4 & 8 & 12 \end{vmatrix} \qquad A'_{6\times 4} = \begin{vmatrix} 1 & 5 & 9 & \\ & 2 & 6 & 10 \\ & & 3 & 7 \quad 11 \\ & & & 4 \quad 8 \quad 12 \end{vmatrix}$$

a) b)

Fig. 4. Mapping the position of operation p: a) matrix $A_{k \times N}$; b) matrix $A'_{2k \times N}$

With aim to simplify the method, we remove modulo dependencies from (4), and emphasize dependence between operation p and its position within matrix $A_{k \times N}$. By removing modulo dependencies new matrix $A'_{2k \times N}$ is derived (Fig. 4b). The matrix element from Fig. 4b on position (s,r) can be described as:

$$p = \left(\left(s + a \cdot k\right) - r\right) \cdot N + (r+1), \; a = \begin{cases} 0, & s \ge r \\ 1, & s < r \end{cases}. \tag{6}$$

Coefficient bits in DFG from Fig. 3 are assigned to operations according to following expressions:

$$i = k_c - 1 - \left\lfloor \frac{p-1}{m_c} \right\rfloor, \text{ and } j = \left(p - 1\right) \bmod m_c. \tag{7}$$

Using (6) and (7), dependencies $i = f(s,r)$ and $j = g(s,r)$, can be obtained as:

$$\begin{aligned} i &= k_C - 1 - \left(\frac{\left(\left(s + a \cdot k\right) - r\right) \cdot N + r}{m_C} \right), \; a = \begin{cases} 0, & s \ge r \\ 1, & s < r \end{cases}, \\ j &= \left(\left(\left(s + a \cdot k\right) - r\right) \cdot N + r \right) \bmod m_C, \; a = \begin{cases} 0, & s \ge r \\ 1, & s < r \end{cases}. \end{aligned} \tag{8}$$

Equations (4) and (8) can be used to determine the FU that performs operation using specific coefficient bit for user-defined number of coefficient and coefficient length.

According to reordering method, i.e. (4) and (8), CBSM can implement reordering algorithm if it is synthesized as 2-dimensional latch array which size depends on parameters k and N, only. Data paths among the latches should be obtained by mapping of coefficient bit reordering algorithm onto latch array.

4. Mapping of Coefficient Bit Reordering Algorithm onto Architecture

In respect to coefficient bit reordering method, CBSM is synthesized as a two-dimensional array that has two operational modes. First mode is initialization mode, where coefficient bits are reordered according to (4), entering the CBSM in LSB manner. Second mod is run mode, where CBSM feeds the array with coefficient bits. The CBSM feeds the architecture through lines denoted with 1, 2 and 3 in Fig. 1. The internal structure of CBSM is given in Fig. 5 a).

The CBSM is implemented as two-dimensional array of latches where each latch stores one bit of the coefficient (Fig. 5a). The number of rows is equal to number of folding sets (k) while the number of columns is equal to the folding factor (N). Output from each row is feeding one FU of the folded array with coefficient bits. Rows are implemented as shift registers, so during the run mode coefficients rotate through the rows from right to left, feeding each row of folded array with coefficient bits in correct order (solid arrows in Fig. 5a).

Problem of providing the correct bit reordering, during initialization mode, can be solved using (4). The trace of the first coefficient bit $c_{k_C-1}^{\;0}$ can be described in respect to (4) by mapping of time instances $t \in \{0,1,\ldots,k \cdot N-1\}$ onto the array position $[\alpha,\beta]$ as:

$$\alpha = t \bmod k, \text{ and } \beta = t \bmod N.$$

Coefficient bits should be entered in bit serial manner (least significant bit first) starting from coefficient c_{k_C-1} through the latch denoted as [0,0] in Fig. 5 a). The coefficient bit is shifted in next time instance for one row up and one column to the left (dashed arrow in Fig. 5 a), while next bit enters CBSM. Initial state of CBSM for $k=3$, $N=4$, $k_C=2$ and $m_C=6$ is shown in Fig. 5 b).

The number of clock cycles, required to initialize the structure, is $k \cdot N$. After $k \cdot N$ clock cycles, coefficient bits

are loaded in proper order for the computation. In the processing mode coefficients rotate through the rows from right to left.

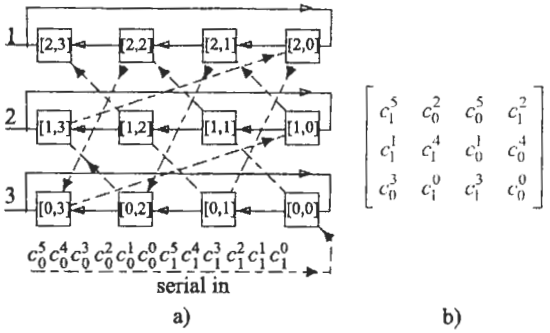

Fig. 5. a) CBSM for k=3, N=4; b) layout of coefficient bits after initialization for k=3, N=4, k_C=2 and m_C=6;

Folded architecture from Fig. 1, fed with coefficient bits by CBRM presented in Fig. 5 a), is able to perform computation with deferent number of coefficients and coefficient length, since array size does not depend on coefficient number and length.

5. Reducible number of operations

Let the coefficient length m_C be reduced to m_C^R, while the number of coefficients k_C, in the implemented architecture, remains constant. Straightforward way to reduce m_C is to replace the most significant bits of coefficients with zeros, but in that case number of computations remains the same.

In order to reduce total number of operations (L) by reducing coefficient number (k_C) or coefficient length (m_C), at least one of parameters k and N has to be changed. The number of rows in CBSM is equal to the number of rows in folded array (k), while the number of columns is equal to folding factor N. The number of rows in folded array from Fig. 1 is equal to the number of folding sets (k). Thus, the number of folding sets cannot be changed in runtime in a straightforward manner. Number of columns in CBSM can be reduced using shift registers with changeable length, as it is shown in Fig. 6.

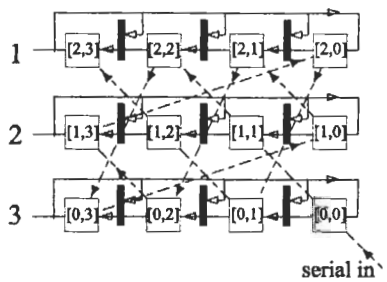

Fig. 6. CBSM with changeable length for k=3, N=4

The architecture can achieve higher throughput in cases where filtering with reduced number of taps or coefficient length is performed.

6. Concluding Remarks

We considered the coefficient bit reordering method for configurable FIR filtering on folded bit-plane array. In order to enable dynamic operation assignment on folded bit-plane array, we developed mathematical method for coefficient bit reordering. Method for operations mapping onto the different hardware units in the processing array was successfully developed by using dependencies inherent to folding set assignment. Thus, the synthesis of coefficient bit supply module for folded FIR filtering array was carried out. The proposed module supports on-the-fly configuration of number of taps and coefficient length. The architecture provides flexible FIR filtering computations. Instead of extending coefficients to full coefficient length in cases where operation with reduced number of taps or coefficient length is required, the proposed CBSM is able to increase the throughput reducing the folding factor.

7. References

[1] L. Paulson, L. Garber, "Reconfiguring Wireless Phones with Adaptive Chips", *IEEE Computer*, Vol. 36, no. 9, pp. 9-11, 2003.

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

[3] K.K.Parhi, C.Y.Wang, and A.P.Brown, "Synthesis of Control Circuits in Folded Pipelined DSP Architectures", *IEEE Journal of Solid State Circuits*, vol. 27, no. 1, pp. 29-43, 1992.

[4] T.C.Denk and K.K.Parhi, "Synthesis of Folded Pipelined Architectures for Multirate DSP Algorithms", *IEEE Transactions on VLSI Systems*, vol. 6, pp. 595-607, December 1998.

[5] I. Milentijevic, V. Ciric, O. Vojinovic and T. Tokic: "Folded Semi-Systolic FIR Filter Architecture with Changeable Folding Factor". *Neural, Parallel & Scientific Computations*, Vol. 10, No 2, Dynamic Publishers Atlanta, USA, June 2002, pp. 235-248.

[6] V.Ciric, I.Milentijevic, "Configurable Folded Bit-Plane Architecture for FIR Filtering", *Proceedings of the WPS-DSD 2003, 29th EUROMICRO Conference*, Belek, Turkey, pp. 23-24, 2003.

Published by the IEEE Computer Society
10662 Los Vaqueros Circle
P.O. Box 3014
Los Alamitos, CA 90720-1314

IEEE Computer Society Order Number P2433
Library of Congress Number 2005930138
ISBN 0-7695-2433-8