

Low Power Implementation and Analysis of Digital Fir Filter Based Low Power Multiplexer Base Shift/Add Multiplier



Engineering

KEYWORDS : Storage Register, Filter coefficient, FIR filter.

D.Hari Priya

Associate Professor & Saveetha School of Engineering, Saveetha University, Chennai.

ABSTRACT

Conventional FIR filters consist of cells equal in number to the length of the filter i.e. the number of data taps. Each cell consists of a storage register, a second register and a multiplier. The storage register stores the data tap values, which are digital samples of the signal being processed by the filter. The second register stores the filter coefficients for a particular tap and the multiplier generates the product of the two register contents. The latter product serves as the output of the cell, and the weighted sum that constitutes the FIR filter output is generated by adding the outputs of all of the cells. FIR filters are said to be finite because they do not have any feedback. FIR digital filters are widely used in DSP by the virtue of its stability, linear phase, fewer finite precision error and efficient implementation.

INTRODUCTION

The FPGA industry sprouted from programmable read-only memory (PROM) and programmable logic devices (PLDs). PROMs and PLDs both had the option of being programmed in batches in a factory or in the field (field programmable). However programmable logic was hard-wired between logic gates.

In the late 1980s the Naval Surface Warfare Department funded an experiment proposed by Steve Cassel man to develop a computer that would implement 600,000 reprogrammable gates. Cassel man was successful and a patent related to the system was issued in 1992.

Some of the industry's foundational concepts and technologies for programmable logic arrays, gates, and logic blocks are founded in patents awarded to David W. Page and Laverne R. Peterson in 1985.

Xilinx co-founders Ross Freeman and Bernard Vander Schmitt invented the first commercially viable field programmable gate array in 1985 – the XC2064. The XC2064 had programmable gates and programmable interconnects between gates, the beginnings of a new technology and market. The XC2064 boasted a mere 64 configurable logic blocks (CLBs), with two 3-input lookup tables (LUTs). More than 20 years later, Freeman was entered into the National Inventors Hall of Fame for his invention.

1.1 FIR Filter Applications

FIR Digital filters find extensive applications in mobile communication systems for applications such as channelization, channel equalization, matched filtering, and pulse shaping, due to their absolute stability and linear phase properties. The filters employed in mobile systems must be realized to consume less power and operate at high speed.

1.2 Importance of Re configurability

Recently, with the advent of software defined radio (SDR) technology, finite impulse response (FIR) filter research has been focused on reconfigurable realizations. The fundamental idea of an SDR is to replace most of the analog signal processing in the transceivers with digital signal processing in order to provide the advantage of flexibility through reconfiguration. This will enable different air-interfaces to be implemented on a single generic hardware platform to support multi-standard wireless communications. Wideband receivers in SDR must be realized to meet the stringent specifications of low power consumption and high speed.

Reconfigurability of the receiver to work with different wireless communication standards is another key requirement in an SDR. The most computationally intensive part of an SDR

receiver is the channelize since it operates at the highest sampling rate. It extracts multiple narrowband channels from a wideband signal using a bank of FIR filters, called channel filters. Using polyphone filter structure, decimation can be done prior to channel filtering so that the channel filters need to operate only at relatively low sampling rates. This can relax the speed of operation of the filters to a good extent.

1.3 Significance of Higher order filters

However due to the stringent adjacent channel attenuation specifications of wireless communication standards, higher order filters are required for channelization and consequently the complexity and power consumption of the receiver will be high. As the ultimate aim of the future multi-standard wireless communication receiver is to realize its functionalities in mobile handsets, where its full utilization is possible, low power and low area implementation of FIR channel filters is inevitable. In, the filter multiplications are done via state machines in an iterative shift and add component and as a result of this there is huge savings in area. For lower order filters, the approach in offers good trade-off between speed and area. But in general, the channel filters in wireless communication receivers need to be of high order to achieve sharp transition band and low adjacent channel attenuation requirements. For such applications, the approach in results in low speed of operation.

1.4 Multipliers

The complexity of FIR filters is dominated by the complexity of coefficient multipliers. It is well known that the common sub-expression elimination (CSE) methods based on canonical signed digit (CSD) coefficients produce low complexity FIR filter coefficient multipliers. The goal of CSE is to identify multiple occurrences of identical bit patterns that are present in the CSD representation of coefficients, and eliminate these redundant multiplications.

A modification of the 2-bit CSE technique in for identifying the proper patterns for elimination of redundant computations and to maximize the optimization impact was proposed in. In, the technique in was modified to minimize the logic depth (LD) (LD is defined as the number of adder-steps in a maximal path of decomposed multiplications) and thus to improve the speed of operation.

1.5 Binary Common Sub-expression Elimination(BCSE)

We have proposed the binary common sub-expression elimination (BCSE) method which provided improved adder reductions and thus low complexity FIR filters compared to. In, a method based on the pseudo floating point method was used to encode the filter coefficients and thus to reduce the complexity of the filter. But the method in is limited to filter lengths less than 40. In general, the methods in are only suit-

able for application specific filters where the coefficients are fixed and hence not suitable for reconfigurable filters.

1.6.1 Canonical Signed Digit

In, a CSD based digit reconfigurable FIR filter architecture was proposed. This architecture was independent of the number of taps because the number of taps and non-zero digits in each tap were arbitrarily assigned. The architecture in demanded huge hardware resources and this makes the method infeasible for power constrained SDR receiver applications. In, a high-speed programmable CSD based FIR filter was proposed.

The filter architecture consisted of a programmable CSD based Booth encoding scheme and partial product Wallace adder tree. The final adder was a carry look-ahead adder. Though this method offered a high speed solution, the resulting filters consume more power. Another high-speed programmable FIR based on polyphone decomposition was proposed. In, the concept of reconfigurable multiplier block (ReMB) was introduced. The ReMB will generate all the coefficient products and a multiplexer will select the required ones depending on the input. It was shown that by pushing the multiplexer deep into the multiplier block design, the redundancy can be reduced. The resulting specialized multiplier design can be more efficient in terms of area and computational complexity compared to the general-purpose multiplier plus the coefficient store.

But the ReMB proposed, has its area, power, and speed dependent on the filter-length making them inappropriate for higher order FIR filters. In, a multiplexed multiple constant multiplications (MMCM) approach was proposed.

This method considers the coefficient set as a constant and uses the graph dependence (GD) algorithms for reducing redundant. But this method follows a directed acyclic graph structure which will result in long LD and thus lower speed of operation. Also the area of the architecture linearly increases with the filter length and filters with filter-length above 40 are infeasible.

1.7 Vector Scaling Operation

In, the common digital signal processing (DSP) operations such as filtering and matrix multiplication were identified and expressed as vector scaling operations. In order to apply vector scaling, simple number decomposition strategies were identified. The idea was to recompute the values such as x , $3x$, $5x$, $7x$, $9x$, $11x$, $13x$, and $15x$, where x is the input signal and then reuse these precipitations efficiently using multiplexers.

The presence of multiplexers gave the option of adaptive computing. In that method it was modified and efficient circuit-level techniques, namely a new carry-select adder and conditional capture flip-flop, were used to further improve power and performance. The architectures are appropriate only for relatively lower order filters and hence not suitable for channel filters in communication receivers.

Although a few works addressed the problem of reducing the complexity of coefficient multipliers in reconfigurable FIR filters, hardly any work demonstrated reconfigurability in higher order filters. Complexity reduction in reconfigurable filters especially for wireless communication applications where higher order filters are often required to meet the stringent adjacent channel attenuation specifications.

2.3 Multiplexer Unit

The multiplexer units are used to select the appropriate

output from the shift and add unit. All the multiplexers will share the outputs of the shift and add unit. The inputs to the multiplexers are the 8/4 inputs from the shift and add unit and hence 8:1/4:1 multiplexer units are employed in the architecture. The select signals of the multiplexers are the filter coefficients which are previously stored in a look up table (LUT). The CSM and PSM architectures basically differ in the LUTs without any modification whereas in PSM, the coefficients are stored in a coded format. The number of multiplexers will also be different for PSM and CSM.

In CSM, the number of multiplexers will be dependent on the number of groups after the partitioning of the filter coefficient into fixed groups. The number of multiplexers in the PSM is dependent on the number of non-zero operands in the coefficient for the worst case after the application of BCSE algorithm.

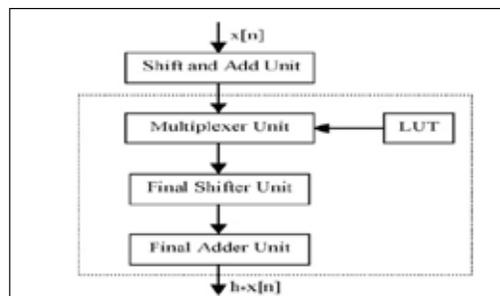


Fig2.Architecture of the proposed method

2.4 Final Shifter Unit

The final shifter unit will perform the shifting operation after all the intermediate additions(i.e., intra-coefficient additions) are done. This can be illustrated using the output expression $y = 2^{-4}x + 2^{-6}x + 2^{-15}x + 2^{-16}x \cdot (x + 2^{-1}x)$ from the shift and add units with the help of multiplexer unit, the final shifter unit will perform the shift operations 2^{-4} and 2^{-15} in (2). The PSM and CSM architectures also differ in the nature of final shifters. In the CSM, the final shifts are constants and hence no PS are required. In the PSM, we have used PS.

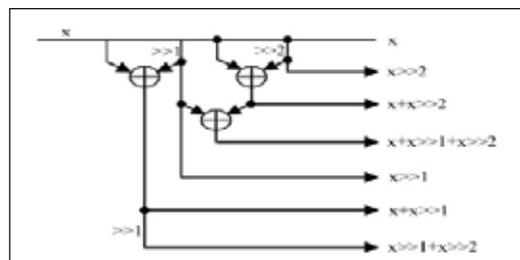


Fig.3. Architecture of Shift and Add Unit

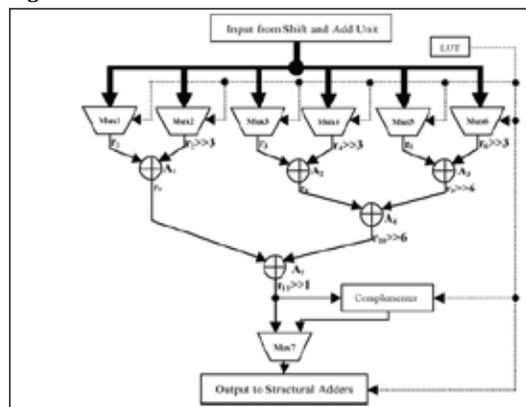


Fig .4.Architecture of PE for CSM

Final adder unit will compute the sum of all the intermediate sums to obtain $h * x[n]$. The architecture of PE for CSM is shown in Fig. 4. The coefficient word length is considered as 16 bits. The filter coefficients are stored in the LUT in sign-magnitude form with the MSB reserved for the sign bit. The first bit after the signbit is used to represent the integer part of the coefficient and the remaining 16 bits are used to represent the fractional part of the coefficient. Thus, each 16-bit coefficient is stored as a 18-bit value in LUTs.

REFERENCE

- [1] "The Evolution of Forth" by Elizabeth D. Rather et al. | [2]"Interfacing a hardware multiplier to a general-purpose microprocessor" | [3] Computer Architecture: A quantitative Approach, Hennessy and Patterson, 1990, Morgan Kaufmann Publishers, Inc. Section A.2 (pages A-3 through A-6) and section A.9 (pages A-39 through A-49). | [4]C.R. Baugh and B.A. Wooley, "A 2's complement parallel array multiplication algorithm," IEEE Transaction on Computers, C-22, pp. 1045-1047, December 1973. | [5]F. Gebali, "Baugh-Wooley Multiplier", University of Victoria, CENG 465 Lab 2, circa 2003 | [6] Wisniewski, Remigiusz (2009). Synthesis of compositional microprogram control units for programmable devices.ZielonaGóra: University of ZielonaGóra. | [7] FPGA Architecture for the Challenge | [8] FPGA Signal Integrity tutorial | [9] NASA: FPGA drive strength | [10] Mike Thompson. "Mixed-signal FPGAs provide GREEN POWER". EE Times, 2007-07-02. | [11]Google Patent Search, "Re-programmable PLA". Retrieved February 5, 2009. | [12] Google Patent Search, "Dynamic data re-programmable PLA". Retrieved February 5, 2009. |