Configurable Folded Bit-Plane Architecture for FIR Filtering

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

1. Introduction

The application of folding technique onto array type architectures for FIR filtering gives designers greater flexibility in finding the best trade-off between hardware size and throughput rate [1,2]. The Bit-Plane Architecture (BPA) as a highly regular architecture [3], 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 [3], is a good candidate for application of folding technique. Since the straight-forward application of folding technique onto the BPA has not been possible, the transformation of the BPA data flow graph (DFG) that enables the synthesis of the folded FIR filter architecture has been proposed in [4]. However, the problem of the designing of folded FIR filter with changeable number of coefficients still remains unsolved. In this paper we propose configurable folded bit-plane architecture for FIR filtering that allows programming of both number of taps and coefficient length. As a starting point we use the transformed DFG for the BPA proposed in [4] and involve new assignment of folding sets. This paper deals with the mapping of the transformed DFG, for the BPA, onto the configurable folded system.

2. Notation and Preparation for Folding

The following notation provides the basis for further explanation of the mapping of DFG for BPA onto the configurable system: m_c – coefficient length; k_c – number of coefficients; c_i^j – bit of coefficient c_i (with weight 2^j); N – folding factor; k – number of folding sets; n – length of input words x_i; L – total number of "operations" in the DFG, where one operation assumes forming of partial product and the addition performed on one "row" of basic cells (basic cell contains AND gate and full adder); p – position of *operation* within the DFG ($0 \le P \le L-1$); S_s – *s-th* folding set ($0 \le s \le K-1$).

Starting DFG, which is well prepared for application of folding technique, is given in [4]. Mathematical description of folding sets assignment ($S_s|r$), on DFG from [4], is done according to the following equations

$$s = p \bmod k$$
$$r = p \bmod N. \quad (1)$$

The idea of mapping different operations onto the different hardware units according to the chosen number of coefficients and coefficient length in fixed array structure is introduced with (1). The proposed mapping of operations enables both changing the number of coefficients and coefficient length including the following constraint: $L = k_c m_c = kN$.

3. Folded Bit-Plane Architecture

Using a new assignment of folding sets, which is applied on the transformed DFG from [4], we obtain folded Bit-Plane architecture in general form (Fig. 1). Input Data Entering Module (IDEM), denoted with dashed lines in Fig. 1, provides input data for the folded architecture in accordance with the allocation table (Table 1).

Sections $S_0, S_1, \ldots, S_{k-1}$ in Fig. 1. are Processing Elements (PE) of the folded architecture. Each section is devoted to computations from the corresponding folding set. Sections are implemented as rows of basic cells, where the basic cell is comprised of AND gate and full adder.

4. Functional Description

Folding sets $S_0, S_1, \ldots, S_{k-1}$ are shown in dashed boxes (Fig. 1). Each folding set contains N operations. In order to clarify the folded FIR filter method of operation, the hardware section that performs the operation from set S_S ($s=0,1,\ldots,k-1$) is also denoted with S_S. Initially, the computation starts in folding set S_0 where the product $2^0 \cdot c_{kc-1}^{\ 0} \cdot x_0$ is obtained in the first clock cycle. In the next clock cycle folding set S_1 generates the partial product $2^1 \cdot c_{kc-1}^{\ 1} \cdot x_0$ adding previously computed partial product from S_0. Thus, the value $(2^0 \cdot c_{kc-1}^{\ 0} \cdot x_0) + (2^1 \cdot c_{kc-1}^{\ 1} \cdot x_0)$ is entered into the next section, which performs the operations from S_2, in the third clock cycle, etc.

Table 1. Allocation table

	input	D_0	D_1		D_{k-1}
0	x_0				
1	x_0	x_0			
2	x_0		x_0		
⋮				…	
k-1	x_0				x_0
k	x_0	x_0			
⋮				…	
N	x_1				
N+1	x_1	x_1			
N+2	x_1		x_1		
⋮					
iN	x_i				
⋮		x_i			
m_c	x_i				
m_c+1	x_i	x_i			
⋮				…	
(i+1)N	x_i				
⋮					
(k-1)N	$x_{(k_c-1)m_c/N}$				
(k-1)N+1	$x_{(k_c-1)m_c/N}$	$x_{(k_c-1)m_c/N}$			
(k-1)N+2	$x_{(k_c-1)m_c/N}$	$x_{(k_c-1)m_c/N}$			
⋮				…	
*k*N-1*	x_{k-1}				

Let us note that in the simultaneous process the architecture starts with computation of all results y_{kc-2}, y_{kc-1}, ..., y_0 in reverse order. The first completely generated result at output is y_0 with latency of $2m_c$-N clock cycles. New result y is generated every N clock cycles. The data flow through the folded architecture for case $k=3$, $N=4$, $k_c=2$ and $m_c=6$ is given in Fig. 2. The proposed architecture supports the operation with changeable number of coefficients and coefficient length.

The mechanism for throughput increasing, described in [4], can be easily exploited on this architecture.

5. Discussion and Conclusions

The proposed folding set assignment on transformed FIR filter DFG enables the successful synthesis of configurable folded FIR filter architecture. The derived folded processing array can be configured to perform FIR filtering with different number of taps and length of coefficients. Synthesized architecture has kept desirable features of source architecture such as extensive pipelining, high regularity, truncation of LSBs of intermediate results without any loss of accuracy. The number of basic cells is reduced to the number of basic cells in one plane of source architecture. The obtained folded semi-systolic architecture is presented by DFG, allocation table, and data flow diagram.

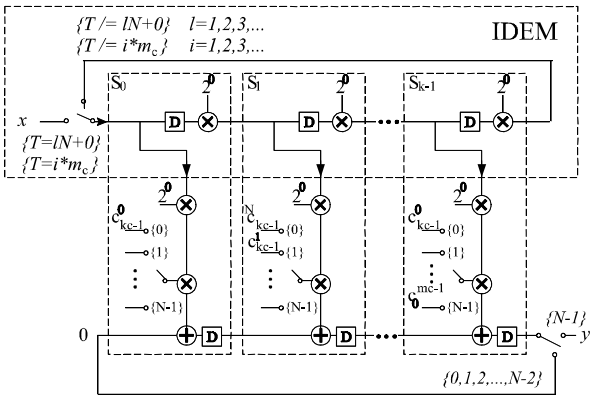

Figure 1. Folded FIR filter architecture with changeable number of coefficients and coefficient length

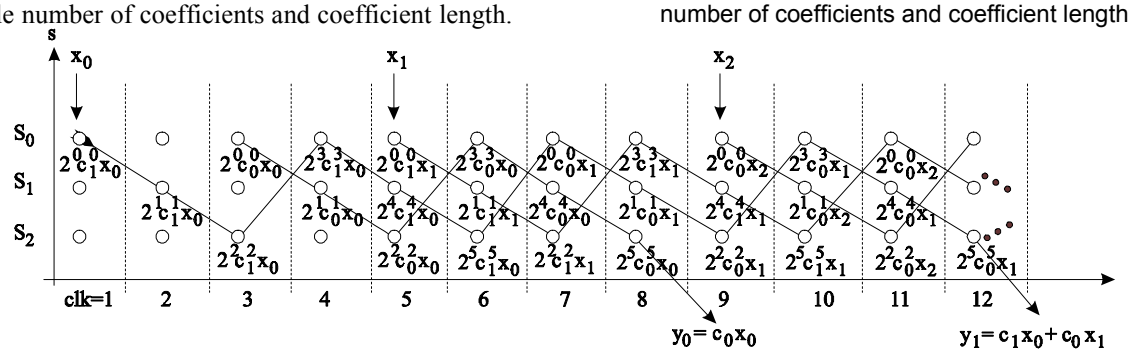

Figure 2. Data flow for folded architecture (k=3, N=4, k_c =2 i m_c=6)

6. References

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

[2] 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.

[3] 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.

[4] I. Milentijevic, V. Ciric, O. Vojinovic, T. Tokic, "Folded Semi-Systolic FIR Filter Architecture With Changeable Folding Factor", *Neural, Parallel & Scientific Computations, Dynamic Publishers,* Atlanta, Vol. 10, No 2, 2002, pp. 235-247.