FIR Filter Design with Re-Configurable Constant Multiplication Circuit Optimized using OSR Method

Kollara Ragi Rajagopalan[1] Bency Varghese A[2]
[1]Student [2]Assistant Professor
[1,2]Department of Electronics and Communication Engineering
[1,2]IES College of Engineering, Thrissur, India

Abstract— FIR filters are used to implement almost any type of digital frequency response and are widely used in digital communications, software-defined radios, adaptive filters, multi-rate signal processing etc. To optimize these filters, different multiplication techniques or optimization techniques for these multipliers has been approached as it is the basic operation in different areas of the digital computations like in signal processing task, high speed integrated circuits and application-specific ICs. Optimizing the multipliers helps decrease the device parameters and increase the performance. So the necessity of high speed and low power multipliers has increased. This paper introduces the application of a computationally efficient Optimal Shift Reassignment method to optimize the hardware architecture for Re-configurable Constant Multiplication (RCM) block to replace the multiplier of an FIR filter. Optimal shift reassignment (OSR) is used in RCM circuits to reduce the number of required multiplexers in the circuit. Using the OSR method, the shift values of different inputs of a multiplexer can be realigned. In this project a RCM block is designed and then optimized using OSR method for the multiplication circuitry in FIR filter. The design is done using the tool Xilinx13.2. Filters are dynamically reconfigured by modifying the filter's coefficients. Coding is done using VHDL and simulation is verified using Xilinx ISE Design Suite.

Keywords: FIR filters, VHDL, VLSI, RCM, Multiplier optimization, Constant multiplier circuit, Image processing

I. INTRODUCTION

Modern-day systems like software-defined radios (SDR), high-efficiency video/data codec, and high rate communication require a re-configurable finite impulse response (FIR) filter hardware where the filter's coefficient can be re-configured dynamically. For various DSP applications such as in loud speaker equalization devices, in echo cancellation tools, in adaptive noise cancellation tools, speech processing techniques, and many communications applications, including software-defined radio devices, we use digital FIR filters. A Finite Impulse Response (FIR) filter has an impulse response of finite duration because it settles to zero in finite time. For high speed in digital communication, filters need a high rate of sampling. FIR filters are the most important digital signal processing elements that comes packed with dedicated hardware instead of software to improve the speed. In a re-configurable FIR filter, the reconfigurable multiple constant multiplication (RMCM) blocks are the critical elements that define the desired filter implementation's overall performance. Hence, the development of a low-complexity RMCM block for re-configurable FIR filter hardware is highly required.

The re-configurable filter structures are implemented to obtain a low power, highspeed implementation of variable size partitioning FIR filter. With an increase in the order of the filter, the number of additions and multiplications required, also increases. As redundant computation is not available in the FIR filter algorithm, its real-time implementation in an environment is resource-restricted and is a demanding job. Usually the coefficients of filters are constant, and they can be used to reduce multiplication complexity.

Many structures of FIR filter using MCM algorithm have been proposed to reduce the area of implementation. MCM is one such area-efficient algorithm, but the designing of Finite Impulse Response filter requires a large area. Later on, designs were made for variable size partitioning FIR filter by using the Multiple Constant Multiplication technique, because it uses less area to design compared with the existing system. Then another method suggested was the implementation of the multiplier circuits with constant coefficients. To perform constant multiplication, operations such as subtractions, shifting of bits, additions etc are used. This is a well-studied research field relating single or multiple constant multiplication (SCM/MCM). A distinctive feature of the constant multipliers has been considered. There will be limited predefined set of N constants during run-time out of which some set of output constant 'c' can be switched. To achieve this computation, multiplexers are embedded in the arithmetic data path. The multiplication cix is performed by the resulting re-configurable constant multiplier (RCM), where x is a fixed-point input and the selectable constant's indices; *i* ranges *0* to *N-1*. There has been many ways in which the generation of RCMs was analyzed. One such prior work on is so called re-configurable/time-multiplexed constant multiplication. RCMs were shown to be more hardware proficient than using generic multipliers as redundant partial circuits can be reused to keep the number of required output constants restricted. Their optimization is critical to realize hardware efficient applications such as DCT/FFT implementations as well as multi-stage filters for the decimation or interpolation like polyphase FIR filters and adaptable run-time filters. A post-optimization of these RCMs helps to reduce the number of required multiplexers further. The detailed study for this optimization and a discussion of related terms and work is provided in the later chapters.

Fig 1.1 shows two examples of such a re-configurable constant multiplication for 12305x and 20746x. A generic multiplication circuit is designed for the multiplication of the two constant multiples 12305 and 20746. Then the circuit is optimized using the Optimal Shift Reassignment method. Observe the figures as to know how the OSR method reduces the implementation of the circuit.

All rights reserved by *www.ijsrd.com*

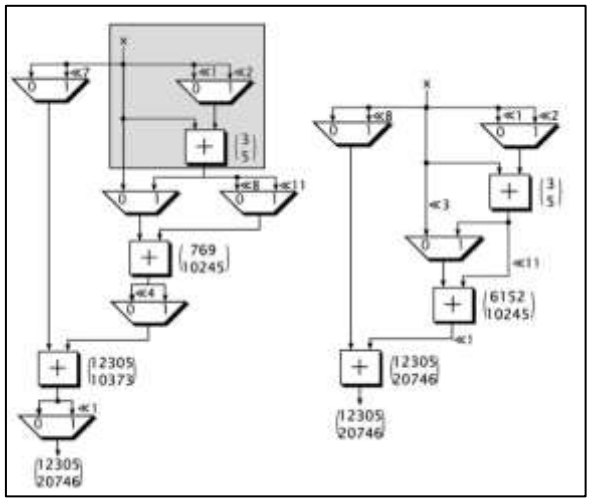

Fig 1.1: Optimization of an original re-configurable constant multiplier solution by DAG fusion

Fig 1.1 (a) shows an original RCM solution obtained from the DAG fusion algorithm [5]. Fig 1.1 (b) shows the optimized version using the proposed Optimal Shift Reassignment (OSR) approach. Both circuits perform a re-configurable constant multiplication, and they consist of adders, bit-shifts, and multiplexers. Column vectors are used to display the final result's scaling factors and the intermediate results beside the respective adder. The upper and the lower value in the column vector belong to a multiplexer selection of 0 and 1, respectively.

The highlighted part of the circuit in Fig 1.1 (a) calculates $3x = x+2^1x$ or $5x = x + 2^2x$, depending on whether the selected multiplexer input is 1 or 0. Both the circuits in Fig 1.1 calculate the same output – one is an RCM circuit and the other one is OSR optimized one. Both the circuits show a considerable difference in the number of required multiplexers. Multiplexers are required in case if the unshifted or shifted inputs of adders have different sources for the different re-configurable constants. Shift reassignment does not change the architectural property of the RCM circuits. The main contribution of prior works has been the minimization of these multiplexers. Moreover, if adder's inputs for the different re-configurable constants have the same source but different shift values, multiplexers are required to select different shift values. It clearly means that if most of the adders' input shifts for the different re-configurable constants were equal, it would be beneficial. Aligning already equal shifts is considered. There the shift values are adopted from odd fundamental graphs as input without any modification. This means the technique forced all intermediate constants to be odd. This property is advantageous in multiplier-less single constant multiplication (SCM) and multiple constant multiplications (MCM) without re-configuration, as it simplifies the optimization. However, the minimum number of required multiplexers is found when re-configurable SCM or MCM is considered is not guaranteed. By dropping this property of allowing reassignment of shift values within each RCM output constant circuit, better results could be achieved. This can be seen in the example showing the distribution of shifts and the intermediate constants in Fig 1.1. OSR is applicable to all the previous solutions for the multiplier-less RCM. This two-step process still does not lead to completely global optimal solutions. But it does improve the state of the art considerably. A globally optimal solution could be reached by including the OSR into the process of multiplier-less RCM generation. These RCM circuits can be then used for a variety of different applications. This project proposes the method of design of an FIR filter with re-configurable multipliers optimized using OSR method which has not been considered and evaluated for the design of an FIR filter so far. A study of papers related to re-configurable circuits, OSR technique and different methods adopted to improvise constant multiplier circuits has been presented in the following sections.

II. LITERATURE SURVEY

Moller et al[1] presents a new method (Optimal Shift Reassignment)to optimize the re-configurable multiplication circuits. These circuits consist of adders, subtractors, shifts and multiplexers. In this method, the multiplication of a desired number is calculated by one out of many constants that are selected dynamically during run-time. This paper contributes to the implementation of multiplication of constant coefficients. A special feature of the constant multipliers is considered. There will be limited predefined set of N constants during run-time out of which some set of output constant c can be switched. It consists of a predefined set of N constants during run-time out of which some set of output constant c can be switched. To achieve this, multiplexers are embedded in the arithmetic data path. The multiplication c_{ix} is performed by the resulting re-configurable constant multiplier (RCM), where x is a fixed-point input and the selectable constant's indice, *i* ranges *0* to *N-1*. Different operations such as additions, subtractions and shifting of bits only are used for constant multiplications.

Dempster et al.[2] proposes a new method for formulating constant integer multiplication. It requires lesser number of adders in general compared to a canonic signed digit (CSD) representation. Graphs are used to demonstrate multiplier implementation. A general suboptimal algorithm is presented for the design of multipliers of any word-length.

Gustafsson et al.[3] introduced simplifications to multiplier graphs that extends its previous work on minimum adder multipliers to five adders. It is shown that this is enough to express all coefficients up to 19 bits. The simplifications include addition reordering and vertex reduction to see that different graphs can generate the same coefficient sets.

Voronenko et al.[4] proposes a new algorithm for the MCM problem. A variable can be multiplied by a given set of fixed-point constants using a multiplier block that consists exclusively of additions, subtractions, and shifts. The generation of a multiplier block from the set of constants is known as the multiple constant multiplication (MCM) problem.

Martin Kumm et al.[5] addresses the direct optimization technique of pipelined adder graphs (PAGs) for the high speed multiple constant multiplication problem (MCM). The optimization methods are described and a description of the pipelined MCM (PMCM) problem is given. It is observed that the PMCM problem is an overall generalization of the MCM problem with the limited adder

All rights reserved by www.ijsrd.com

depth (AD). A novel algorithm named RPAG named to solve the PMCM problem heuristically is presented.

Martin Kumm et al.[6] analyzed two re-configurable FIR filter architectures based on the CFGLUT primitives which can be mapped to all modern FPGAs of Xilinx.

Tummeltshammer, et al.[7] studied area-efficient arithmetic circuits that multiplies a fixed-point input value selectively by one out of the several predefined fixed-point constants. An algorithm is presented that generates a class of solutions to the time-multiplexed MCM problem by "fusing" SCM circuits for the required constants. The evaluation compares the solution against a baseline implementation style that employs a full multiplier and a lookup table for the constants. The evaluation shows that a significant area advantage is gained, at the cost of increased latency. The evaluation further shows that this solution is better suited for standard-cell ASICs than prior works on Re-configurable Multiplier Blocks (ReMBs). This paper has solution is based on "fusing" SCM circuits of the desired constants.

Moller et al.[8] adopted the idea of Tummeltshammer et al. [6] for pipelined RCMs on FPGAs. Introducing pipelining is necessary to overcome long routing delays in FPGAs. This work discusses a new heuristic to generate a pipelined run-time RCM circuit for FPGAs. Chen et al. [9] presents a digit-re-configurable finite impulse response (FIR) filter architecture with a very fine granularity. It provides a flexible yet compact and low-power solution to FIR filters with a wide range of precision and tap length. Based on the architecture, an 8-digit re-configurable FIR filter chip is implemented in a single-poly quadruple-metal 0.35- m CMOS technology.

III. PROPOSED SYSTEM

The Proposed System consists of an FIR filter in which the generic multiplier is replaced with a Re-configurable Constant Multiplier (RCM) block that is already optimized using Optimal Shift Reassignment (OSR) method.

A. Re-Configurable Constant Multiplier (RCM) Block

A re-configurable constant multiplier is a multiplication circuit. In this, the scaling constant can be selected from a restricted predefined set of constants during the run-time. For the given application domains, two to six of such adjustable coefficient sets are very common. The switching during the run-time is achieved by inserting the multiplexers into several available constant multiplication circuits, in order to achieve a reuse of the redundant partial circuits. Therefore, a reduction of the required resources occurs. The problem is to find the possibly best solution when inserting the fusing multiplexers into the circuit. In order to avoid the large routing delays, technique such as pipelining is used for high-speed applications. In contrast to application specific ICs, this is specifically advisable for FPGA designs, due to their inherent performance disadvantage.

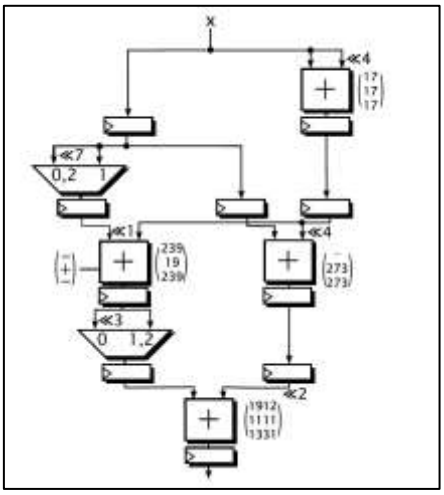

Fig. 1.2: Re-configurable single constant multiplier which can be switched between the constants 1912, 1111, 1331

An example for such a pipelined re-configurable constant multiplier that can be switched between the constants 1912, 1111, 1331 can be seen in the Fig 1.2. Pipelined registers are inserted after each stage that includes registers in the multiplexer stages. The wires can be related with a left shift and a sign. The value vector noted besides each of the operation corresponds to the intermediate or the output factors for a specific configuration of multiplexer. A switchable adder/subtractor is considered as an adder with an additional input with a sign vector. These RCM circuits can be again re-optimized using Optimal Shift Reassignment method.

B. Optimal Shift Reassignment method

The OSR method is applied to already optimized re-configurable constant multiplication circuits. To analyze, the solutions of Tummeltshammer et al.'s DAG fusion are taken into consideration. DAG fusion is a method used to generate the non-pipelined multiplier-less RCMs based on optimal SCM solutions. Optimal SCM means minimization of the number of required adders in the input circuits. The foundation of such kind of multiplier-less multiplication is in their composition of an addition of shifted inputs. A constant shift can be hard-wired. It is assumed that these hard-wires can be realized at no cost. All the approaches mainly focusses on minimizing the adder costs while the construction of RCMs and multiplexer. But no special attention provided to the shift value selection. It is seen that OSR method focusses on the shift values. Re-assignment of these shift values can reduce the total number of required multiplexers in the multiplier-less re-configurable multiplication circuits.

In the example in Fig 1.1, a reduction of 50% of required multiplexers can be seen. Obviously, this example is exceptionally good, therefore it is especially suitable to explain the effects of the shift reassignment method. The important steps to get the optimized form of circuit in the Fig 1.1 (b) out of Fig 1.1 (a), which is repeated in Fig 1.3, are described in the following.

All rights reserved by www.ijsrd.com

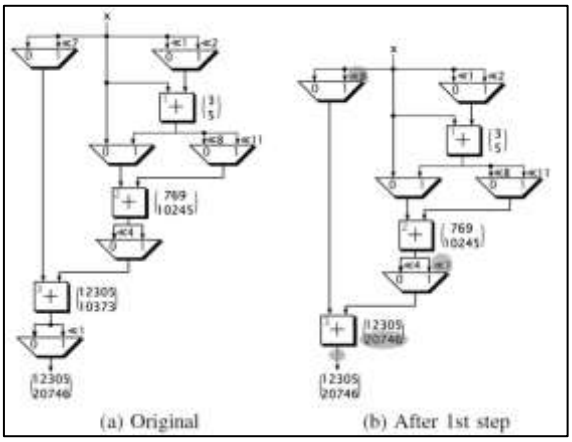

Fig. 1.3: Original and intermediate result after step one for the optimization of the re-configurable multiplier shown in Fig 1.1

First, observe the left shift of 1 at the input of the output multiplexer of Fig 1.3 (a). It is moved to the inputs of the previous adder's multiplexers within the circuit of the re-configurable constant output, c_1 = 20746. The intermediate constant of adder 3 changes from 10373 to 20746 and therefore, eliminates the multiplexer at the output. To retain a valid circuit with valid output, the shift at the left input of adder 3 is changed from 7 to 8. Also, the shift present at the right input of the adder 3 is moved from 0 to 1 in the circuit of the constant c1. The resulting circuit achieved after this step can be seen in Fig 1.3(b). In the next step, a left-shift of 3 at the resulting output of adder 2 is transferred to the adder's input multiplexers within the circuit of c_0 =12305. The shift at the left-input of adder 2 is moved from 0 to 3 in the circuit of constant c_0. A shift of 1 exists after adder 2, but it is now similar to the shift which was reassigned in the first step and hence, no multiplexer is needed. At the same time, both the shifts at the right input of adder 2 are 11. So the multiplexer can be removed. The resulting optimized circuit is the one shown in Fig 1.1 (b).

C. An FIR Filter with OSR Optimized RCM Block

In this method, the proposed design consists of an FIR filter in which the generic multiplier is replaced with a Re-configurable Constant Multiplier (RCM) block that is already Optimized using Optimal Shift Reassignment (OSR) method.

1) Fir Filters

For various DSP applications we use digital FIR filters such as in loud speaker equalization devices, in echo cancellation tools, in adaptive noise cancellation tools, speech processing techniques and in many applications of communications, including software defined radio device. A Finite Impulse Response (FIR) filter has impulse response of finite duration, because it settles to zero in finite time.

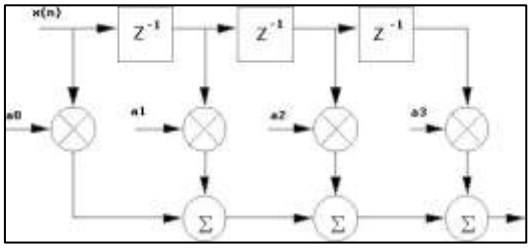

Fig. 1.4: A 4-tap FIR filter

Fig 1.4 show a generic 4 tap FIR Filter. It is observed that for high speed in digital communication, filters need high rate of sampling. Moreover, it is observed for each filter output that the number of additions and multiplications required increases in linear with order of filter. As redundant computation is not available in the FIR filter algorithm its real time implementation in an environment which is resource restricted is a demanding job. Usually coefficients of filters are constant and they can be used to reduce multiplication complexity. The MCM method reduce additions number which are required for recognition of multiplication by common sub-expression sharing for a value of input which is multiplied by large constant sets. The MCM method is more systematic because an operand which is common is multiplied by number of constants.

For implementing FIR filters of large order with fixed coefficients MCM method is suitable. But the drawback with MCM blocks is that they are form in transpose form configuration only. Redundant partial circuits can be reused. So as long as the number of required output constants is limited, RCMs were shown to be much hardware efficient than using generic multipliers. Their optimization is critical to realize hardware efficient run-time adaptable filters, DCT/FFT implementations as well as some multi-stage filters for operations as decimation or interpolation like poly-phase FIR filters. A post-optimization of these RCMs further reduces the number of required multiplexers. One such post optimization technique is the OSR Method. The OSR method is established on the idea that shift values can be located at different positions throughout the circuit, maintaiining the calculated output constant as the same. This contradicts from prior approaches that were limited by the reality that all constants within the constant multiplier were compelled to be odd. This method eventually releases this restriction. So, the number of required multiplexers in the circuit can be reduced. This happens when the shift reassignment aligns the shift values of different inputs of a multiplexer. In this project, the generic multiplier is replaced with an optimized RCM circuit. This RCM circuit is then again optimized using OSR method to reduce the number of multiplexers. The generic FIR filter can be redrawn as shown in the figure 1.5

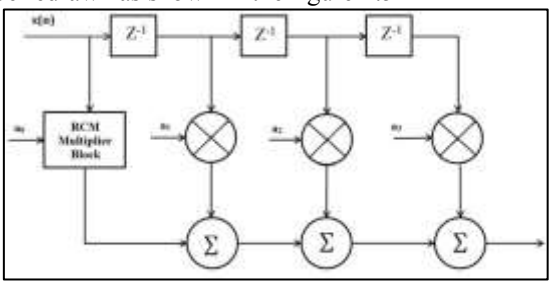

Fig. 1.5: A 4-tap FIR filter with RCM multiplier block

Fig 1.5 shows a 4 tap FIR Filter replaced with an RCM circuit optimized using OSR method.

2) OSR Optimized RCM Blocks

In this project, the generic multiplier in a 4-tap FIR Filter is replaced with an optimized RCM circuit. This RCM circuit is then again optimized using OSR method to reduce the number of multiplexers. In order to design a 4-tap FIR filter, we need 4 different coefficients – a_0, a_1, a_2, a_3 . Let us consider the four coefficients to be 122, 201, 202, 209 respectively. Firstly, the RCM blocks for these four coefficients are

All rights reserved by www.ijsrd.com

redesigned using the OSR optimized RCM blocks structure as shown in the figure 1.1(b). These optimized RCM blocks is then placed replacing the multiplier in the 4-tap FIR Filter. The design for the RCM block for the required four co-efficient are shown in the figure 1.6 (a) and figure 1.6 (b). Figure 1.6 (a) shows the design optimized for getting the multiplication constants for the coefficients 209 and 202. Whereas, figure 1.5(b) shows the design optimized for getting the multiplication constants for the coefficients 201 and 122. For the multiplexer input 0, the multiplication constants for the coefficients 209 and 201 are obtained. Whereas, for the multiplexer input 1, the multiplication constants for the coefficients 202 and 122 are obtained. 'x' is the multiplication input factor.

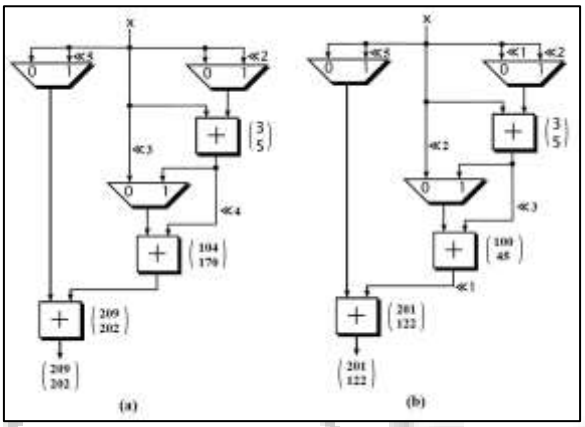

Fig. 1.6: RCM Multiplier Block for the coefficients 209,202 and 201,122

IV. Simulation Analysis

The coding is done using VHDL language and the simulation is verified using Xilinx. The following simulation results are verified:

1) Original RCM block (as in figure 1.1 for the coefficient 12305)
2) Original RCM block (as in figure 1.1 for the coefficient 20746)
3) OSR optimized RCM block for the coefficients 12305 & 20746
4) OSR optimized RCM blocks for the coefficients 209 and 202
5) OSR optimized RCM blocks for the coefficients 201 and 122
6) A 4-tap FIR filter
7) A 4-tap filter with OSR optimized RCM blocks

Let us analyze the outputs of the following blocks.

A. Re-Configurable Constant Multiplier Block

Firstly, coding were done and simulated for a basic RCM blocks for the coefficients 12305 and 20746. Following are the simulation results.

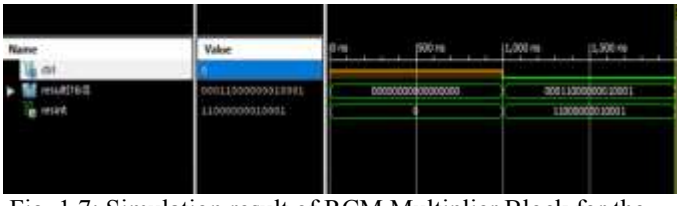

Fig. 1.7: Simulation result of RCM Multiplier Block for the coefficient 12305

In this, the RCM multiplier bock is examined for the multiples of 12305 when the ctrl is '0'. In the below figure, the RCM block is examined for the ctrl signal '1'for the output 20746.

Fig. 1.8: Simulation result of RCM Multiplier Block for the coefficient 20746

Next, the RCM block is optimized using Optimal Shift Reassignment method with lesser number of multiplexers. It is checked whether the outputs are multiples of 12305 and 20746 for the control lines 0 and 1 respectively.

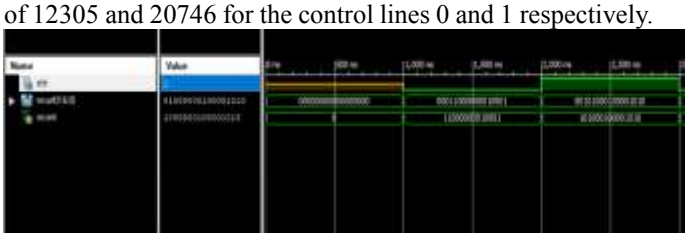

Fig. 1.9: Simulation result of OSR Optimized RCM Multiplier Block for the coefficients 12305 and 20746

In the next step, the FIR filter is designed for four random coefficients, supposedly 122, 201, 202 and 209. The RTL schematic of the FIR filter is as shown.

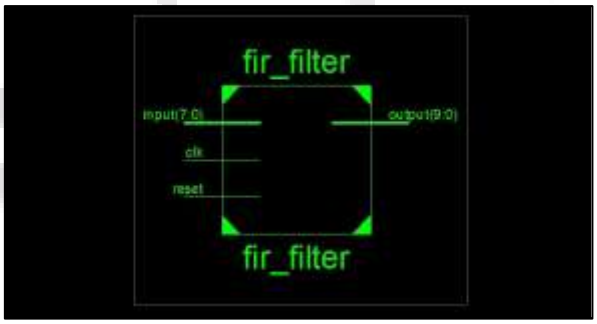

Fig. 1.10: Simulation result of OSR Optimized RCM Multiplier Block for the coefficients 12305 and 20746

The RCM multiplier block is re-designed for the coefficients 122, 201, 202 and 209. VHDL coding is done and the simulation result was observed as follows:

Fig. 1.11: Simulation result of a 4 tap FIR filter with coefficients 122, 201, 202 and 209

The multiplier of the 4-tap FIR filter is replaced with the OSR optimized RCM blocks for the coefficients 122,201,202,209 that is shown in the figure 1.6 (a) and (b). The RTL schematic of the new OSR optimized RCM FIR filter is as shown

All rights reserved by www.ijsrd.com

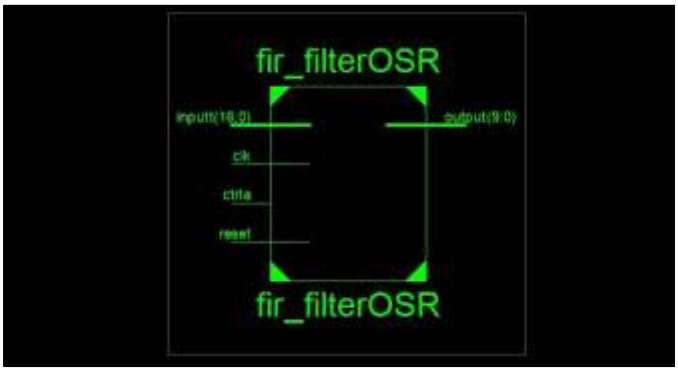

Fig. 1.12: RTL schematic of the FIR filter with OSR optimized RCM block with coefficients 122, 201, 202 and 209

Finally, the simulation result after the VHDL coding of the FIR filter with OSR optimized RCM block with coefficients 122, 201, 202 and 209 is as shown

Fig. 1.13: Simulation result of FIR filter with OSR optimized RCM Block

It is observed that the simulation output is obtained in lesser number of clock cycles. Also comparing the other constraints, we get the following results.

Device Utilization and Timing Summary Report		
Parameters	4 Tap FIR Filter	FIR Filter with OSR optimized RCM block
Number of LUT Flip Flop pairs used	135	131
Number with an unused Flip Flop	34 out of 135 (25%)	30 out of 131 (22%)
Number with an unused LUT	37 out of 135 (27%)	29 out of 131 (22%)
Number of fully used LUT-FF pairs	64 out of 135 (47%)	72 out of 131 (54%)
Minimum period	3.223ns	3.212ns
Maximum Frequency	310.308MHz	310.308MHz
Total number of paths	1533	1416

Table 1.1: Device Utilization and Timing Summary Comparison Report of the 4 tap FIR filter and the 4 tap FIR filter with OSR optimized RCM bock with coefficients 122, 201, 202 and 209

V. Conclusion

An FIR filter architecture with re-configurable constant multiplication block that is optimized using optimal shift reassignment method is proposed. The comparison of the results of different parameters like delay, frequency and device slice parameters are shown. Optimal Shift Reassignment method is used to optimize the RCM block. The OSR method is based on the idea that shifts can be placed at different positions along the circuit, while the calculated output constant stays the same. The number of required multiplexers in the circuit can be reduced. This happens when the shift reassignment aligns the shift values of different inputs of a multiplexer. Thus the complexity and device parameters are reduced for proposed structure. In future, the block size and filter length can be increased as per the requirement and can be used for specific purpose. The simulation result after the VHDL coding of the FIR filter with OSR optimized RCM block with coefficients 122, 201, 202 and 209 is also shown.

Acknowledgment

I would like to convey my heartfelt gratitude towards my guide Ms. Bency Varghese A for her constant guidance, encouraging help and inspiring words. I am thankful to the department of computer engineering for their support.

References

[1] Moller, K., Kumm, M., Garrido, M., & Zipf, P. (2018). “Optimal Shift Reassignment in Re-configurable Constant Multiplication Circuits”. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 37(3), 710–714. doi:10.1109/tcad.2017.2729467

[2] A. G. Dempster and M. D. Macleod, “Constant Integer Multiplication Using Minimum Adders,” Circuits, Devices and Systems, IEE Proceedings, vol. 141, no. 5, pp. 407–413, Oct 1994.

[3] O. Gustafsson, A. G. Dempster, and L. Wanhammar, “Extended Results for Minimum-Adder Constant Integer Multipliers,” IEEE Int. Symp. on Circuits and Systems, vol. 1, pp. I–73–I–76, Aug 2002.

[4] Y. Voronenko and M. P¨uschel, “Multiplierless Multiple Constant Multiplication,” ACM Trans. Algorithms, vol. 3, no. 2, May 2007.

[5] M. Kumm, P. Zipf, M. Faust, and C.-H. Chang, “Pipelined Adder Graph Optimization for High Speed Multiple Constant Multiplication,” in IEEE Int. Symp. on Circuits and Systems, May 2012, pp. 49–52.

[6] Kumm, M., Moller, K., & Zipf, P. (2013). Dynamically re-configurable FIR filter architectures with fast reconfiguration. 2013 8th International Workshop on Re-configurable and Communication-Centric Systems-on-Chip (ReCoSoC). doi:10.1109/recosoc.2013.6581517

[7] P. Tummeltshammer, J. C. Hoe, and M. P¨uschel, “Time-Multiplexed Multiple-Constant Multiplication,” IEEE Trans. Comput.-Aided Design Integr. Circuits Syst., vol. 26, no. 9, pp. 1551–1563, Sep. 2007.

[8] K. Moller, M. Kumm, M. Kleinlein, and P. Zipf, “Re-configurable Constant Multiplication for FPGAs,” IEEE Trans. Comput.-Aided Design Integr. Circuits Syst., vol. 36, no. 6, pp. 927–937, Jun. 2017.

[9] Chen, K.-H., & Chiueh, T.-D. (2006). A Low-Power Digit-Based Re-configurable FIR Filter. IEEE Transactions on Circuits and Systems II: Express Briefs, 53(8), 617–621

[10] Indranil Hatai, Indrajit Chakrabarti, and Swapna Banerjee, "A Computationally Efficient Re-configurable

All rights reserved by www.ijsrd.com

Constant Multiplication Architecture Based on CSD Decoded Vertical–Horizontal Common Sub-Expression Elimination Algorithm", IEEE Transactions On Circuits And Systems–I: Regular Papers 1,1549-8328 © 2017 IEEE.

[11] R.Brindha and Dr.G.Rajakumar, "Multiple Constant Multiplication Based Variable Size Partitioning FIR Filters", in International Conference on Smart Systems and Inventive Technology (ICSSIT 2018).

[12] Martin Kumm, Member of IEEE, "Optimal Constant Multiplication Using Integer Linear Programming", in IEEE Transactions On Circuits And Systems—II: Express Briefs, Vol. 65, No. 5, May 2018.

[13] Mohanty, B. K., & Meher, P. K. (2016). A High-Performance FIR Filter Architecture for Fixed and Re-configurable Applications. IEEE Transactions on Very Large Scale Integration (VLSI) Systems, 24(2), 444–452.

[14] Sang Yoon Park, & Meher, P. K. (2014). Efficient FPGA and ASIC Realizations of a DA-Based Re-configurable FIR Digital Filter. IEEE Transactions on Circuits and Systems II: Express Briefs, 61(7), 511–515.

[15] Mahvish Quraishi and V.D. Alagdeve, "Energy Efficient Re-configurable Fir Filter Architecture", International Conference on Communication and Signal Processing, April 6-8, 2017, India

[16] Ahmet Can Mert, Hasan Azgin, Ercan Kalali, Ilker Hamzaoglu, "Efficient Multiple Constant Multiplication Using DSP Blocks in FPGA", 2018 International Conference on Field-Programmable Logic and ApplicationsR. H. Turner and R. F. Woods, "Highly Efficient, Limited Range Multipliers for LUT-Based FPGA Architectures," IEEE Trans. VLSI Syst., vol. 12, no. 10, pp. 1113–1118, Oct 2004.

[17] S. S. Demirsoy, I. Kale, and A. G. Dempster, "Re-configurable Multiplier Blocks: Structures, Algorithm and Applications," Circuits, Systems and Signal Processing, vol. 26, no. 6, pp. 793–827, Jan 2008.

[18] J. Chen and C. H. Chang, "High-Level Synthesis Algorithm for the Design of Re-configurable Constant Multiplier," IEEE Trans. Comput.- Aided Design Integr. Circuits Syst., vol. 28, no. 12, pp. 1844–1856, Dec2009.

All rights reserved by *www.ijsrd.com*