National Journal of Multidisciplinary Research and Development
ISSN: 2455-9040
Impact Factor: RJIF 5.22
www.nationaljournals.com
Volume 3; Issue 1; January 2018; Page No. 1255-1259

Interpolation for multirate system application

Hansa Rani Gupta
Assistant Professor, Department of Electrical Engineering, IET, Alwar, Rajasthan, India

Abstract
In this paper we put forward the literature review related to the multirate systems. One of the main objective of this survey paper to find the comparison between various multirate systems with software defined radio. Another objective is to provide the novel approach in order to help readers with choosing the most appropriate SDR. This technique has been used to take an optimal advantage of CIC interpolator because the use of embedded LUTs not only increase the speed but also saves the resource of the target device FPGA. The proposed interpolator has been designed using half band polyphase FIR structure with Matlab, simulated with Modelsim XE, synthesized with Xilinx Synthesis Tools (XST) and implemented on Spartan-3E based 3s500efg320-4 FPGA device.

Keywords: CIC, SDR, FIR, LUT, XST

Introduction
The extensive use of digital representation of signals for transmission and storage has created challenges in the area of digital signal processing. The applications of digital FIR filter and up/down (interpolator/decimator) sampling techniques are found everywhere in modem electronic products. For every electronic product, lower circuit complication is always an important design target since it reduces the cost. There are many applications where the sampling rate must be changed. Up sampler and down sampler are used to change the sampling rate of digital signal in multi rate DSP systems. This rate conversion requirement leads to creation of undesired signals associated with aliasing and imaging errors. So some kind of filter should be placed to attenuate these errors.

Interpolators
Interpolators and decimators are utilized to increase or decrease the sampling rate. Up sampler is basic sampling rate alteration device used to increase the sampling rate by an integer factor. An up-sampler with an up-sampling factor L, where L is a positive integer, develops an output sequence $x_u[n]$ with a sampling rate that is L times larger than that of the input sequence x[n]. The up sampler is shown in Figure1. Up-sampling operation is implemented by inserting equidistant zero-valued samples between two consecutive samples of x[*n*].

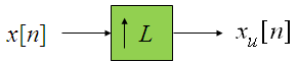

Fig 1: Up Sampler

The input and output relation of up sampler can be expressed as

$$x_u[n] = \begin{cases} x[n/L], & n = 0, \pm L, \pm 2L, \cdots \\ 0, & \text{otherwise} \end{cases}$$

The zero-valued samples inserted by the up-sampler are replaced with appropriate nonzero values using some type of filtering process called interpolation [1].

Software defined radios
Software defined radios is a kind of device that wirelessly transmits or receives signals in the radio frequency (RF) part of the electromagnetic spectrum to facilitate the transfer of information. In today's world, radios exist in a multitude of items such as cell phones, computers, car door openers, vehicles, and televisions.
In other words it can be defined as radio in which some or all of the physical layer functions are software defined. SDR enables the family of radio “products” to be implemented using common platform architecture, allowing new products to be more quickly introduced into the market. Software to be reused across radio "products", reducing development costs dramatically. Over-the-air or other remote reprogramming, allowing "bug fixes" to occur while a radio is in service, thus reducing the time and costs associated with operation and maintenance.

Cascadeed Interpolator Comb Filter
In digital signal processing, it is an optimized class of finite impulse response (FIR) filter combined with an interpolator or decimator.

A CIC filter consists of one or more integrator and comb filter pairs. In the case of an interpolating CIC, the input signal is fed through one or more cascaded integrators, then a up-sampler, followed by one or more comb sections (equal in number to the number of integrators).

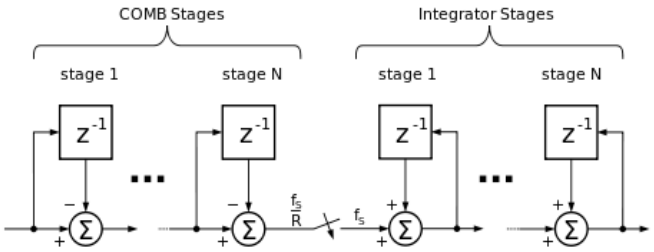

Fig 2: CIC

Comparison between various multirate systems

In the paper, eight independent multi-rate signal interpolators, with real-time change of rate capability, were implemented on a FPGA. CIC filter and the respective compensation filter performs a fixed rate change of 4 and implemented as a 129 taps finite impulse response (FIR) filter. The FIR filter coefficients were attained from the MATLAB simulation, based on the inverse sinc (x) function. In this, CIC was designed to have six stages, a differential delay of 1, a variable rate change factor ranging from 10 up to 10,000 and fixed sampling rate of 40 MSPS. These signals are subsequently converted to the analog domain by 16 bit digital-to-analog converters.

The CIC interpolator magnitude frequency response with six stages, differential delay of 1 and interpolating factor ranging from 10 up to 10 000 which is multiplied by 4, due to the fixed rate change executed by the compensation filter shown in figure 3.

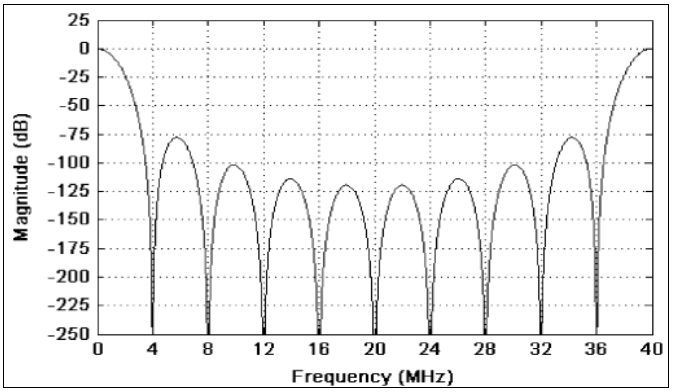

Fig 3: CIC interpolator magnitude frequency response for R = 10 which corresponds to a 1 MSPS JET signal sampling rate and therefore to a 500 kHz signal bandwidth.

To compensate the passband magnitude attenuation of the CIC filter, a finite impulse response (FIR) interpolator, with fixed rate change of 4 and an inverse sinc (x) magnitude frequency response in the passband, was simulated on MATLAB®. A set of 129 coefficients were obtained for the FIR filter implementation. In Fig. 4, the magnitude frequency response of the compensation filter can be seen.

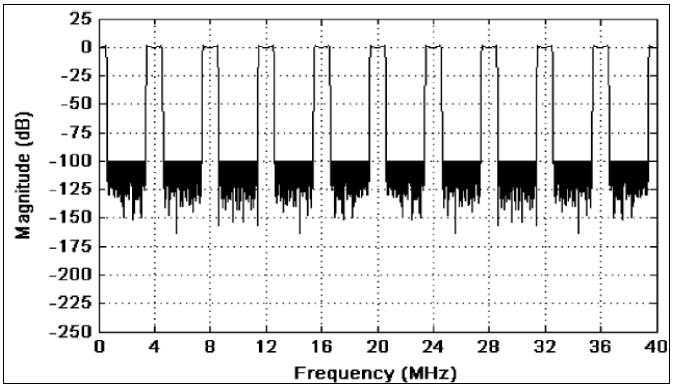

Fig 4: Magnitude frequency response of the compensation filter for a 1 MSPS signal sampling rate.

Fig. 5 shows the magnitude frequency response of the resulting filter and was obtained simulating the cascade of the FIR compensation filter with the CIC interpolator.

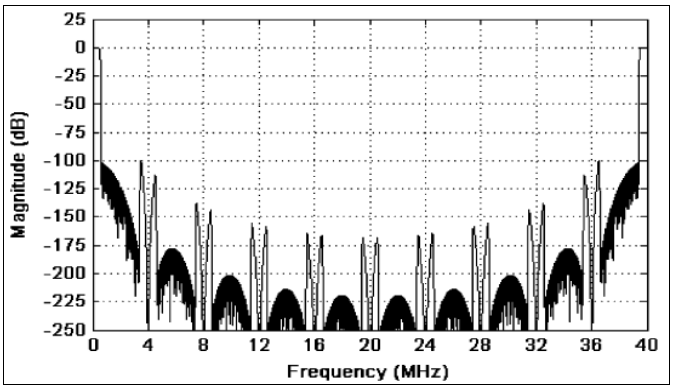

Fig 5: Resulting filter magnitude frequency response for a CIC with interpolation factor R = 10.

The magnitude frequency response of a seventh-order Chebyshev type II is presented in Fig.6 and was obtained from the simulation of a dedicated program for passive filter implementations.

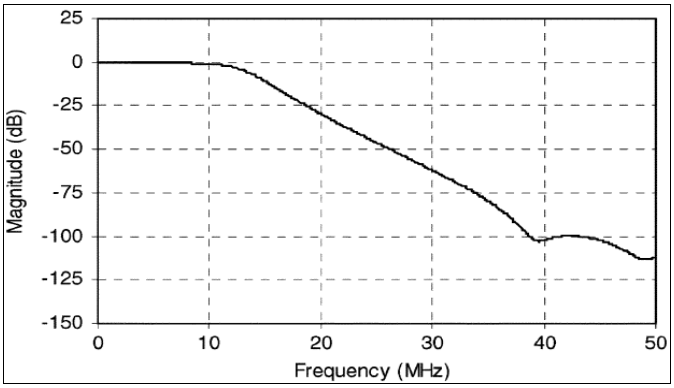

Fig 6: Reconstruction filter magnitude frequency response.

The first imaging appears at 40 MHz and a stop band attenuation of 100 dB, starting near 39 MHz, guarantees a free imaging signal frequency spectrum above that stop band shown in fig 7.

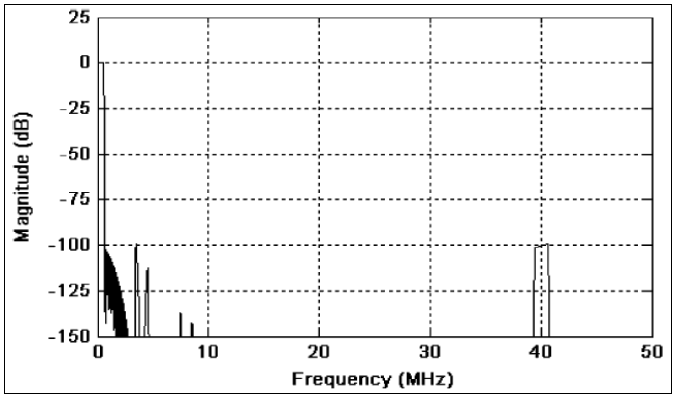

Fig 7: Channel output magnitude frequency response (simulated).

The magnitude frequency response of the outputs is presented in Fig. 8 (the tail is due to differences between the simulation and the implementation in the FPGA).

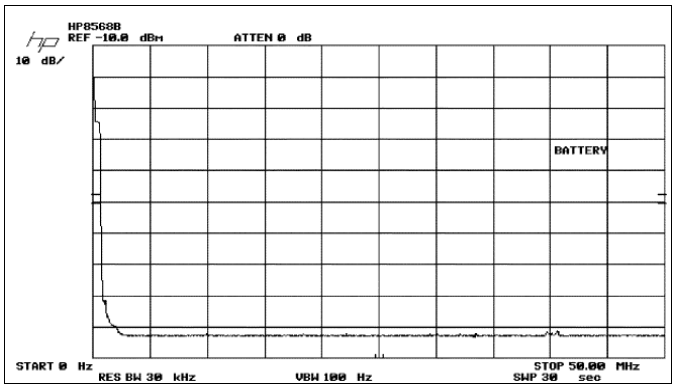

Fig 8: Channel output magnitude frequency response (measured).

Fig. 9 shows a 15.625 kHz sine wave spectrum resulting in a total harmonic distortion of 71.6 dB and a signal-to-noise ratio of 103.1 dB (1 Hz). The effective number of bits is 11.6 bits [2].

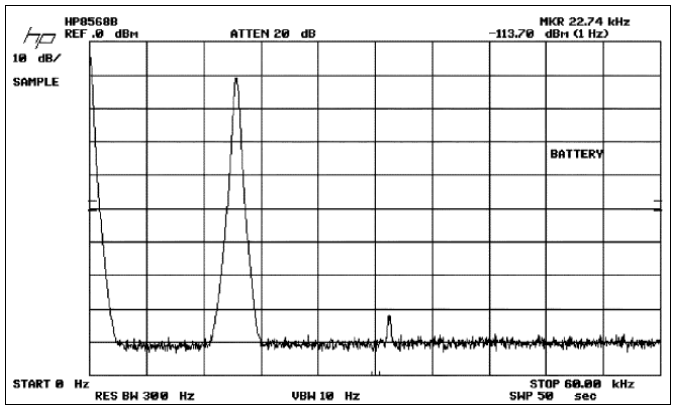

Fig 9: Spectrum of a 15.625 kHz sine wave (measured).

In this proposed work fully pipelined 3 stages CIC interpolator is designed using MATLAB and Xilinx Accel DSP by taking R as 8 and M as 2. First, the floating point output of m-code file is verified and analyzed as shown in Fig 10.
Then the equivalent fixed point file is generated and verified by AccelDSP whose output is shown in Fig 11. The red plot shows the input sequence, green plot shows the ideal response and blue plot is the output from CIC decimator.
A 3 stage interpolator is designed to accomplish three things here. First, we have slowed down half of the filter and therefore increased efficiency. Second, we have reduced the number of delay elements needed in the comb sections. Third, and most important, the integrator and comb structure are now independent of the rate change. This means we can design a CIC filter with a programmable rate change and keep the same filtering structure.
The first step in design flow is to develop m-code for CIC interpolator with required specifications using MATLAB. First, the floating point output of m-code file is verified and analyzed as shown in Fig 10.

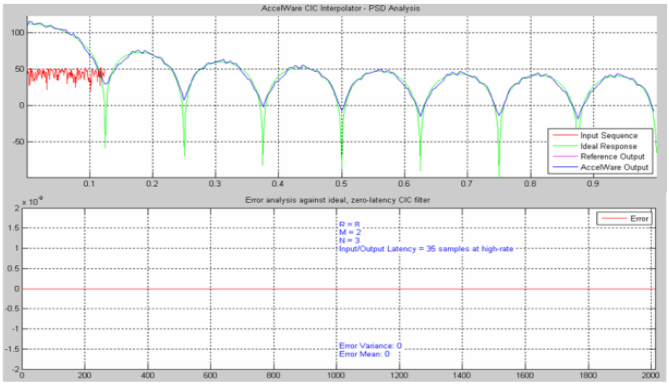

Fig 10: Floating Point Output of CIC Decimator

Then the equivalent fixed point file is generated and verified by AccelDSP whose output is shown in Fig5. The red plot shows the input sequence, green plot shows the ideal response and blue plot is the output from CIC decimator.

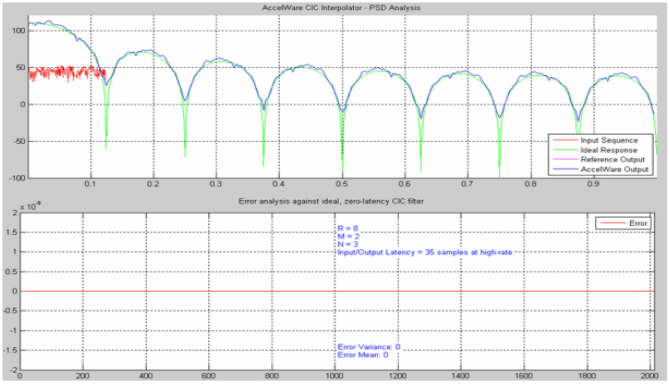

Fig 11: Fixed Point Output of CIC Decimator

As shown in table 1, the proposed LUT based design can work at an estimated frequency of 276.6 MHz by using considerable less resources of target FPGA [3].

Table 1: Transposed Form Performance Evaluation

Clock Name	Requested Frequency	Estimated Frequency	Estimated Period	Max Throughput	Input Sampling
Clock	100.0 MHz	276.6 MHz	3.6150 ns	1	276.625 MSPS

In this proposed work DSP 48E slices are replaced with multipliers of target FPGA device by preserving the required specification. GSM DUC design has been optimized by using

the concept of polyphase decomposition for efficient mapping on multiplier based Virtex II Pro FPGA. The speed performance has been enhanced by using MAC algorithm which is implemented with embedded multipliers of target FPGA. The first stage FIR pulse shaping filter with order 31, second stage FIR compensating filter with order 10 and third stage CIC filter have been designed and cascaded together using MATLAB [12] to develop final DUC as shown in Fig.12.

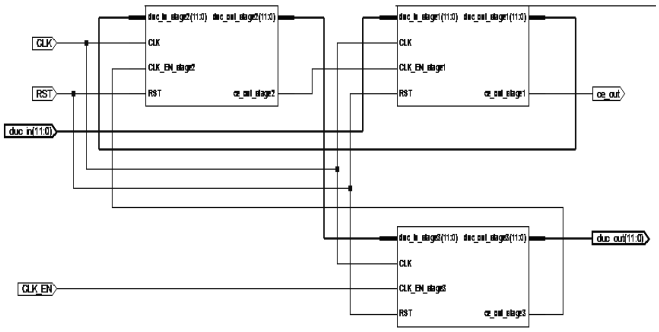

Fig 12: GSM DUC Design

First two stages have been implemented using multiplier based MAC algorithm and third stage has been implemented using multiplier less technique. The developed design has used 22 embedded multipliers for implementing first to stages.

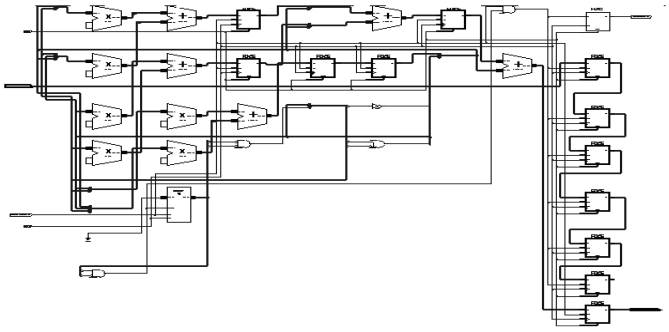

Fig 13: MAC Algorithm based DUC Stage

First stage has been developed using 16 multipliers to process 32 taps and second stage has been developed using 6 multipliers to process 11 taps. The MAC algorithm based structure of stage 2 has been shown in Fig.12.

The VHDL code of proposed DUC has been simulated using Modelsim simulator whose output has been shown in Fig.14.

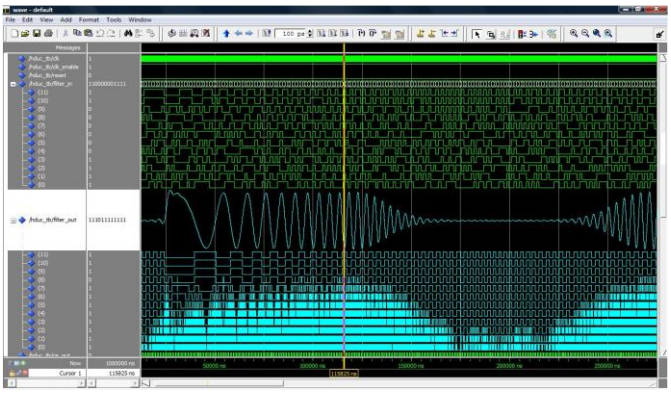

Fig 14: DUC Simulated Output

Table 2: GSM DUC Resource Utilization

Logic Utilization	Used	Available
# of Slices	539	13696
# of LUTs	621	27392
# of DSP48Es	-	-
# of MULTs	22	136

Table 3: Resource Utilization Comparison

Logic Utilization	Optimized DUC	DUC [9]
# of Slices	557	965
# of LUTs	621	1317
# of DSP48Es	-	43
# of MULTs	22	-

The optimized DUC can operate at a maximum frequency of 145.5 MHz as compared to 125 MHz in case of existing GSM DUC design by consuming 557 Slices, 621 LUTs and 22 Multipliers. The resource utilization of optimized design has been compared design of as shown in Table 3 [4].

Conclusion

In this review paper, the implementation of the CIC filter interpolator did not consume high amounts of resources of the FPGA—namely the multiplier blocks—which were used to build the FIR compensation filter. Nevertheless, 90% of the FPGA resources are used due to the number of channels. At the beginning of the FPGA design, it was intended to scale the CIC output automatically. A Xilinx AccelDSP based approach is presented to minimize the time to market factor. The proposed fully pipelined CIC interpolator filter is designed by using embedded LUTs of target device. This show enhanced performance in terms of speed and area utilization and can operate at an estimated frequency of 276.6 MHz by consuming considerably less resources to provide cost effective solution for wireless based SDR applications. Area efficiency has been achieved by limiting the filter order and speed has been enhanced by utilizing optimal number of embedded multipliers of target FPGA. DSP 48 slices have been replaced with multipliers to improve cost effectiveness. The developed DUC can operate at a maximum frequency of 145.5 MHz by consuming 557 Slices, 621 LUTs and 22 Multipliers.

References

1. Rajesh Mehra, Ravinder Kaur. “Reconfigurable Area & Speed Efficient Interpolator using DALUT for Wireless Communication Systems” Advances in Networks and Communications, Communication in Computer and Information Science 2016; 132:117-125.
2. Batista AJN, Alves D, Cruz N, Sousa J, Varandas CAF, Joffrin E *et al.* Farthing,“an FPGA-based multi-rate interpolator with real-time rate change for a JET test-

bench system", IEEE Transactions on nuclear science 2016; 53(3):756-760
3. Rajesh Mehra, Sumana Chatterjee. "FPGA Based Design of CIC Interpolator using Embedded LUT Structure", ISP Journal of Electronics Engineering 2017; 1(1):1-4
4. Rajesh Mehra. "Reconfigurable GSM Digital up Converter for Multi-rate Systems", Recent Advances in Finite Differences and Applied & Computational Mathematics, 2017, 19-22
5. Santhosh YN, Namita Palacha, Cyril Prasanna Raj. "Design and VLSI Implementation of interpolators/decimators for DUC/DDC", Third International Conference on Emerging Trends in Engineering and Technology, 2010, 756-759
6. Rajesh Mehra, Pattnaik SS. "Reconfigurable Design of GSM Digital down Converter for Enhanced Resource Utilization", International Journal of Computer Applications 2013; 57(11):41-47
7. "8-Channel, 12-Bit, 40MSPS Analog-to-Digital Converter with Serial LVDS Interface", Texas Instrument, SBAS293F–JANUARY 2004–REVISED, 2009, 1-31.
8. Batista AJN. "A test-bench for the real-time project phase 2 of JET", Fusion Engineering and Design 2004; 71:83-88
9. Smith SW. "The Scientist and Engineer's Guide to Digital Signal Processing", 2nd ed. San Diego, CA: Analog Devices, 1999.
10. Hogenauer EB. "An economical class of digital filters for decimation and interpolation," IEEE Trans. Acoust., Speech, Signal Process 1981; 29(2):155-162.
11. ShyhJye Jou, Kai-Yuan Jheng, Hsiao-Yun Chen, An-Yeu Wu. "Multiplierless Multirate Decimator I Interpolator Module Generator", IEEE Asia-Pacific Conference on Advanced System Integrated Circuits, 2004, 58-61.
12. Shahriar Emami "New Methods for Computing Interpolation and Decimation Using Polyphase Decomposition", IEEE Transactions on education, 1999; 42(4):311-314.
13. Sheikh F, Masud S. "Efficient sample rate conversion for multi-standard software defined radios", IEEE Int. Conf. on Acoustics, Speech and Signal Processing, HI, April 2017, II-329 – II-332.
14. Khouja N, Grati K, Ghazel A, Le Gal B. "Design and implementation of a reconfigurable decimation and channel selection filter for GSM and UMTS radio standards," IEEE Wireless Communications and Networking Conference. WCNC 2009, 1-6, 5-8.
15. Schelle G, Fifield J, Grunwald D. "A software defined radio application utilizing modern FPGAs and NoC interconnects," International Conference on Field Programmable Logic and Applications, 2007, 177–182.
16. Shahana TK, Jose BR, James RK, Jacob KP, Sasi S. "RNS based programmable multi-mode decimation filter for WCDMA and WiMAX," IEEE Vehicular Technology Conference, 2008, 1831-1835.