DIAGNOSTYKA, 2025, Vol. 26, No. 3

e-ISSN 2449-5220
DOI: 10.29354/diag/208854

EFFICIENT IMPLEMENTATION OF FRACTIONAL SAMPLING RATE CONVERSION (SRC) ON FPGA

Noor Talal GADAWE, Rasha Waleed HAMAD, Sahar Lazim QADDOORI *

Electronic Engineering Department, Electronics Engineering College, Ninevah University, Mosul 41005, Iraq
* Corresponding author Email: sahar.qaddoori@uoninevah.edu.iq

Abstract

Field Programmable Gate Arrays (FPGAs) include versatile features, which make them useful for use in Digital Signal Processing (DSP)-based systems that require high levels of performance. This paper introduces hardware implementation of decimator (M), Interpolator (L), and Sample Rate Conversion (SRC) by factor L/M, using different structure realizations (direct, efficient, and polyphase). Initially, a digital low-pass Finite Impulse Response (FIR) filter is designed using the Remez algorithm for filter coefficient calculation and realized with a decimator (M=2), an Interpolator (I=5), and SRC (5/2). These design structures are implemented using Xilinx Simulink blocks on the Artix 7 (XC7A-1csg324) FPGA development board. In the decimator, the polyphase structure represents the best design in terms of resource utilization, such as registers, Look Up Table (LUT), flip-flops, total real-time, memory usage, and multiplexers, while the direct structure consumes more resources. The same results are in the Interpolator. For SRC, it can be noted that the efficient design with linear phase is better in terms of device utilizations, while the direct structure is best in the number of unique control sets and number of multiplexers.

Keywords: Finite Impulse Response (FIR), SRC, decimator, interpolator, polyphase structure

List of Symbols/Acronyms

FPGA – Field Programmable Gate Arrays;
DSP –Digital Signal Processing;
M –Decimator;
L –Interpolator;
SRC –Sample Rate Conversion;
FIR –Finite Impulse Response;
LUT –Look Up Table;
LPF – Low Pass Filter.

1. INTRODUCTION

The process of changing the effective sample rate of a discrete-time signal is known as Sample rate conversion. In real-time processing, the Sample Rate Converter (SRC) is vital to exchange digital data between two hardware modules working at altered sample rates [1].

The conversion of a signal from one sample rate to another is referred to as fractional sampling rate adjustment or resampling. This technique is utilized widely for various applications such as speech, digital audio, radar, communication systems, and antenna systems, where each of them works at a different sampling rate [2,3,4].

A multirate system employs interpolators with an integer factor L and decimators with an integer factor M. When these two elements are there, the result is an SRC system where the sampling rate is changed by a rational factor L/M. as declared in Figure 1. The filtering operation is required to avoid imaging and aliasing. This operation is illustrated in Figure 1(a), where the input signal x(n) is first upsampled by a factor L and then filtered using a low-pass interpolation filter LPF_L (denoted as $H_L(z)$). After the interpolation operation, the signal passes through a low-pass antialiasing filter LPF_M ($H_M(z)$) and is subsequently downsampled by factor M. As a result, the output signal x(m) has a sampling rate that is L/M times the original signal x(n) [5]. In practice, since both $H_L(z)$ and $H_M(z)$ work at the same sampling rate, they can be combined and replaced with a single low-pass filter H(z), as illustrated in Figure 1(b). This filter has stopband edge frequency ω_s to remove imaging, at the same time addressing the aliasing issue arising from decimation [6,7].

The FPGA-based architecture offers a low development cost, along with great flexibility and functionality, and can be developed with a faster time to market [8,9]. However, the implementation of DSP algorithms on FPGA encounters several

Received 2025-01-10; Accepted 2025-07-31; Available online 2025-08-02

basic issues, such as determining the sampling rates and computational complexity of several applications [10]. Due to the use of FPGA in the implementation of SRC devices, they are highly flexible for frequency and phase shifting as well as superior in high realizations [11].

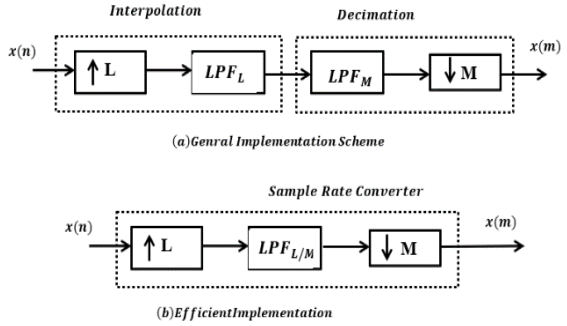


Fig. 1. Sampling Rate Conversion by L/M

The main contribution of this paper is the hardware implementation of reconfigurable architectures for decimators, Interpolators, and SRC with different structures. To assess speed performance and mapped hardware resource utilization on the XC7A-100TCSG324 FPGA, Xilinx FPGA synthesis tools are used. The components are realized by employing the direct, efficient and polyphase structures that were developed early, hence leading to improvements in area, power and speed.

This paper is organized into five sections with this section as follows: Section 2 declares the related works. Section 3 presents the research method of SRC. In Section 4, the results and discussions are done. Finally, section 5 includes the conclusions.

2. RELATED WORKS

SRC is a well-researched topic, as the presented literature proves, and there are many books and resources depicting its solutions. However, getting a specific piece of information is not easy as each solution is provided from a different view. Moreover, the literature on this topic mostly lacks information regarding the real-life application of the concept.

Mahamudul Hassan et al. [12] have described a direct approach to the design and implementation of SRC with narrow passband and transition width for converting the frequency sample rate by a factor of L/M equal 2/2 and 2/3. This approach was used to provide the necessary impulses as a bio-signal, for example, Electrocardiography (EEG) signals, with an Integrated Synthesis Environment (ISE) Suite 14.7 and Altera Cyclone DE II FPGA board.

Qingfeng Jing et al. [13] designed multi-rate digital filters, which are significant to realize multi-rate decimation and interpolation using a Hamming window, a half-band filter and a Cascaded Integrate-Comb (CIC) filter. This system is tested on an Altera FPGA development board.

Ali. Zeineddine et al. [14] studied the hardware realization of Arbitrary Sample Rate Conversion (ASRC) by variable fractional delay filter (V-FDF) constructions. The authors compared the employment of altered lately suggested V-FDF choices based on Newton for Hermite interpolation and Farrow structures. The implementation was done on both FPGA and Application-Specific Integrated Circuit (ASIC).

Dhandapani Vaithiyanathan et al. [15] proposed the design of cascaded stages of a multirate linear phase FIR filter. Different architectures are considered, namely polyphase, folded pipeline, systolic array, and Farrow architectures, with a focus on optimizing both power and speed. The analysis and synthesis are carried out using Altera Quartus-II 10.0.

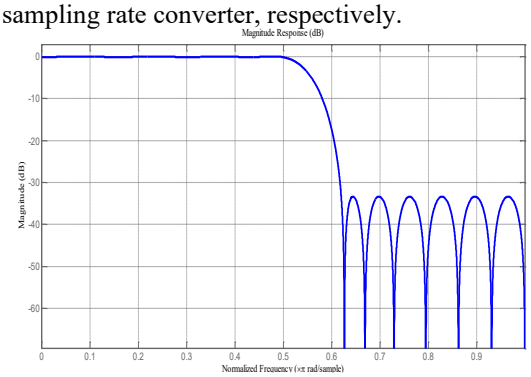
Jotirmayee Ghosh et al. [16] focused on the design and implementation of a configurable Asynchronous Sample Rate Converter (ASRC) for various digital audio applications. The algorithm begins by taking the input audio data and storing it in Random Access Memory (RAM) and coefficients stored in a Read-Only Memory (ROM). The core of the ASRC algorithm is the implementation of polyphase filters, which are used in faster processing of sample rate conversion for parallel data to ensure high-quality audio data transfer with reduced noise and improved efficiency. Shahriar Shahabuddin, et al. [17] proposed an algorithm for a sample rate converter (SRC) that is designed to handle high sample rates efficiently, particularly in embedded systems, while maintaining performance-limited clock frequencies, using polyphase SRC for parallelized algorithms and multiphase output SRC. On a Virtex-7 FPGA, the implementation of a very large-scale integration (VLSI) structure for the dual-phase output SRC was achieved.

Swetha Pinjerla et al. [10] developed and optimized a sample rate conversion structure for multi-standard radio application on the Virtex-6 XC6VCX240t-2FF484 FPGA. To avoid aliasing in the downsampling operation, a 16-taps low-pass FIR filter is used. Various techniques and considerations are employed to achieve an efficient and flexible sample rate conversion process, making it suitable for modern communication systems.

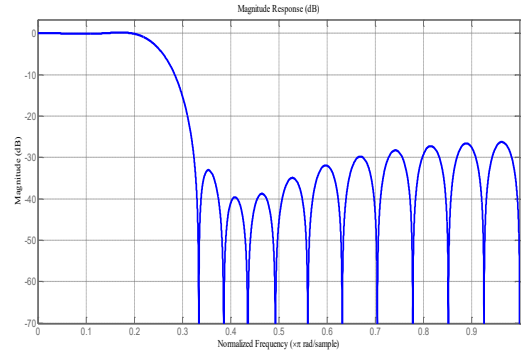
3. RESEARCH METHOD

To clarify the basic concepts of sample rate converter transformations and all related structures, an FIR low-pass filter was taken with a stable and linear phase response. The filter specifications have a stopband attenuation of 30dB, a passband ripple of 0.1dB, and a filter length of 30. The Remez algorithm is used to calculate FIR filter coefficients, as presented in Table 1. Furthermore, Figure (2) displays the filter's magnitude response for the decimator and Interpolator. The realization of the

filter can be accomplished using a decimator with M=2, an Interpolator with I=5, and SRC with M/I=5/2. The cutoff frequency w_c (pi/2, pi/5, min [pi/2, pi/5]) for the decimator, Interpolator, and sampling rate converter, respectively.



(a) Decimator



(b) Interpolator

Fig. 2. Magnitude Response of the filter

Figure 3 shows the realization of the proposed filter using a decimator with different structures. For direct realization, there are two structures. The first is found by passing the input sequence x[n] through an FIR filter and then downsampling it by a factor M=2, as shown in Figure 3(a). The second structure is derived from the first structure, which is calculated by passing all input sequences through down sampling, as illustrated in Figure 3(b). The efficient implementation of a linear-phase decimator is obtained from the use of filter coefficient symmetry, as shown in Figure 3(c). With more than two stages, the FIR filter may be implemented in the parallel structure. In Figure 3(d), the polyphase decomposition of the filter's transfer function is used to express the parallel structure of the filter.

For the Interpolator, the direct implementation has (I – 1) zeros placed at the time base of x[n], as shown in Figure 4(a), which represents the conventional system. Multiplication operations are done with the sampling rate of the input signal, and then the result sequence is up-sampled by L for each branch, as represented in Figure 4(b). An effective structure for designing a linear-phase FIR filter for the Interpolator using interpolated coefficients is shown in Figure 4(c).

It is quite the same as in decimators, and due to the inherent symmetry of the coefficients, the number of multiplications may be halved. Figure 4 (d) declares the polyphase realization of the FIR filter.

Table 1. The Coefficients of the Proposed Filter Design

Coefficients	Decimator	Interpolator	SRC
h (1) =h (30)	0.006	0.0063	0.0063
h (2) =h (29)	-0.0128	-0.0147	-0.0147
h (3) =h (28)	-0.0028	-0.0010	-0.0010
h (4) =h (27)	0.0136	-0.0028	-0.0028
h (5) =h (26)	0.0046	0.0104	0.0104
h (6) =h (25)	-0.0197	0.0214	0.0214
h (7) =h (24)	0.0159	0.0194	0.0194
h (8) =h (23)	0.0213	-0.0003	-0.0003
h (9) =h (22)	-0.0349	-0.0300	-0.030
h (10) =h (21)	-0.0156	-0.0498	-0.0498
h (11) =h (20)	0.0640	-0.0373	-0.0373
h (12) =h (19)	-0.0073	0.0184	0.0184
h (13) = h (18)	-0.1187	0.1074	0.1074
h (14) =h (17)	0.0980	0.1995	0.1995
h (15) =h (16)	0.4922	0.2579	0.2579

As for the structure shown in Figure 5(a), SRC has the general case of a direct implementation. Furthermore, SRC by a factor (I/M) may also be achieved using a linear time-variant filter described by a response function, as shown in equation

$$g\,[n,m]\;=\;h\,[nI\;-\;(m\,M)\,I] \qquad (1)$$

Where h[n] stands for an impulse response of FIR LPF for simplicity. Thus, the coefficients set {g[n, m]} for each m = 0, 1, 2, I-1. As previously pointed out, g[n, m] is periodic with a period of (I); therefore, y(m) can be expressed as shown in Eq. (2):

$$y[m] = \sum_{n=0}^{K-1} g\left(n, m - \left\lfloor \frac{m}{I} \right\rfloor I\right) * x\left(\left(\left\lfloor \frac{mM}{I} \right\rfloor - n\right)\right. \qquad (2)$$

If the rational rate conversion is possible by the factor (I/M), then there are (I) polyphase filters, as represented in Figure 5(c). Polyphase configurations enable arithmetic operations to be performed at the lowest possible sampling frequency [4, 18, 19].

4. RESULTS AND DISCUSSION

The arithmetic choice in DSP implementation on processor platforms, in most cases, is determined by the platform. Yet, in FPGA implementations, the arithmetic choice is critical not only for determining the algorithm's capacity but also for defining significant system characteristics such as area, power consumption, and speed. Where the multirate realization is attained using several approaches which include direct, efficient, and polyphase. The output is expressed in terms of input and filter coefficients. This section provides the design of the proposed filter in MATLAB's Simulink toolbox, which consists of the Interpolator, decimator, and SRC. Then, the VHDL file for the filter's Simulink implementation is developed by using the MATLAB HDL Coder library. The synthesized bitstream file is

uploaded at speed grade -1 onto FPGA (XC7A-1CSG324). Figure (6) shows the hardware implementation of the decimator with different structures.

The hardware consumption and device utilization summary are illustrated in Table (2). From this Table, it can be noted that the second direct structure consumes more slices compared with other structures for decimator and interpolator design. In contrast, the polyphase structure represents good design in resource utilization (registers, LUTs, flip-flops) by saving 50% in the decimator and 42.9% in the Interpolator as in memory usage and multiplexer. Still, the first direct realization is best in the number of unique control sets utilized. The efficient realization is the best in the analysis of real-time and CPU time to XST by saving 37% in decimator and 52% for each Interpolator and SRC, this is because of the use of symmetry of filter coefficients. Figure 7 (a-e) shows the device utilization summary. It can be noted that the number of register/flip-flop, multiplexer, and adder/subtractor is approximately equal in direct realization for all structures, while efficient realization of SRC is the best in consumption of register/flip-flop, and then polyphase realization of Interpolator and decimator, on the other word, the polyphase realization of decimator consumed less multiplexer, adder/subtractor and memory usage compared with Interpolator and SRC. The comparison of diverse SRC employments on FPGA with related works is shown in Table 3. In the related works, many SRC filters are designed and implemented with different structures and various devices, so comparison with our work is somewhat difficult.

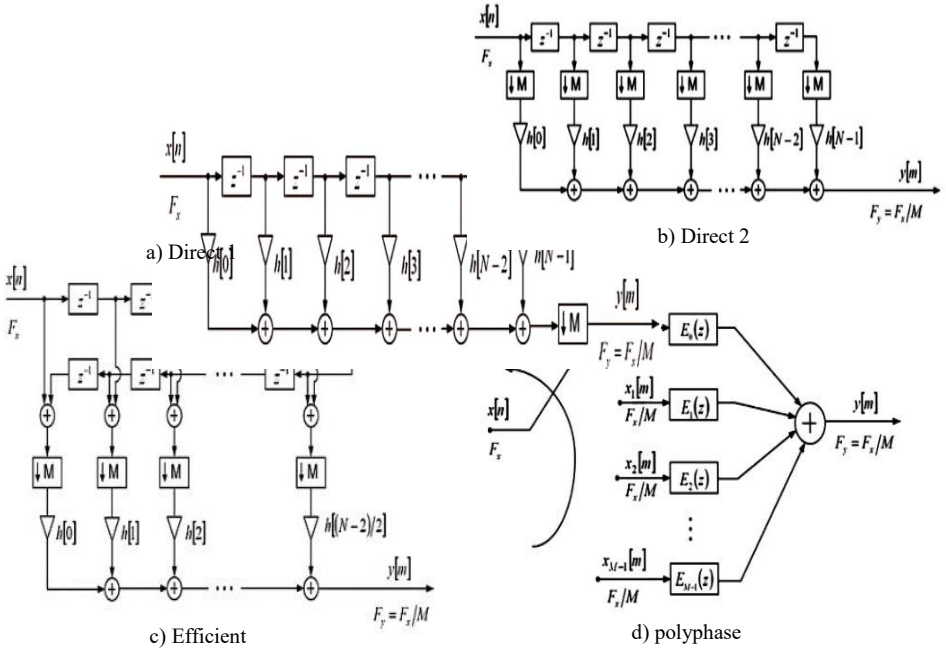


Fig. 3. Realization of Decimator with Different Structures

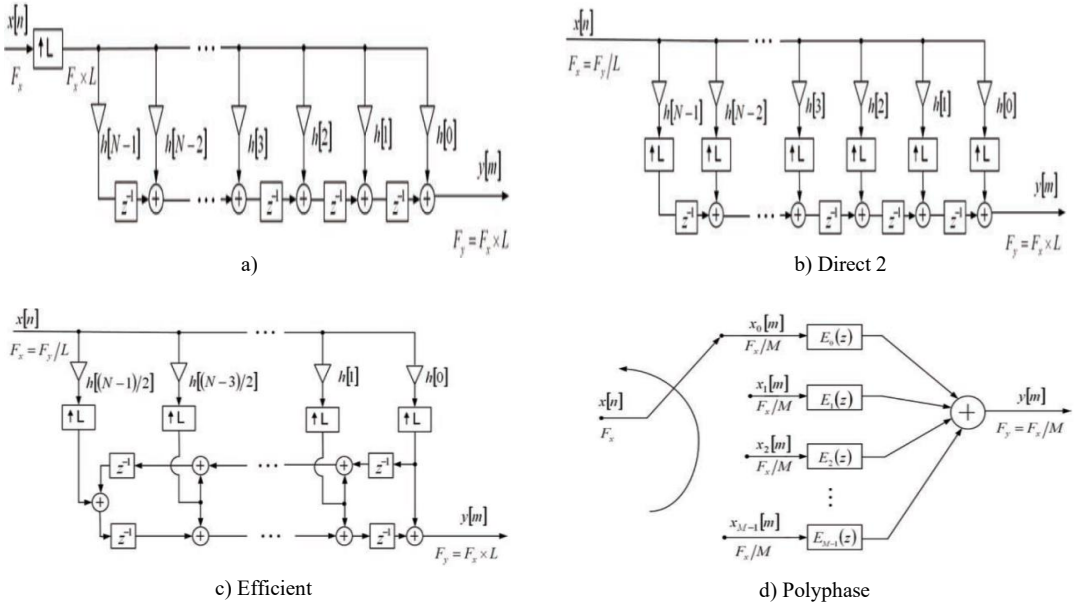
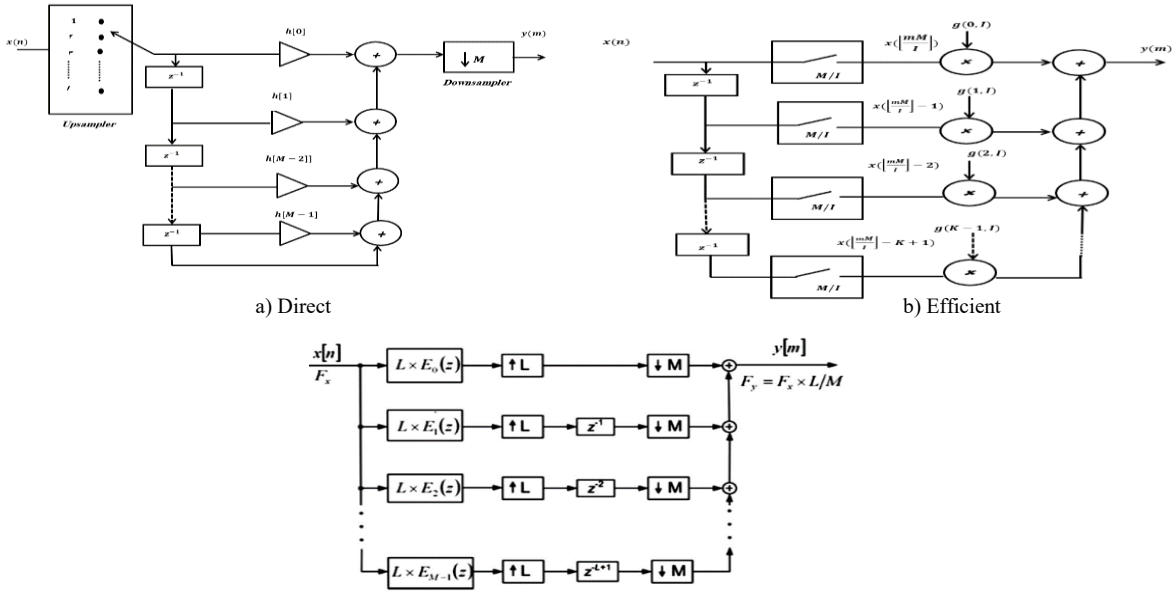


Fig. 4. Realization of Interpolator with Different Structures



a) Direct b) Efficient

c) polyphase

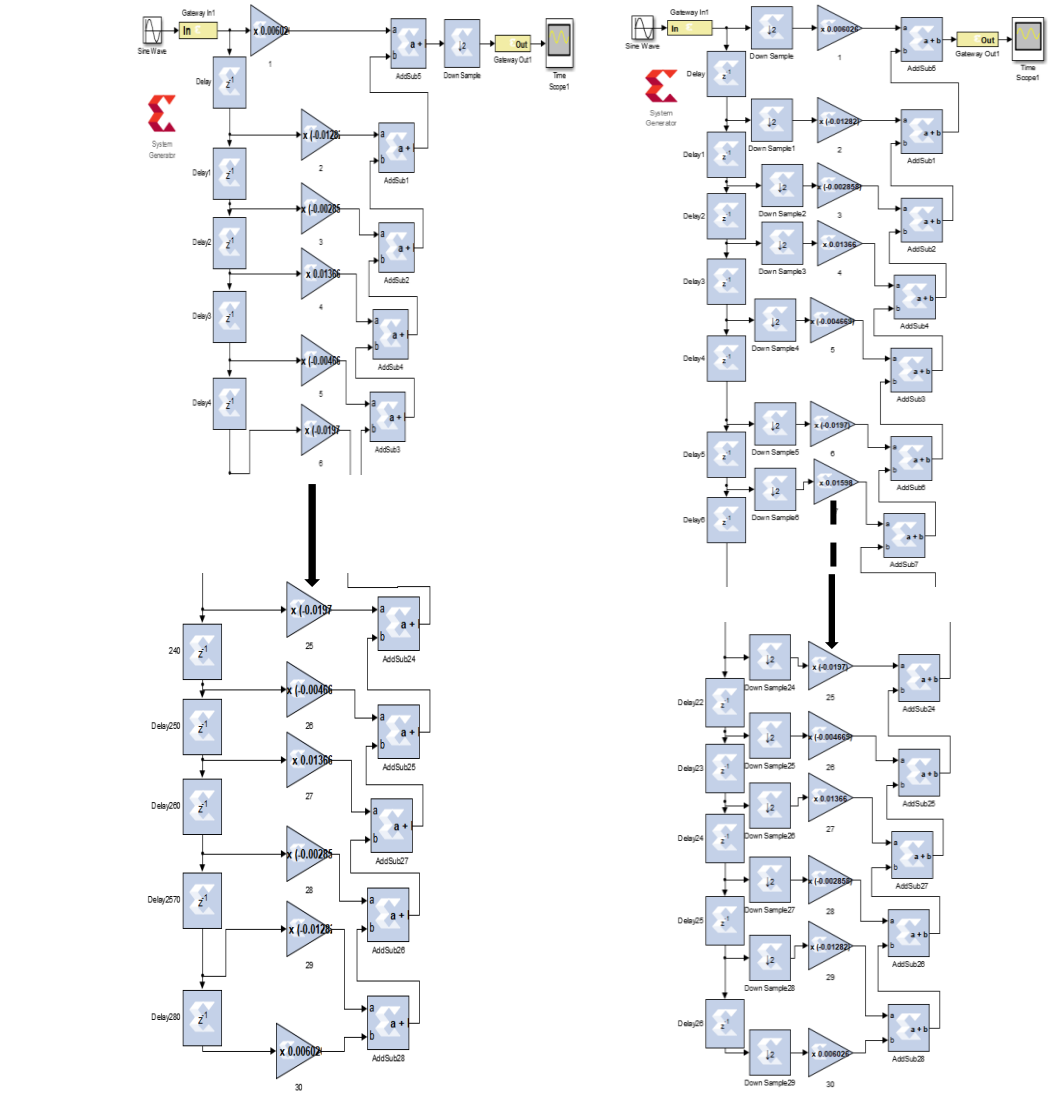
Fig. 5. Realization of SRC with Different Structures

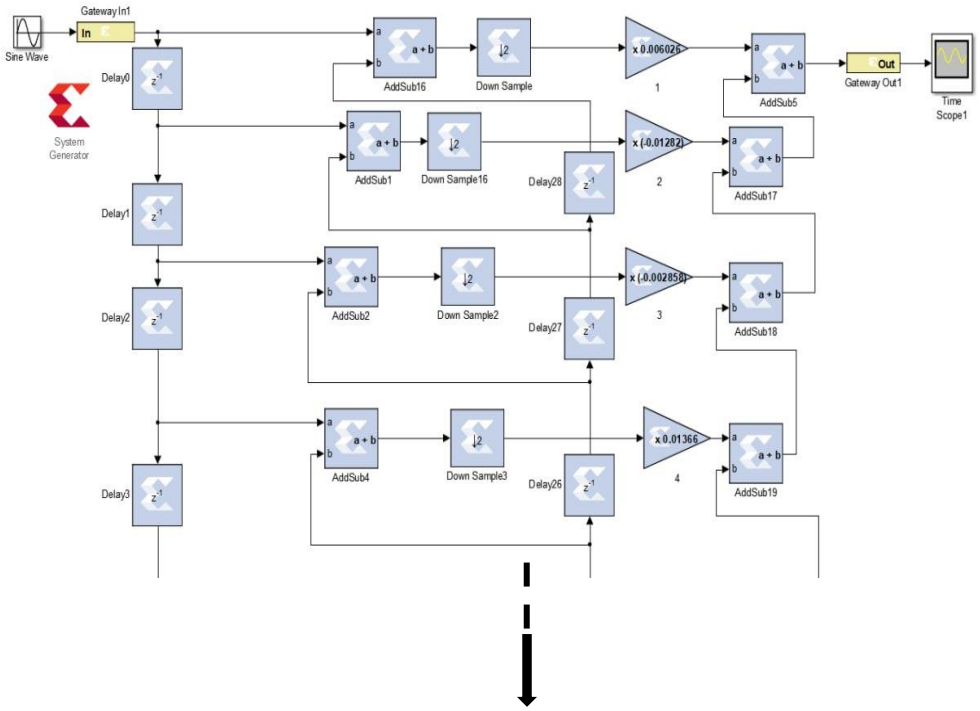
Table 2. FPGA Implementation of SRC Structures

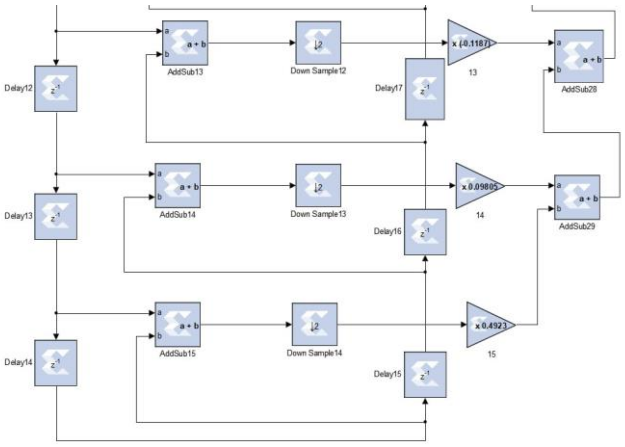
	Decimator				Interpolator				SRC		
Device Utilization Summary	D1	D2	ED	PD	In1	In. 2	E. I	P. I	SRC	E. SRC	P.SRC
Number of slices register	973	1930	1435	974	912	941	862	404	988	358	563
Number of slices LUTs	1143	2071	1036	620	1147	2075	1040	1371	1180	1500	1505
Number of LUTs flip-flop pairs used	2034	3832	1945	1484	1948	2566	1612	1571	2064	1618	1776
Number of unique control sets	3	32	17	5	3	3	3	4	4	9	10
Total real-time to XST completion(s)	18	27	10	19	21	21	11	14	11	11	21
Total CPU time to XST completion(s)	17.6	26.76	10.59	18.44	21.37	21.20	11.31	14.09	11.77	10.96	20.21

Table 3. The comparison of diverse SRC employments on FPGA with related works

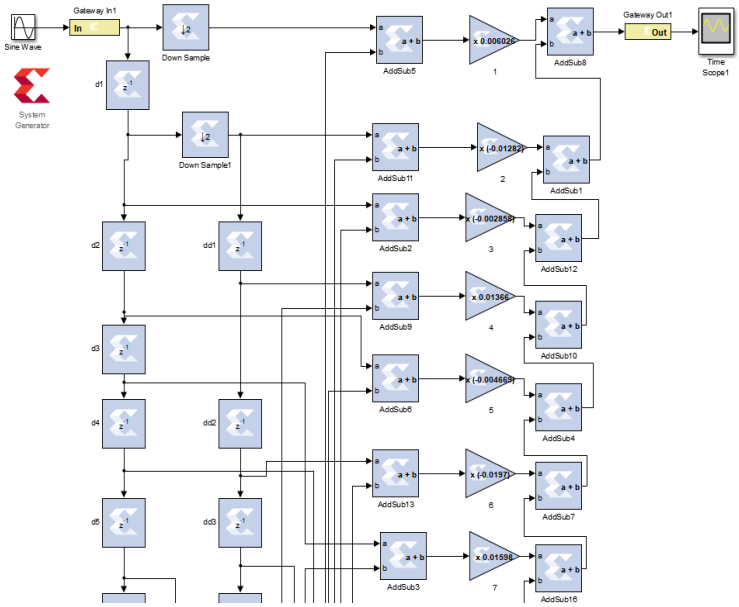
Ref. No.	Structures	Algorithm	Application	Device
10	SRC (low pass FIR filter)	N/A	Multi-rate signal processing, SDR	Virtex 6
12	SRC (L=2, M=2) & (L=3, M=2) Decimator Filter	N/A	EEG	Altera cyclone DE II
13	Multi-rate filter (half-band FIR & CIC)	Hamming Window	Multirate Signal Processing System	Altera develop
14	ASRC Newton & farrow structures	V-FDF N estimated (Belanger	Multi-standard digital front end	ASIC, FPGA (Virtex 6)
15	SRC using Cascaded multi-rate linear phase FIR(polyphase, folded, systolic) structures	Iterative procedure	Audio & Communication System	Altera Quartus II
16	ASRC Polyphase FIR filter	N/A	Digital Audio System	N/A
17	SRC with multiple parallel output phases	N/A	N/A	Virtex 7
Proposed	SRC with Direct, efficient, and polyphase structures	Remez	N/A	Artix 7

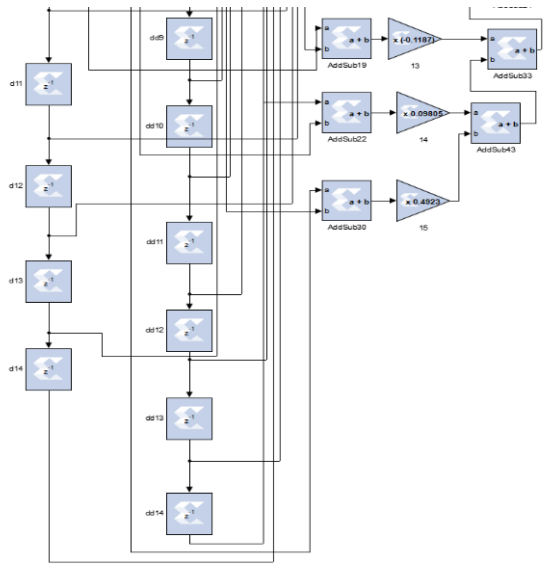

Gateway In1
Sine Wave
System Generator
Down Sample
Gateway Out1
Time Scope1
AddSub5
AddSub1
AddSub2
AddSub4
AddSub3
Down Sample1
Down Sample2
Down Sample3
Down Sample4
Down Sample5
Down Sample6
AddSub6
AddSub7
Down Sample24
Down Sample25
Down Sample26
Down Sample27
Down Sample28
Down Sample29
AddSub24
AddSub25
AddSub27
AddSub26
AddSub28
a)
b)


Gateway In1
Sine Wave
System Generator
Delay0
Delay1
Delay2
Delay3
AddSub16
Down Sample
AddSub5
Gateway Out1
Time Scope1
AddSub1
Down Sample16
Delay28
AddSub17
AddSub2
Down Sample2
Delay27
AddSub18
AddSub4
Down Sample3
Delay26
AddSub19



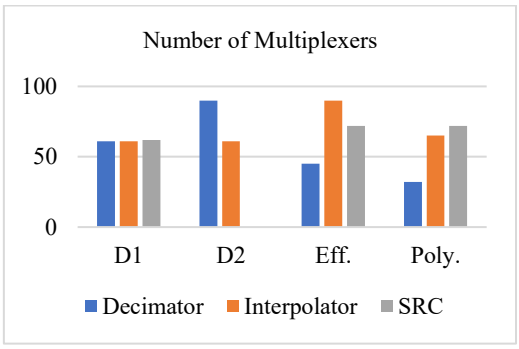
c) efficient



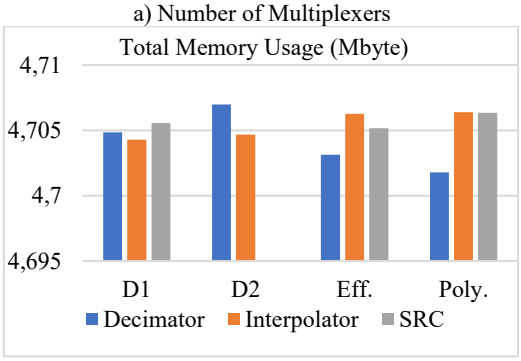


d) polyphase

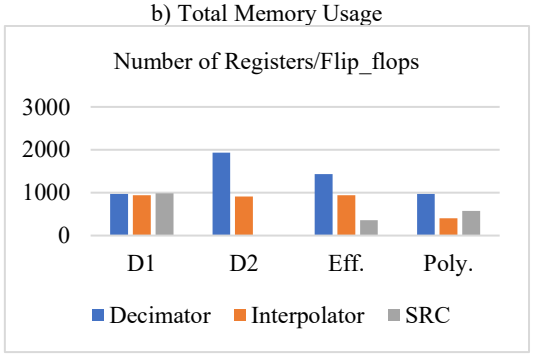
Fig. 6. Hardware Implementation of the Decimator



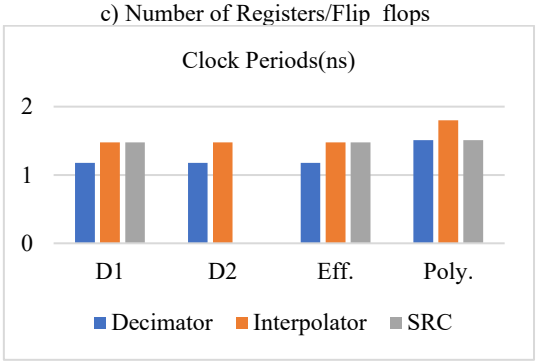
a) Number of Multiplexers



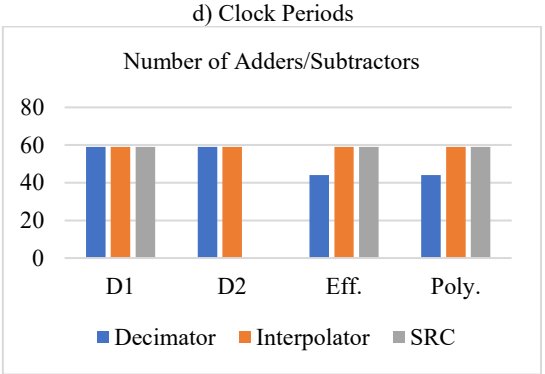
b) Total Memory Usage



c) Number of Registers/Flip_flops



d) Clock Periods



e) Number of Adders/Subtractors

Fig. 7. Device Utilization Summary

5. CONCLUSIONS

This paper focuses on digital filter design for multi-rate signal processing using different structures and its performance by synthesizing to Artix 7(XC7A100T-1CSG3324) FPGA. The polyphase method of designing the decimator and Interpolator was found to be more advantageous by providing an enhanced 67% ratio for operational speed and 46% ratio for hardware utilization compared to direct and linear phase structures for enhancing signal processing. As for the SRC, an efficient linear phase design is preferable, as it uses fewer resources and is easier to implement.

Source of funding: *This research received no external funding.*

Author contributions: *research concept and design, N.T.; Collection and/or assembly of data, R.W.; Data analysis and interpretation, R.W.; Writing the article, N.T.; Critical revision of the article, S.L.; Final approval of the article, S.L.*

Declaration of competing interest: *The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.*

REFERENCES

1. Lyons RG. Understanding digital signal processing, 3/E. Pearson Education India; 1997.
2. Sondermeyer J. Rational sample rate conversion with blackfin processors. Engineer to Engineer Note EE-183: 1-25.
3. Tan L, Jiang J. Digital signal processing: fundamentals and applications. Academic press; 2018 Oct.
4. Ingle VK, Proakis JG. Digital signal processing using MATLAB. CL-Engineering 3/E; 2011.
5. Zeineddine A, Nafkha A, Paquelet S, Moy C, Jezequel PY. Comprehensive survey of FIR-based sample rate conversion. Journal of Signal Processing Systems. 2021;93(1):113-25. https://doi.org/10.1007/s11265-020-01575-6.
6. Milic L. Multirate filtering for digital signal processing: MATLAB applications: MATLAB applications. IGI Global. 2009.
7. Gadawe NT, Hamad RW, Qaddoori SL. Realization of IIR digital filter structures for ecg denoising. Journal Européen des Systèmes Automatisés. 2024;57(2):599. https://doi.org/10.18280/jesa.570228.
8. Arnaldi LH. Implementation of a polyphase filter bank channelizer on a ZYNQ FPGA. In2020 Argentine Conference on Electronics (CAE) 2020: 57-62. https://doi.org/10.1109/CAE48787.2020.9046377.
9. Datta D, Mitra P, Dutta HS. FPGA implementation of high performance digital down converter for software defined radio. Microsystem Technologies. 2022;28(2):533-42. https://doi.org/10.1007/s00542-019-04579-w.
10. Pinjerla,S, Rao S, and Reddy P. Design of energy efficient and reconfigurable sample rate converter

using FPGA devices, Indonesian Journal of Electrical Engineering and Computer Science. 2024: 854-862. http://doi.org/10.11591/ijeecs.v36.i2.pp854-862.
11. Woods R, McAllister J, Lightbody G, Yi Y. FPGA-based implementation of signal processing systems. John Wiley & Sons; 2008.
12. Hassan M. Design and implementation of sampling rate conversion system for electroencephalogram (EEG) on FPGA device (Doctoral dissertation, Khulna University of Engineering & Technology (KUET), Khulna, Bangladesh).
13. Jing Q, Li Y, Tong J. Performance analysis of multi-rate signal processing digital filters on FPGA. EURASIP Journal on Wireless Communications and Networking. 2019;4(1):31. https://doi.org/10.1186/s13638-019-1349-9.
14. Zeineddine A, Paquelet S, Nafkha A, Jezequel PY, Moy C. Efficient arbitrary sample rate conversion for multi-standard digital front-ends. In2019 17th IEEE International New Circuits and Systems Conference (NEWCAS). 2019. http://doi.org/10.1109/NEWCAS44328.2019.8961230.
15. Vaithiyanathan D, Mariammal K, James BP. Performance Analysis of Multirate Filter Structures for Signal Processing Applications. In2023 3rd International Conference on Smart Generation Computing, Communication and Networking (SMART GENCON). 2023. http://doi.org/10.1109/SMARTGENCON60755.2023.10442272.
16. Ghosh J, Jatoth RK, Bhattacharjee J. Design and Implementation of a configurable Asynchronous Sample-Rate Converter for Digital Audio. In 2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT). 2023. http://doi.org/10.1109/ICCCNT56998.2023.10307751.
17. Shahabuddin S, Manninen P, Juntti M. A fractional sample rate converter with parallelized multiphase output: algorithm and FPGA implementation. Journal of Signal Processing Systems. 2022; 4(12):1459-69. https://doi.org/10.1007/s11265-022-01776-1.
18. Touil L, Hamdi A, Gassoumi I, Mtibaa A. Design of low-power structural FIR filter using data-driven clock gating and multibit flip-flops. Journal of Electrical and Computer Engineering. 2020;(1): 8108591. https://doi.org/10.1155/2020/8108591.
19. Proakis JG. Digital signal processing: principles, algorithms, and applications, 4/E. Pearson Education India. 2007.
20. Digilent ND. FPGA Board Reference Manual, Rev. B, February. 2017;27.
21. Digilient Inc. Nexys4 DDR™ FPGA Board Reference Manual.
22. Stearns SD, Hush DR. Digital signal processing with examples in MATLAB. CRC press; 2016.

Noor TALAL received the Master's Degree from university of Mosul. He is currently working in Electronics engineering department, Electronics engineering college, Ninevah university. His main research direct is digital signal processing with different applications.

e-mail:noor.gadawe@uoninevah.edu.iq

Rasha WALEED received the Master's Degree from university of Mosul. He is currently working in Electronics engineering department, Electronics engineering college, Ninevah university. His main research direct is digital signal processing with different applications.

e-mail: rasha.hamad@uoninevah.edu.iq

Sahar LAZIM received the PHD degree from university of Mosul. He is currently working in Electronics engineering department, Electronics engineering college, Ninevah university. His current research interests include computer networks, embedded system, and network security.

e-mail:sahar.qaddoori@uoninevah.edu.iq