International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

A MILP based Optimization and FPGA Implementation of Efficient Polyphase Multirate Filters

[1]Dr Gopal S. Gawande [2]Dr. Dhanraj R.Dhotre [3]Dr Nitin Choubey, [4]Dipali S. Mate

Submitted: 28/01/2023 **Accepted:** 02/04/2023

Abstract: Effectively implementing digital filters at cheap cost is made possible by the DSP field's multirate filtering method. By taking advantage of concurrency in multirate digital filters, the processing burden of polyphase decomposition techniques is reduced. Since each positive bit requires an extra adder in circuitry execution and more toggling at the end of CMOS circuits, reducing the number of positive terms in the filter coefficients is a primary focus of filter improvement. In order to lessen the quantity of nonzero factors in the filter coefficients, an innovative method is proposed in this article. Polyphase shapes are used to test the suggested algorithm's effects on power usage. The constructions are manufactured for Spartan6 xc6slx150T-4fgg676 FPGA panel consuming system generator for obtaining the other performance indices like throughput, speed, logic area and computation rate. In spite of occupying more area, efficient polyphase decimator structure is found to be superior to the polyphase and decimation structures.

Keywords*: Polyphase Structure; MILP; Power consumption; Logic area; system generator; Computation rate; Throughput .*

1. Introduction

The conversion of sample rates is an operation that is important to modern Digital Signal Processing (DSP) programmes and is also utilised often in these programmes. In the vast majority of these applications, a sample rate converter of very high quality is needed. Interpolators and decimators are two types of digital signal processing tools that may be used to alter the sample rate of a digital signal [2, 9], [10]. Multirate systems are able to complete a processing job with better performance characteristics while simultaneously delivering those improved performance characteristics at a cost that is much lower than more conventional methods. When down sampling is used in a multirate system, a decimation filter is one of the main building pieces that makes up the system. The attenuation of undesirable aliasing is directly governed by the accuracy of the design of the decimation filter, making it one of the most important aspects of the filter. The use of polyphase interpolator and decimator structures makes it possible to execute interpolation and decimation with high levels of efficiency. The polyphase breakdown of the transfer function of the interpolation or decimation filter is what leads to the formation of patterns like these [1, 4]. There are a few different approaches that may be used when putting the polyphase decimation filter into action. This platform was used to carry out the implementation. By decreasing the amount of ones or power of two (POT) components in the filter coefficients, the dynamic power consumption in digital circuits may be brought down to a more manageable level. The price of the filter coefficient is stated as the number of bits in the coefficient that are not zero, also known as the number of ones. In this article, we provide a unique technique that we created in order to determine an ideal value of the filter coefficient in terms of cost, without compromising the frequency characteristics that were specifically required. Parameters like as speed, throughput, calculation rate, logic area, and power consumption are used in an investigation into and comparison of the performance of polyphase architectures.

2. Literature Review

S. New techniques for calculating interpolation and decimation of signals were presented by S. Emami [2]. [3] Kai-Yuan Cheng conducted a review of different designs of finite impulse response (FIR) digital filters, and he utilised numerous design techniques to decrease the complexity of the hardware required for low-power and high-speed applications. P. Jacob and colleagues came up with an innovative design for FIR filters, decimation filters, and polyphase decimation filters, and they implemented all of these filters using FPGA. The findings demonstrated that a polyphase decimation filter, in contrast to traditional filters, is capable of slowing down the rate at which an input data stream is sampled [11].

One of the most significant drawbacks of the FIR filter is its

1Associate professor : Deptt of E & TC Engg. Marathwada Mitra Mandal's College of Engineering Karve Nagar, Pune
gopalgawande@mmcoe.edu.in
2Faculty Computer Deptt., School of Engineering,MIT Art Design & Technology University,Pune.
dhanraj.dhotre@mituniversity.edu.in
3Faculty, SVKM'S NMIMS MUKESH PATEL SCHOOL OF TECHNOLOGY MANAGEMENT & ENGINEERING, Shirpur, Dist : Dhule
nitin.choubey@nmims.edu
4BE ME Computer Sci.&Engg. Pune,
dipumate@gmail.com

computational complexity, which is caused by the large number of multiplications required. This makes the computational complexity of the FIR filter the most significant drawback. The researchers have come up with a few different suggestions on how to cut down on the amount of electricity used. These techniques optimise hardware by using a variety of approaches, such as reducing the total number of POT terms, maximising the normalised peak ripple, maximising the ripple optimization, and decreasing the amount of quantization [15, 18, 19, 20]. MILP,PP, and SDP are some of the methods that may be used to locate the best possible solution [17, 22, 24, 25]. The traditional MILP method has the benefit of ensuring the production of the best possible design (the global minimum), but it requires an unnecessary amount of processing time. The research that has been done on this topic has uncovered a number of different strategies that may be used to reduce the amount of nonzero components included in filter coefficients [16–23]. Finding the minimal value of the FIR filter issue may be expedited by the use of local search rather than global search to cut down on the amount of time needed for calculation of the MILP [26]. In the paper [6], a local search technique is used in order to cut down on the cost of the FIR filter. A unique hybrid monotonic local search method has been presented by Ahmed Shaheinet al. [6] for the purpose of optimising filter coefficients, with a significant increase in the rate at which computations may be performed. This reduction in the search space required to determine the ideal value of the filter coefficient was the key to achieving this gain in computing time. Under the framework of the suggested method, our goal is to reduce the amount of money spent on the filter coefficients. The technique for maximising filter coefficients using the local search strategy described in [6] has been simplified and implemented by our team after it was developed. Also, the complexity of the technique presented in [6] is simplified in the approach that is provided.

3. Design and Realization of Decimation Filter

It is common knowledge that a decimator has an anti-aliasing lowpass filter and then a downsampler as its two main components. It can be shown that an anti-aliasing filter has the transfer function H(z) given by (1).

$$H(z) = \frac{Y(z)}{X(z)} = \sum_{k=0}^{N-1} h(k)\, z^{-k} \qquad (1)$$

The concept of the decimator may be accomplished utilising a variety of different architectures. Use of polyphase structures allows for an effective execution of this concept. Polyphase decomposition is a method that may split a filter into M-sections of sub-filters that have the capability of being realised in parallel [7][13]. The phase characteristics are the sole point of differentiation amongst the sub-filters in this breakdown. In an anti-aliasing filter, the coefficients h(k) are split into M polyphase sub-filters, each of which has (N/M) taps, where N is the length of the filter and M is the decimation factor.

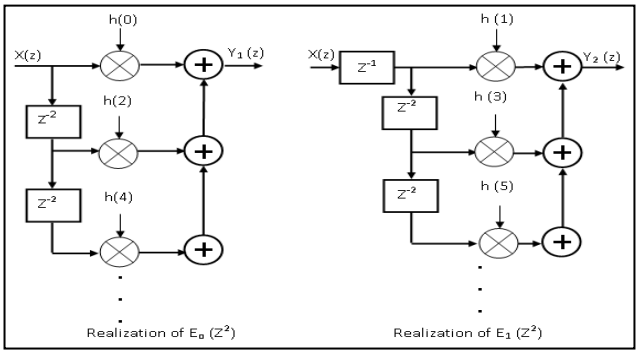

Fig.1. Direct form realization of $E_0(z^2)$ and $E1(z^2)$

The next section will offer the mathematical explanation of the polyphase decomposition that was discussed before.

Following is a list of words that result from expanding (1):

$H(z) = z^0\, h(0) + z^{-1}\, h(1) + z^{-2}\, h(2) + z^{-3}\, h(3) + z^{-4}\, h(4) + z^{-5}\, h(5)\ldots.$

Let us rearranged the term into two sections:

$H(z) = [z^0\, h(0) + z^{-2}\, h(2) + z^{-4}\, h(4) + \ldots.] + [\, z^{-1}\, h(1) + z^{-3}\, h(3) + z^{-5}\, h(5)\ldots]$

$H(z) = [(z^2)^0 h(0) + (z^2)^{-1}\, h(2) + (z^2)^{-2}\, h(4) + \ldots.]$

$+ z^{-1}\, [(z^2)^0 h(1) + (z^2)^{-1} h(3) + (z^2)^{-2} h(5)\ldots.]$

$H(z) = E_0(z^2) + z^{-1} E_1(z^2)$

where, $E_0(z^2) = [(z^2)^0 h(0) + (z^2)^{-1}\, h(2) + (z^2)^{-2}\, h(4) + \ldots.]$ and $E_1(z^2) = [(z^2)^0 h(1) + (z^2)^{-1} h(3) + (z^2)^{-2} h(5)\ldots.]$

In Figure 1, we can see the direct form FIR realizations for both E0(z2) and E1(z2). In order to create the polyphase implementation of the FIR decimation filter structure, E0(z2) and E1(z2) may be linked in parallel, and the input will need to be separated by a delay. The polyphase implementation of a decimation filter with two portions is shown in figure 2 below. [3] [8][12].

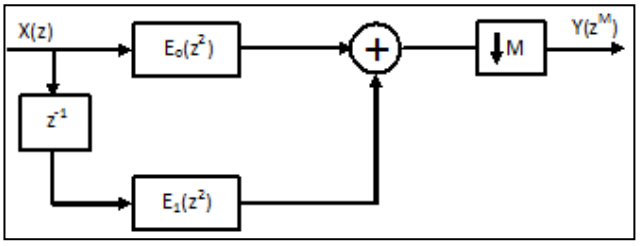

Fig.2. Filter for the polyphase elimination of phases

The decimator structure seen in Fig. 2 is computationally inefficient since it only chooses one output sample for every M samples it receives from the input. It is possible to achieve

a high level of computational efficiency by including a down sampler block before the multiplier in the FIR filter structure. This will ensure that the processed samples are not wasted. Figure 3 is a block diagram that illustrates the functioning of the superposition principle, and Figure 4 is a schematic that illustrates the consequent efficient polyphase decimator [5, 14].

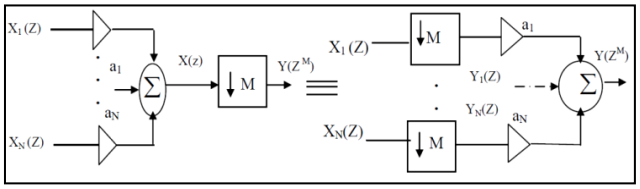

Fig.3. The General Superposition Principle

4. Performance Indices of FPGA Implementation

Speed, power consumption, logic area, throughput, and calculation rate are the metrics that are used to evaluate the performance of FPGA implementations. The use of logic devices extracted from the FPGA fabric is what gives an indication of the logic area in FPGA implementations. Since it is a synchronous device, an FPGA has to be supplied with a clock signal. The clock rate is what determines how quickly the DSP algorithm that is implemented in FPGA can do its tasks. It is measured in Mega Hertz (MHz), and it is designated by maximum frequency (F), which is explicitly stated by FPGA suppliers in order to offer some indication of performance. The throughput () of a system may be determined by using the formula presented in (2). Inversely related to the size of the input is the total throughput, which is directly proportional to the operating frequency (F) and the degrees of parallelism (). (N). It is expressed as mega samples per second as a unit of measurement (MSPS). The calculation rate, denoted by the symbol (), is derived from a formula described in (3); its unit of measurement is the number of Mega Multiply Accumulate operations performed in one second (MMACPS). It has a relationship that is inversely proportional to the maximum clock frequency and the levels of parallelism (). (F) [28].

$$\mu = \frac{F * \alpha}{N} \qquad (2)$$

$$\gamma = F * \alpha \qquad (3)$$

I. Power Consumption in Digital Circuits

The fabrication of digital integrated circuits often makes use of the CMOS technology. The entire amount of power PTotal that CMOS digital circuits utilise may be expressed as (4).

$$P_{Total} = P_{Switching} + P_{Short} + P_{Leakage} \qquad (4)$$

where, The term $P_{Switching}$ refers to the process of switching power, which occurs as a result of the charging and discharging of capacitors that are driven by the circuit. $P_{Short}$is the power that is generated by the short circuit currents that are generated when pairs of PMOS/NMOS transistors are conducting simultaneously. $P_{Leakage}$is the power that is generated by the leakage that is induced by substrate injection and sub-threshold effects. In the article, our primary goal was to determine how to reduce the switching power (Dynamic) of a CMOS circuit (referred to in 5) [27]).

$$P_{switching} = \frac{1}{2} \times C_L \times V_{DD}{}^2 \times f_{clock} \times E_{SW} \qquad (5)$$

where CL represents the gate's output load, VDD represents the supply voltage, f_{clock}represents the clock frequency, and ESW refers to the switching activity of a gate. By lowering the number of POT terms and, as a result, the number of adders in the hardware implementation, dynamic power consumption is lowered as well.

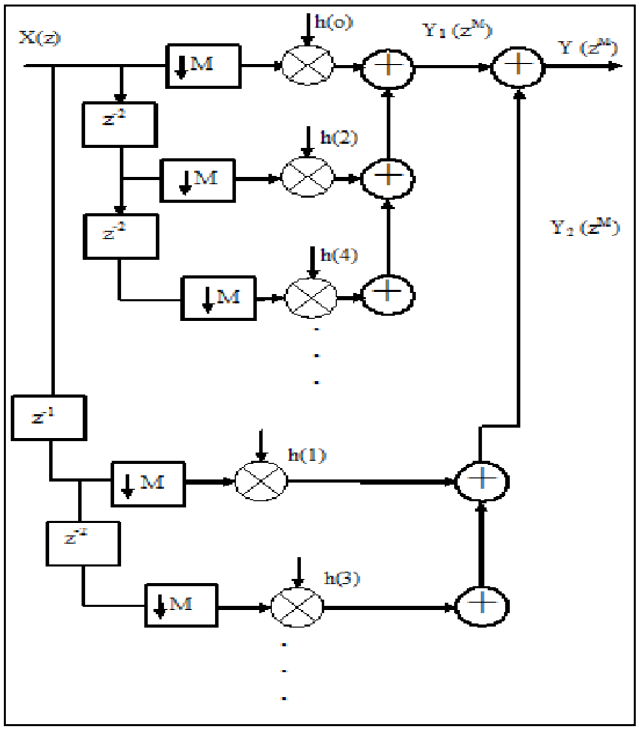

Fig. 4. Efficient Polyphase decimation filter

5. Proposed Algorithm for Coefficient Optimization

Finding ideal values of the filter coefficients while incurring the lowest possible cost is the primary purpose of the method that has been developed. The algorithm begins its search for the best value for each coefficient by first determining the minimum and maximum values for that coefficient. The objective of the coefficient optimization issue is stated in (6) with the caveat that it is subject to the constraints stated in (7).

$$\text{Minimize} \sum_{k=1}^{M1} \text{cost}(\hat{h}(k)) \quad (6)$$

Subject to:

$$\{(1-\delta p)\times 2^{Q-1}\} \le \hat{A}(\omega) \le \{(1+\delta p)\times 2^{Q-1}\}\ ;\ \omega T \in [0,\omega_p]$$

$$-(2^{Q-1}\times \delta s) \le \hat{A}(\omega) \le (2^{Q-1}\times \delta s)\ ;\ \omega T \in [\omega_s,\pi] \quad (7)$$

Where (k) represents the scaled coefficients that were acquired from the quantized filter coefficients h(k) by applying the scaling factor, (8). () is the scaled zero-phase resonance response or the amplitude of the phase-locked FIR filter, and it can be calculated by using the following formula: (9). It is a real number that represents the amplitude of the Fourier transform of k's Fourier coefficients [6, 23]. Quantization bit breadth is denoted by Q, and passband ripple, stopband ripple, standardized passband edge velocity in radians per sample, and adjusted stopband edge frequency in radians, or per sample are correspondingly passband noise, stopband ripple, and generalized passband edge velocity in radians per example.

$$\begin{aligned}\hat{h}(k) &= \lfloor h(k)\times 2^{(Q-1)}+0.5\rfloor; && h(k)>0\\ &= \lceil h(k)\times 2^{(Q-1)}-0.5\rceil; && h(k)<0\\ &= 0; && h(k)=0 \quad (8)\end{aligned}$$

$$\hat{A}(\omega)=\sum_{k=1}^{M}\hat{h}(k)\times c(k,\omega T) \quad (9)$$

From the theory of Fourier transform, constant c(k,ωT) for odd filter length is given by (10) while for even filter length it is given by (12). Due to symmetry only half of the filter coefficients (M1) are considered for computation and is given by (11) for odd filter length and by (13) for even filter length.

For odd filter length,

$$c(k,\omega T)=x(k)\times\cos\big(\omega T(k-1)\big) \quad (10)$$

where,

$$x(k)=\begin{cases}1; & k=1\\ 2; & k=2,3,\dots,M1\end{cases}$$

and

$$M1=\left\lfloor\left(\left(\frac{N}{2}\right)+1\right)\right\rfloor \quad (11)$$

For even filter length,

$$c(k,\omega T)=2\times\cos\big(\omega T(k-1)\big)\,;k=1,2,3,\dots\dots,M1 \quad (12)$$

and

$$M1=\left\lfloor\left(\frac{(N+1)}{2}\right)\right\rfloor \quad (13)$$

The amount of time required for calculation can be decreased by narrowing the search space to locate the best number for each coefficient. Therefore, the upper limit (Ub) and the lower bound (Lb) are computed for each coefficient k by using the equations (14) and (15) in the appropriate places [6].

$$Lb(\hat{h}(k))=\begin{cases}2^{L},\ L=\lfloor\log_2(|\hat{h}(k)|)\rfloor & \hat{h}(k)>0\\ -2^{L},\ L=\lfloor\log_2(|\hat{h}(k)|)\rfloor, & \hat{h}(k)<0\\ 0\ , & \hat{h}(k)=0\end{cases} \quad (14)$$

$$\text{Ub}(\hat{h}(k))=\begin{cases}2^{U}\ ,\ U=\lceil\log_2(|\hat{h}(k)|)\rceil\ , & \hat{h}(k)>0\\ -2^{U}\ ,\ U=\lceil\log_2(|\hat{h}(k)|)\rceil, & \hat{h}(k)<0\\ 0\ , & \hat{h}(k)=0\end{cases} \quad (15)$$

The proposed algorithmic steps for minimizing cost of filter coefficients are as follows:

Step 1: Specifications for the input filter (Low pass), including the passband edge fraction (Fp) in Hertz, the stopband edge fraction (Fst) in Hertz, the passband attenuation (Ap) in dB, the stapband attenuation (As) in dB, the sampling frequency (Fs) in Hertz, Q and N. Using the formula presented in (16), obtain the normative passband edge digital frequency (p) from its analog cousin (Fp). Similarly, the generalized passband edge the internet frequency (s) is to be formula presented in (16).

$$\omega_p=\left(2\times\frac{F_p}{F_s}\right)\pi \quad (16)$$

Step 2: Determine the values of the variable-valued filter coefficients utilizing the optimum filter design methodology that is applicable.

Step 3: These coefficients should be quantized and rounded off before being reduced to a defined amount of bits.(Q).

Step 4: In order to achieve, we scaled up the coefficients of the coefficients. ĥ(k) using (6).

Step 5: Perform the binary conversion on all of the k and then tally up the expenses. It is presumed that the cost of each coefficient is the highest possible cost for that particular coefficient, and this cost is represented by the symbol m.

Step 6: Define a search space that is limited by the antecedent lower limits Lb((k)) and subsequent higher constraint Ub((k)) integers for the scalable parameter (k) that is obtained by using solutions (14) and (15) correspondingly. This will allow you to find the optimum replacement for a filter variable.

Step 7: Determine the evolution vector EVL((k), with the goal of producing a collection of potential integers with a cost that is lower than m for each component that has been limited by the lower and higher limits that were allocated in the stage before this one.

Step 8: Create groups out of the collection of potential integers Ci((k)) based on their prices; that is, integers that share the same price must be included in the same group. Where, i is between zero and (m-1).

Step 9: Now, attempt each option from $C_i(k)$ one at a time to see whether they fulfil the criteria outlined in (7) for the grid of T's possible values. If it fulfils the conditions, the procedure should be terminated, and you should consider that value to be the best possible one for that specific coefficient. In the event that none of the possible integer values are able to fulfil the requirements, the initial coefficient should be kept. '(k) opti' is where you should store the optimised coefficients. This process will continue until the optimum replacements have been found for all of the coefficients.

Step 10: Convert $\hat{h}$(k)_opti to h(k)_opti using (17).

$$h(k)\_opti = \frac{\hat{h}(k)\_opti}{2^{Q-1}} \qquad (17)$$

So, all the optimized coefficients are stored in h(k)_option.

6. Implementation

MATLAB 2013a was used during the development of the unique reduced local search method for coefficient optimization that was suggested. The Equiripple FIR filter design approach was used during the development of the anti-aliasing filter. The Filter Design and Analysis (FDA) Tool tool in MATLAB is used to acquire the floating point coefficients. After that, the coefficients are quantized to the standard 16.14 fixed point forms. The technique employs a tactic that involves building a filter with a stopband attenuation value that is marginally higher than the one that is wanted. For instance, if the target stopband attenuation is -57 dB, then the filter should be constructed for -60 dB, and then a flexibility of 3 dB should be allowed for optimization purposes. It was determined from the results that were provided by the previous researchers [6] that the stopband attenuation could tolerate a relaxation of 3 dB. In order to ensure that the filter criteria are met to the letter, the conditions described in number five need to be validated over a fine grid of temperature values. In the approach that has been presented, these conditions are examined using a fine grid with 1024 values of T ranging from 0 to while simultaneously determining the best value for each coefficient.

The performance of several polyphase structures is evaluated and compared using the Xilinx system generator, which is a block-level simulation tool. The decimation filter architectures are then generated for the Spartan 6 FPGA board. The Xilinx system generator makes it possible to programme Xilinx devices by using simulink, which is an environment for MATLAB programming that is widely used and known for its ease of use. It is a high level design tool that works very well for the purpose of generating bespoke DSP data routes in FPGAs. It offers a collection of simulink blocks (models) for a number of different hardware operations that might be carried out on a variety of Xilinx FPGAs.

7. Results and Discussions

Two decimators have been created specifically for the requirements outlined in Table 1. It is clear from looking at Table 3 that the polyphase decimation filter takes the least amount of logic space. As compared to polyphase decimation filter, an efficient polyphase filter uses 22 percent more FFs, 0.7 percent more LUTs, and 5.4 percent more slices. Although though there was only a little increase in the amount of logic area, the speed, throughput, and calculation rate of the efficient polyphase decimation filter for Filter 1 were all significantly boosted. It is clear from looking at Table 6 that the power consumption for Filter1's efficient polyphase structure is at its lowest possible level. The application of optimised coefficients to these structures results in a further decrease in the amount of power that is used. In the event that Filter 1 uses an efficient polyphase filter, this value is decreased by about 31 mW.

In addition to that, the synthesis is carried out for Filter 2, and the same kinds of outcomes are acquired. Table 4 presents the results of a comparative investigation of the highest frequency, throughput, and calculation rate. As can be shown in Table 5, an efficient polyphase decimation filter utilises (23.21%) more FFs, (3.5%) more LUTs, and (6.7%) more slices than the construction of a polyphase decimation filter. When we applied the optimised coefficients to the decimation Filter 2, using the structures that were described, we saw a reduction in the amount of power that was being used that ranged from 3 to 11 mW.

TABLE 8 presents the quantized and optimised coefficients that were obtained by applying the suggested approach to Filter 1. Also included is the cost of these coefficients. It has been discovered that the total price of the decimator has dropped from 192 to 179. Figure 5 shows the frequency response of the decimator both with and without optimised coefficients. This ensures that the necessary filter parameters are met.

Table 1. Specifications of decimation filters

Filter	**M**	**ω_p** (radians/sample)	**ω_s** (radians/sample)	**A_p (dB)**	**A_s (dB)**	**N**

Filter 1	3	0.1428 π	0.3334 π	1	-58.0	21
Filter 2	4	0.1071428 π	0.25 π	1	-56.25	28

Table2. Performance indices of filter 1

Performance indices	**Decim-ation Filter**	**Polyphase Decimation Filter**	**Efficient Polyphase Decimation Filter**
Maximum Frequency (MHz)	43.169	98.164	111.794
Throughput (MSPS)	2.05	14.023	15.971
Computation Rate (MMACPS)	43.169	294.492	335.382

Table3. Logic devices utilization for Filter 1

Logic devices	**Decimation Filter**	**Polyphase Decimation Filter**	**Efficient Polyphase Decimation Filter**
No. of slice registers/FFs	1376	1364	1754
No. of slice LUTs	7381	7229	7282
No. of slices occupied	2048	2004	2120

Table4. Performance indices for Filter 2

Performance indices	**Decimation Filter**	**Polyphase Decimation Filter**	**Efficient Polyphase Decimation Filter**
Maximum Frequency (MHz)	30.646	84.239	107.204
Throughput (MSPS)	1.09	12.034	15.315
Computation Rate (MMACPS)	30.646	336.956	428.816

Table5. Logic devices utilization for Filter 2**Logic devices**	**Decimation Filter**	**Polyphase Decimation Filter**	**Efficient Polyphase Decimation Filter**
No. of slice registers/FFs	1835	1766	2300
No. of slice LUTs	9941	9367	9710
No. of slices occupied	2738	2600	2789

Table 6. Power Consumption by Polyphase Decimation Structures for Filter 1

Power consumption (mW)	**Decimation filter**	**Polyphase decimation filter**	**Efficient polyphase decimation filter**
Without optimized coefficients	268	251	214
With optimized coefficients	264	251	183

TABLE 7. POWER CONSUMPTION BY POLYPHASE DECIMATION STRUCTURES FOR FILTER 2

Power consumption (mW)	**Decimation filter**	**Polyphase decimation filter**	**Efficient polyphase decimation filter**
Without optimized coefficients	287	260	229
With optimized coefficients	276	252	226

8. Conclusion

It is clear, based on the findings of the synthesis performed for Filter 1 and Filter 2, that the power consumption of an efficient polyphase decimation filter is much lower in comparison to that of the other structures. Using the suggested method results in a further decrease in the amount of electricity that is used. So, it is possible to draw the conclusion that the suggested method has the effect of reducing the total number of ones. Because of this decrease in POTs, there will surely be less switching activity in CMOS and fewer adders in the hardware implementation, which will result in a large reduction in the amount of power that is used dynamically. In comparison to all of the previous decimation filter designs, the polyphase decimation filter structure has consumed a less amount of logic space. The efficient polyphase decimation filter structure is preferable than the basic polyphase decimation filter structure because it has greater speed, a higher computation rate, and a higher throughput. This is true despite the fact that it occupies a little larger logic space. It has been discovered that an efficient decimation filter structure is a more power-efficient realisation.

References

[1] H. Johansson and L. Wanhammar, "Filter Structures Composed of All-Pass and FIR Filters for Interpolation and Decimation by a Factor of Two*," IEEE Trans. On Circuits and Systems-II: Analog and Digital Signal Processing*, vol. 46, no.7, pp. 896–905, July 1999.

[2] ShahriarEmami, "New Methods for Computing Interpolation and Decimation Using Polyphase Decomposition," IEEE Trans. On Education, vol.42, no.4, pp.311–314, Nov. 1999.

[3] Kai-Yuan Cheng, "Multiplierless Multirate FIR Digital Filter /Decimator / Interpolator Module Generator," National Central University Jhongli 320, Taiwan, R.O.C., 2003.

[4] M. B. Yeary, W. Zhang, J. Q. Trelewicz, Y. Zhai and B. McGuire, "Theory and Implementation of a Computationally Efficient Decimation Filter for Power-Aware Embedded Systems," IEEE Trans. Instru. Meast., 55(5), pp-1839–1849,2006.

[5] N. Onwuchekwa, G.A. Chukwudebe, "Implementation Of Computationally Efficient Algorithms for Multirate Digital Signal Processing Systems," Afr. Journal of Comp. & ICT, vol. 1, no. 1, pp. 33 – 45, 2008.

[6] Ahmed Shahein, Qiang Zhang, NiklasLotze, and YiannosManoli, "A Novel Hybrid Monotonic Local Search Algorithm for FIR Filter Coefficients Optimization," IEEE Trans. on Circuits and Systems—I: Regular Papers, vol. 59, no. 3, March 2012.

[7] N. Younis, M. Ashour, and A. Nassar, "Power-Efficient Clock/Data Distribution Technique for Polyphase Comb Filter in Digital Receivers," IEEE Trans. Circuits Syst. II, Express Briefs, 56(8), pp-639–643, 2009.

[8] Dr.K.B.Khanchandani, Kundan Kumar, "Design and Implementation of Custom Low Power DSP blocks for Biomedical Applications," Int. Journal of Advanced Engineering & Application, 2011.

[9] M. Madheswaran and V. Jayaprakasan, "Implementation And Comparison Of Different CIC Filter Structure For Decimation," *ICTACT Journal On Communication Technology*, 4(2), pp-709–716, 2013.

[10] V. Jayaprakasan and M. Madheswaran, "FPGA Implementation of FIR based Decimation Filter Structure for WiMAX Application," *International Journal of Advanced Research in Computer and Communication Engg.*, vol. 2, pp. 2830–2837,2013.

[11] P. Jacob and Mr. Anoop B.N, "Design and Implementation of Polyphase Decimation Filter," *International Journal of Computer Networks and Wireless Communications (IRACST)*, vol. 4, no. 2, pp. 123–127, April 2014.

[12] Rajendra M. Rewatkar, Dr. Sanjay L. Badjate, "Optimization of Multirate Polyphase Decimator using MCM and Digit Serial Architecture," *International Journal of Computer Science and Information Technologies*, vol. 5, 2014.

[13] Robert D. Turney, Chris Dick, and Ali M. Reza, "Multirate Filters and Wavelets: From Theory to Implementation", *Xilinx Inc., San Jose*, CA 95124, USA.

[14] E. Ifeachor and B. W. Jervis, *Digital Signal Processing*. 2nd ed. India: Pearson, 2011.

[15] D.M. Kodek, "Design of Optimal Finite Wordlength FIR Digital Filters Using Integer Programming Techniques," *IEEE Trans. On Acoustics, Speech, and Signal Processing*, vol. ASSP-28,no. 3, pp. 304–308, June 1980.

[16] [16]Y.C. Lim, R. Yang, D.N. Li, J.J. Song, "Signed Power-of-Two Term Allocation Scheme for the Design of Digital Filters," *IEEE Trans. on Circuits and Systems-II: Analog and Digital Signal Processing*, vol.46, no.5, pp. 577–584, May 1999.

[17] O. Gustafsson, L. Wanhammar, "Design of Linear-Phase FIR Filters Combining Subexpression Sharing with MILP," in *Proceedings of 45th Midwest symposium on Circuits and Systems (MWSCAS'02)*, vol. 3, pp. 9–12, Aug. 4-7, 2002.

[18] Y.C. LIM, "Design of Discrete-Coefficient-Value Linear Phase FIR Filters With optimum Normalized Peak Ripple Magnitude," *IEEE Trans. on Circuits and Systems*, vol. 37, no.12, pp.1480-1486, Dec. 1990.

[19] Chao-Liang Chen, Michael C. Chen, Alan N. Willson Jr, "A Trellis Search Algorithm for the Design of FIR Filters with Signed-Powers-of-Two Coefficients," in *Proceedings of IEEE Midwest symposium on Circuits and Systems*, Ames, IA,pp.619-622, Aug. 18-21,1996.

[20] Stefano Traferro, FulvioCapparelli, Francesco Piazza and Aurelio Uncini, "Efficient Allocation Of Power-of-Two Terms In FIR Digital Filter Design Using Tabu Search," in *Proceedings of IEEE International symposium on Circuits and Systems (ISCAS'99)*, Orlando, FL, vol.3, pp.411-414, May 30- June 02, 1999

[21] Chia. Yu Ya, "A Study of SPT-term distribution of CSD numbers and its application for designing fixed-point linear phase FIR filters," in *Proceedings of IEEE International symposium on Circuits and Systems (ISCAS'01)*, vol.2, pp.301-304, 2001.

[22] Tetsuya Fujie Rika Ito Kenji SuyamaRyuichiHirabyashi, "A new heuristic signed-power of two term allocation approach for designing of FIR filters," in *Proceedings of IEEE International symposium on Circuits and Systems (ISCAS'03)* , vol. 4, pp.285-288, May 25-28, 2003.

[23] Oscar Gustafsson, Håkan Johansson, and Lars Wanhammar, "An MILP Approach for the Design of Linear-Phase FIR Filters with Minimum Number of Signed-Power-of-Two Terms," in *Proceedings of European Conference on Circuit Theory Design*, Espoo, Finland, 2001.

[24] W.S. Lu and T. Hinamoto, " Design of FIR filters with Discrete Coefficients via Polynomial Programming: Towards the Global Solution," in *Proceedings of IEEE International symposium on Circuits and Systems (ISCAS'07)*,New Orleans, LA, pp. 2048-2051, May 27-30, 2007.

[25] W.S. Lu, "Design of FIR filters with Discrete coefficients: a Semidefinite Programming Relaxation Approach," in *Proceedings of IEEE International Symposium on Circuits and Systems (ISCAS'01)*, Sydney, NSW, vol. 2, pp. 297-300, May 6-9, 2001.

[26] Z. Ye and C.-H. Chang, "Local search method for FIR filter coefficients synthesis," in *Proceedings of IEEE International Workshop on Electronic Design, Test and Appllications (2nd DELTA'04)*, pp. 255–260, 2004.

[27] Roger Woods, John McAllister, Y. Yi and G. Lightbody, *FPGA-based Implementation of Signal Processing Systems.* John Wiley and Sons, 2008.

[28] Sami Kadhim Hasan, "FPGA Implementations for Parallel Multidimensional Filtering Algorithms," Ph.D Thesis, Agriculture and Engineering Newcastle University, June 2013.

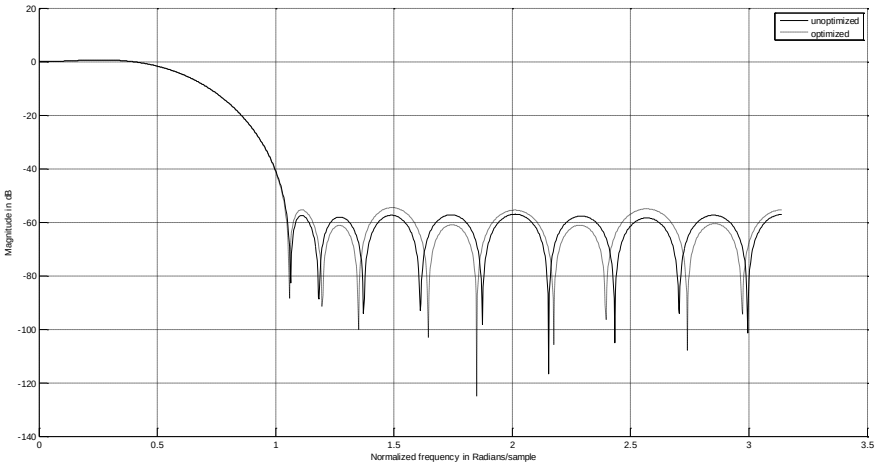

Fig.5. Frequency response of decimation filter with and without optimized coefficients

TABLE 8.

COMPARISON OF OPTIMIZED AND UNOPTIMIZED COEFFICIENTS

Quantized coefficients (16:14 format)	Unoptimized Scaled Coeff.	Cost	Optimized Scaled	Cost	Optimized Coefficient
-	65415	12	65415	12	-0.003662109375
-	65197	12	65197	12	-0.01031494141
-	64901	10	64900	09	-0.01937866211
-	64685	11	64684	10	-0.02597045898
-	64791	11	64792	09	-0.02267456055
-	65461	13	65460	12	-0.002288818359
0.038085937500000	1248	04	1248	04	0.0380859375
0.093566894531250	3066	09	3066	09	0.09356689453
0.151611328125000	4968	06	4968	06	0.1516113281
0.195800781250000	6416	04	6416	04	0.1958007813
0.212341308593750	6958	08	6944	05	0.2119140625
0.195800781250000	6416	04	6416	04	0.1958007813
0.151611328125000	4968	06	4968	06	0.1516113281
0.093566894531250	3066	09	3066	09	0.09356689453
0.038085937500000	1248	04	1248	04	0.0380859375
-	65461	13	65460	12	-0.002288818359
-	64791	11	64792	09	-0.02267456055
-	64685	11	64684	10	-0.02597045898
-	64901	10	64900	09	-0.01937866211
-	65197	12	65197	12	-0.01031494141
-	65415	12	65415	12	-0.003662109375