International Journal for Modern Trends in Science and Technology
Volume: 03, Issue No: 04, April 2017
ISSN: 2455-3778
http://www.ijmtst.com

Design of Efficient Feed Forward Filter for Fixed and Reconfigurable Applications

V.Akshaya[1] | S.G.Ilakkiya[2] | D.Indhumathi[3] | L.Barathi[4]

[1-4]Department of ECE, K.Ramakrishna College of Engineering, Samayapuram, Trichy,Tamilnadu, India.

To Cite this Article
V.Akshaya, S.G.Ilakkiya, D.Indhumathi and L.Barathi, "Design of Efficient Feed Forward Filter for Fixed and Reconfigurable Applications", *International Journal for Modern Trends in Science and Technology*, Vol. 03, Issue 04, April 2017, pp. 150-153.

ABSTRACT

The power consumption and speed are the two main challenging factors in Very Large Scale Integrated Circuit (VLSI) design techniques. The computation saving is one of the way to obtain the optimized power consumption and speed. The design of Finite Impulse Response (FIR) filter using transpose form structure is naturally pipelined and upholds multiple constant multiplication (MCM) technique. This MCM technique results in large computation saving. But, the transpose form configurations does not support the block processing. In the existing method, the possibility of realization of FIR filter in transpose form configuration to achieve efficient area and delay for large order FIR filters were explored. In the FIR filter structure , the ripple carry adder is used to add the partial inner products. The ripple carry adder provides efficient area utilization but its operating speed is slow. In this proposed method, Baugh- Wooley multiplier is used to increase the speed and also to reduce the power consumption. The proposed structure significantly reduces the area delay product (ADP) and energy per sample (EPS) than the existing FIR structure.

KEYWORDS: *Transpose form, Block processing, MCM, Reconfigurable, FIR*

I. INTRODUCTION

Finite-Impulse response (FIR) filter is always used in several digital signal processing applications like speech processing, echo cancellation, and various communication applications, including software-defined radio (SDR).Finite Impulse Response (FIR)filters are digital filters that have Finite-Impulse response (FIR) filter is always used in several digital signal processing applications like speech processing, echo cancellation, and various communication applications, including software-defined radio (SDR).Finite Impulse Response (FIR)filters are digital filters that have a finite impulse response. FIR filters operate on present and past input values and are the simplest filters to design. FIR filters (having fixed coefficients)using distributed arithmetic (DA)- based designs use lookup tables (LUTs) to store the pre-computed results which reduce the computational complexity. Reconfiguration time is more in the DA based designs. canonic sign digit (CSD)-based RFIR filter, where the nonzero CSD values are used. These values are modified to reduce the precision of filter coefficients without impact on filter behavior . Constant shift method requires redundant adders but consumes more area and power. There are some applications, like SDR channelizer, where FIR filters need be implemented in a reconfigurable hardware to support wireless communication. In this paper, realization of block FIR filter in transpose form configuration is done in order to take advantage of the MCM method and pipelining for area-delay efficient realization of large order FIR

filters which is used for both fixed and reconfigurable applications.

II. System Overview

Finite Impulse Response(FIR) filter had several applications. In some applications the coefficients of FIR filters remain fixed , while in some other applications, it requires separate FIR filters of different specifications to extract one of the desired narrow band channels from the wideband RF front end. These FIR filters need to be implemented in Reconfigurable Finite Impulse Response (RFIR) structure to support multi-standard wireless communication. In the proposed system, transpose form block filter for reconfigurable applications is designed. A low-complexity design using the Multiple Constant Multiplication (MCM) scheme is presented for the block implementation of fixed Finite Impulse Response (FIR) filters. The proposed system is designed for both fixed and reconfigurable applications of Finite Impulse Response (FIR) filters.

A. Reconfigurable FIR Filter

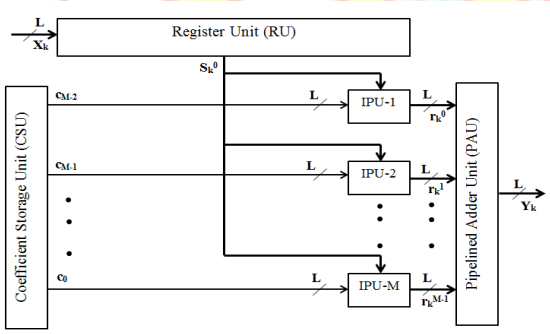

Fig. 1 Block Diagram of FIR Filter For Reconfigurable Applications

The proposed structure for block FIR filter for reconfigurable applications is shown in figure. The block diagram for FIR filter consists of one Coefficient Selection Unit(CSU), one Register Unit(RU), M number of Inner Product Unit(IPU) and one Pipeline Adder Unit(PAU). The Coefficient Storage Unit stores coefficients of all the filters to be used for reconfigurable applications. The CSU is implemented using N ROM LUTs of P words each, such that filter coefficients of any particular channel filter are obtained in one clock cycle. N is the filter length.

The RU has (L-1) registers of B-bit width. It receives x_k during the k^{th} cycle and produces L rows of S_k^0 in parallel. L rows of input are transmitted to M IPUs. The IPU receive M short weight vectors from the CSU such that during the cycle, the $(m + 1)^{th}$ IPU receives the weight vector c_{M-m-1} from the CSU and L rows of S_k^0 from the RU. Each IPU performs matrix-vector product of S_k^0 with the short-weight vector c_m, and computes a block of L partial filter outputs r_k^m. Therefore, each IPU performs L inner-product computations of L rows of S_k^0 with a common weight vector c_m.

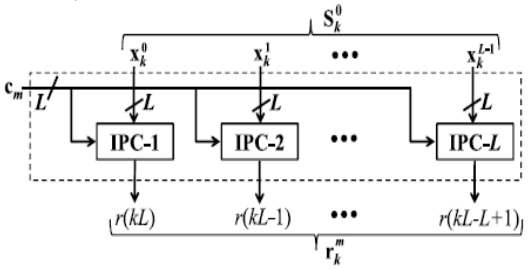

Fig. 2 Internal Structure of IPU

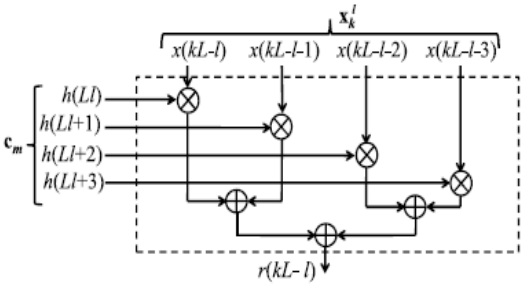

Fig. 3 Internal Structure of IPC

The Inner Product Unit consists of L number of L-point inner-product cells (IPCs). Each IP cell involves multipliers and adders. The $(l+1)^{th}$ IPC receives the $(l+1)^{th}$ row of S_k^0 and the coefficient vector c_m, and computes a partial result of inner product r(kL-l), for $0 \le l \le L-1$. All the M IPUs work in parallel and produce M blocks of result r_k^m . The partial inner products are added in the PAU to obtain a block of L filter outputs. In each cycle, the proposed structure receives a block of L inputs and produces a block of L filter outputs.

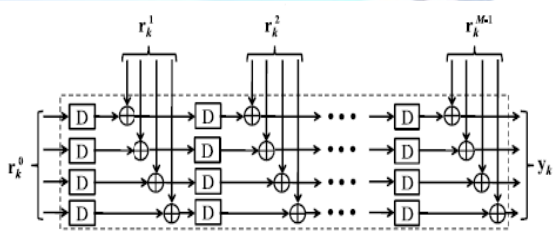

Fig. 4 Internal Structure of PAU

B. Fixed coefficient FIR filter

Design of an efficient architecture for fixed point FIR filter has been considered as the major research. In any FIR filter, the multiplier is the major constraint which defines the performance of the desired filter. The fixed coefficient filter is implemented using Multiple Constant Multiplication (MCM) method.

C. *Multiple Constant Multiplication (MCM) Method*

The multiplication operation which is performed between one particular variable (the input) and many constants (the coefficients) is known as the multiple constant multiplication (MCM). MCM method is more effective since it reduces the number of additions required for the realization of FIR filters.

The algorithms proposed to implement the MCM for an efficient FIR filter design are graph based algorithms and common sub-expression elimination (CSE) algorithms. MCM has been on more effective common sub expression elimination, the optimization of adder-trees, which sum up the computed sub-expressions for each coefficient, is largely omitted. MCM blocks can be formed only in the transpose form configuration of FIR filters.

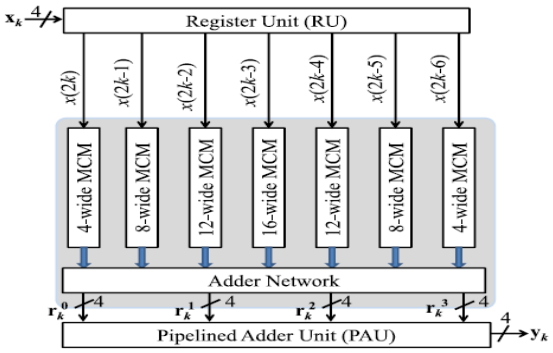

Fig. 5 Block Diagram for Fixed Block FIR Filter

For fixed-coefficient implementation, the CSU and IPUs are not required. The multiplications are required to be mapped to the MCM units for a low-complexity realization. MCM-based implementation of block FIR filter makes use of the symmetry in input matrix S_k^0 to perform horizontal and vertical common subexpression elimination and to minimize the number of shift-add operations in the MCM blocks.

For fixed coefficient FIR filter,

$$Y(z) = z^{-1} \dots z^{-1}(z^{-1}r_{M-1} + r_{M-2} + r_{M-3}) + \cdots + r_1 + r_0 \quad \text{---------- (2.2)}$$

$$R = S_k^0 . C \quad \text{---------- (2.3)}$$

Where R and C are defined as

$$R = [r_0^T \;\; r_1^T \;\; \dots\dots \;\; r_{M-1}^T] \quad \text{---------- (2.4)}$$

$$C = [c_0^T \;\; c_1^T \;\; \dots\dots \;\; c_{M-1}^T] \quad \text{---------- (2.5)}$$

MCM can be applied in both horizontal and vertical direction of the coefficient matrix. For larger values of N or the smaller block sizes, the row size of the coefficient matrix is larger that results in larger MCM size across all the samples, which results into larger saving in computational complexity. The MCM-based structure involves six MCM blocks corresponding to six input samples. Each MCM block produces the necessary product terms. The subexpressions of the MCM blocks are shift added in the adder network to produce the inner-product values (r_l, m) for $0 \le l \le L-1$ and $0 \le m \le (N/L)-1$. The inner-product values are finally added in the Pipelined Adder Unit (PAU) to obtain the feed-forward filter output.

III. Simulation Results

```
Power summary:                          I(mA)     P(mW)
------------------------------------------------------------------
Total estimated power consumption:                   34
                                        ---
                          Vccint 1.20V:       8       10
                          Vccaux 2.50V:       8       20
                          Vcco25 2.50V:       2        4
                                        ---
                            Clocks:           0        0
                            Inputs:           0        0
                             Logic:           0        0
                           Outputs:
                              Vcco25          0        0
                           Signals:           0        0
                                        ---
               Quiescent Vccint  1.20V:       8       10
               Quiescent Vccaux  2.50V:       8       20
               Quiescent Vcco25  2.50V:       2        4
```

Fig. 6 Power Output

CLOCK PERIOD: 24.205NS (FREQUENCY: 41.314MHZ)

IV. Conclusions

In this paper, we have explored the possibility of realization of block FIR filters in transpose form configuration for efficient realization of both fixed and reconfigurable applications. We have presented a scheme to identify the MCM blocks for horizontal and vertical subexpression elimination in the proposed block FIR filter for fixed coefficients to reduce the computational complexity. Performance comparison shows that the proposed structure involves significantly less ADP and less EPS than the existing block direct-form structure for medium or large filter lengths while for the short-length filters, the existing block direct-form structure has less ADP and less EPS than the proposed structure.

ACKNOWLEDGMENT

We thank the almighty god, without whom it would not have been possible for us to complete our project. We wish to address my profound gratitude to our Chairman DR.K.Ramakrishnan, our Executive Director DR.S.Kuppusamy, our principal DR.D.Srnivasan, MR.T.Muruganatham, Head of the Electronics and Communication

Engineering Department, our guide Mrs.S.Satiya, Assistant Professor, Electronics and Communication Engineering Department, K. Ramakrishnan college of Engineering.

REFERENCES

[1] Basant Kumar Mohanty, and Pramod Kumar Meher (2015,) 'A High-Performance FIR Filter Architecture for Fixed and Reconfigurable Applications', *IEEE Transactions on VLSI System.*

[2] Mahesh R. and Vinod A. P. (2010), 'New reconfigurable architectures for implementing FIR filters with low complexity', *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst., vol. 29, no. 2, pp. 275–288.*

[3] Meher P.K. (2010), 'New approach to look-up-table design and memory based realization of FIR digital filter', *IEEE Trans. Circuits Syst. I, Reg. Papers, vol. 57, no. 3, pp. 592–603.*

[4] Mohanty B. K. and Meher P. K. (2013), 'A high-performance energy-efficient architecture for FIR adaptive filter based on new distributed arithmetic formulation of block LMS algorithm,' *IEEE Trans. Signal Process., vol. 61, no. 4, pp. 921–932.*

[5] Park S.Y. and Meher P.K. (2014) , 'Efficient FPGA and ASIC realizations of a DA-based reconfigurable FIR digital filter', *IEEE Trans. Circuits Syst. II, Exp. Briefs, vol. 61, no. 7, pp. 511–515.*

[6] Pramod Kumar Meher, ShrutisagarChandrasekaran and Abbes Amira (2008), 'FPGA Realization of FIR Filters by Efficient and Flexible Systolization Using Distributed Arithmetic', *IEEE Transactions On Signal Processing, Vol. 56, No. 7.*

[7] Pramod Kumar Meher (2006), 'Hardware-Efficient Systolization of DA-Based Calculation of Finite Digital Convolution', *IEEE Transactions On Circuits And Systems-II: Express Briefs, Vol. 53, No. 8.*

[8] Vinod A. P. and Lai E. M. (2006), 'Low power and high-speed implementation of FIR filters for software defined radio receivers', *IEEE Trans. Wireless Commun., vol. 7, no. 5, pp. 1669–1675.*