Indonesian Journal of Electrical Engineering and Computer Science
Vol. 36, No. 2, November 2024, pp. 854~862
ISSN: 2502-4752, DOI: 10.11591/ijeecs.v36.i2.pp854-862

Design of energy efficient and reconfigurable sample rate converter using FPGA devices

Swetha Pinjerla[1], Surampudi Srinivasa Rao[1], Puttha Chandrasekhar Reddy[2]
[1]Department of Electronics and Communication Engineering, Malla Reddy College of Engineering and Technology, JNTUH, Secunderabad, India
[2]Department of Electronics and Communication Engineering, JNTU College of Engineering Hyderabad, Telangana, India

Article Info

Article history:

Received Feb 1, 2024
Revised Jul 4, 2024
Accepted Jul 14, 2024

Keywords:

Aliasing effect
Analog to digital converter
Field programmable gate arrays
Lowpass filter
Multi-rate signal processing
Sample rate conversion

ABSTRACT

The technique of sampling rate conversion is frequently employed in various fields. A discrete time-varying filter, as well as a sample skip or sample duplicate operation, are required for the most general instance of an irrational and time-variable conversion factor. A wide band of signals is employed in a communication system, especially in specific situations where data must be transferred directly. A broadband sample rate converter with changeable filter parameters is necessary in such cases. Sample rate conversion is a communication system technology that accepts a band-limited high sample rate modulated signal and uses filtering to retrieve the original message signal. In this work, an energy-efficient implementation of a reconfigurable field programmable gate arrays (FPGA) architecture for a sample rate converter is proposed. In applications such as multi-rate signal processing and the construction of channelized receivers, sample rate conversion is used. In this work, a new FPGA based design is proposed to perform multiple sample rate conversion for various data transmission protocols such as Wi-Fi, ZigBee and Bluetooth. A lowpass filter with a 2.45 GHz filter with the minimum number of taps is used to avoid the aliasing effect. Xilinx synthesis tools are used to estimate hardware resource utilization and speed analyses. XC6VCX240t-2FF484 FPGA achieves 15% hardware resource occupancy at a maximum clock speed of 133 MHz.

Corresponding Author:

Swetha Pinjerla
Department of Electronics and Communication Engineering
Malla Reddy College of Engineering and Technology, JNTUH
Secunderabad-500100, Telangana, India
Email: pswetha0823@gmail.com

1. INTRODUCTION

The sample rate converter (SRC) is employed in modern communication systems to increase the overall efficacy of the system. Multirate systems are those that use various sample rates. Sample rate conversion is the process of converting a signal from one rate to another needed rate, and multi-rate digital signal processing systems are used in the processing of digital signals that use various sampling rates. In a digital signal processing (DSP) system, the SRC block computes samples at a new rate utilizing data from different sampling rates. In communication systems, sample rate converters have become a key component. Among these are the half-band (HB) filter, the cascaded-integrated comb (CIC) filter, the polyphase filter, the Farrow-based variable fractional delay (VFD) filter, and the LUT-based variable digital filter (VDF) [1]. Field programmable gate arrays (FPGAs) have grown into a flexible platform for digital systems. FPGA-based SRC devices offer a lot of frequency and phase flexibility, and they perform well in high-precision calculations [2].

The SRC factor determines the capacity of a polyphase filter's data and coefficient RAMs. RAM resources may exceed the capability of hardware devices such as FPGAs if SRC is conducted simply using a polyphase filter.

Hogenauer filters, also known as cascaded integrated comb filters, are multi-rate filters used in digital systems to achieve significant sample rate variations. These are structures without multipliers, consisting simply of adders, subtractors, and registers [3]. Processing in parallel for multiband orthogonal frequency division multiplexing (OFDM)-based software-defined radio (SDR) systems, the SRC structure will enable high-speed data transfer. At this time, digital communication is quite popular. The frequency range of interest may be pushed down the spectrum to a baseband signal at 100 Hz using a sample rate converter, making signal processing easier [4]. Multi-rate systems [5] are systems that have numerous sample rates. Interpolators with an integer factor L and decimators with an integer factor M are the main building pieces of a multi-rate system. When these two blocks are combined, a system that alters the sample rate is created by a rational factor M/L [6]. It should first suppress the up-sampling-induced picture spectra, and then it should avoid the aliasing effects that occur after down-sampling [7]. Wu *et al.* [8] designed Xilinx ISE to create a digital down converter (DDC) that has features such as across clock region and FIFO interface. The sum and difference frequency components are obtained by mixing or multiplying, this digitized stream of samples with a digital cosine for the phase channel and a digitized sine for the quadrature channel at the DDC's first step. Similar digital filters may be used to filter phase and quadrature signals, which is not an issue with FPGA digital filters. A digital filter's frequency response is normally symmetrical around 0.5Fs [8].

Sahukar *et al.* [9] proposed a method with 8 zero crossing long since function-based rate converters and fractional rate converters save space (FRC). The digital-to-analog converter (DAC) uses a low pass filter to accept just the spectrum around zero frequency and reject all other spectrum pictures. To preserve rectangular window-type frequency domain features in an ideal low pass filter, indefinite length coefficients constituting a sinc function are required. To generate the virtual DAC output, these filter coefficients are convolutional with digital sample values of the weighted sinc functions at each sample instant with a weight equal to the sample value [9], [10]. Zhang and Su [11] proposed DDC model is based on orthogonal mixing. As a local carrier, the DDS IP CORE generates digitalized sine and cosine signals. The system clock is 100 MHz, the spurious-free dynamic range (SFDR) is 96 dB, and the frequency resolution is 0.0233 Hz in this design. The complex multiplier is thought of as a mixer that combines two complex signals. The output is reduced to 20 bits. The CIC compiler core is used to specify the clock frequency and input sample frequency. It's a five-stage CIC filter with a sample-per-stage differential delay in this design [11].

Zhang and Zhu [12] proposed a 1D interpolator for Reed-Solomon. With polynomial complexity, algebraic soft-decision decoding (ASD) of RS codes can yield significant coding gain. Iterative bit-level generalized minimum distance (BGMD) decoding, among feasible ASD algorithms, can yield equivalent or larger coding gain with less complexity. The interpolation algorithm's intrinsic serial nature limits the maximum attainable speed of this phase. In addition, effective very-large-scale integration (VLSI) architectures for implementing the described system are proposed [12]. Tietche *et al.* [13] proposed an arbitrary ratio resampler for SDR applications based on an FPGA that allows the designer to handle a spurious-free dynamic range (SFDR) while giving a simple solution that requires no extra clock, making it suitable for FPGA clock-limited systems. The SFDR-based technique produces resource use predictions that are very accurate when compared to real resource utilization statistics acquired after the FPGA circuit has been designed, making it a useful tool for design prototyping [13].

Dolecek and Torres [14] proposed using the factorization approach, an improved cascaded integrator comb filter implementation for SRC of several wireless protocols. Boukhtache *et al.* [15] proposed a highly efficient bicubic interpolator for 1D and 2D applications. Because of its high quality, bicubic interpolation is commonly utilized in real-time image processing systems. Otunniyi and Myburgh [16] proposed a low-complexity filter designed using a hybrid Farrow algorithm that combines a modulated Farrow filter with a frequency response interpolated coefficient decimated masking filter. A design example illustrates that the HFarrow filter bank reduced multipliers by 50%, 70%, and 64% compared to non-uniform modified discrete fourier transform filter bank (NU MDFT FB), cosine modulated filter bank (CD FB), and iterative combinatorial design method (ICDM) filter algorithms [16]. Sinha and Kumar [17] proposed the fundamental structure of a CIC filter and some of the crucial parameters. As a result, the emphasis is placed on the incorporation of the CIC filter into the decimator and the interpolator. This investigates the possibility of developing a method for enhancing the qualities possessed by this filter and highlights several issues that are connected to it [17].

From the above discussion, it is very clear that previously several works have been proposed to perform FPGA implementation for sample rate conversion architecture. The majority of the works are focused on reducing power consumption and area. Also, more implementation is performed based on FPGA and ASIC devices.

This work proposes an energy-efficient FPGA architecture for sample rate converter. To prevent aliasing effect, a lowpass filter with a 2.45 GHz cutoff with minimum number of taps to reduce area consumption. The research focuses on an area-optimized dynamic FPGA implementation for low-power multirate sample rate converter. Xilinx FPGA synthesis tools are used to evaluate the hardware resource utilization and speed analysis. XC6VCX240t-2FF484 FPGA shows just 15% area occupancy and a maximum clock speed of 133 MHz.

The primary goal and purpose of this work is as follows:

- To minimize area and power consumption of multiple sample rate converter in a single chip with maximum speed for various high speed digital communication related applications.
- To make perfect synchronization with various communication protocols such as Wi-Fi, ZigBee and Bluetooth.

2. METHOD

In this method the input data is sent to the data buffer which is controlled by the control logic. Then this buffered data is sent to the (DFF) D Flip Flop, here the enable signal is used to enable the Flip Flop. From this Flip Flop, the data is sent to the DMUX, and a selection line of DMUX will select the signal for upsampling or downsampling. Finally, MUX at the end gives the selected output signal. The simulation of the sample rate conversion algorithm is done with the MATLAB 2020a tool. The FPGA implementation is done with Verilog HDL and timing is verified by using Modelsim software. At a lower sampling rate, an FIR filter is used to extract the in-phase and quadrature components of the baseband signal. The design is implemented on a Virtex-6 XC6VCX240t-2FF484 FPGA.

2.1. Block diagram

The control logic present in the architecture will control the address generation and flow of data from the device. The rate selection line available in this architecture will be used to decide which kind of bit rate conversion is to be performed. The proposed sample rate converter uses an input buffer to perform signal catch. To perform the downsampling process a simple DFF is used with the required reduced clock rate. The upsampling process is performed by generating a new sample from two intermediate samples which is stored in the shift registers. A central multiplexer is used to select the required output performed by the sample rate converter. The output is controlled by a single-bit selection line input value given by the user. The input to the downsampler or upsampler is controlled by a demux block. Figure 1 shows the block diagram of the proposed multiple-sample rate converter technique.

Figure 2 shows the FIR filter used to remove the unwanted noise which is present in the output of the sample rate converter. Here N=8 is considered to perform the implementation of a low pass FIR filter with the cutoff frequency of 2.5 GHz. The passband frequency depends upon the coefficient used in the filter. Here h_N represents the n_{th} coefficient used to perform the filtering process. The frequency response of the 2.5 GHz low pass filter is given in Figure 3.

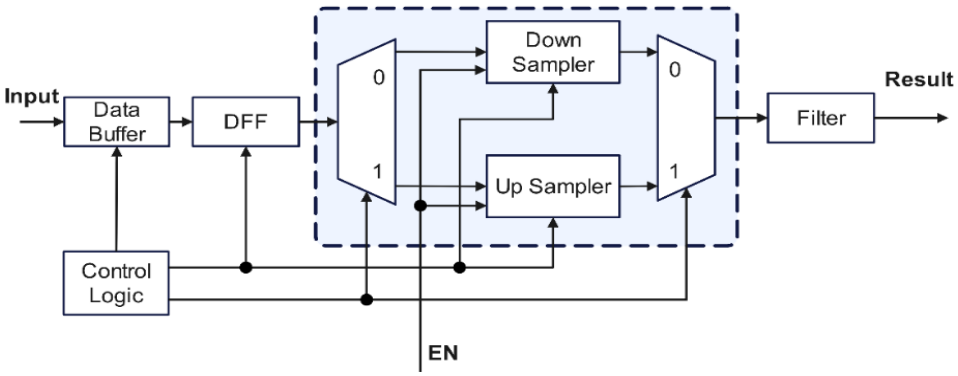

Figure 1. Block diagram for the proposed method

Table 1 shows the specification of the filter used to avoid aliasing problems in the sample rate converter. A passband frequency of 2.1 GHz is used and 2.4 GHz as the stopband to control the noise. To attenuate the unwanted frequency 80 dB attenuation factor is used. Which can lead to a noisy signal with zero

amplitude. The passband ripple value is set to 0.1 to control the passband amplitude. The sampling frequency of the input signal is set to 4.8 GHz to avoid the aliasing effect.

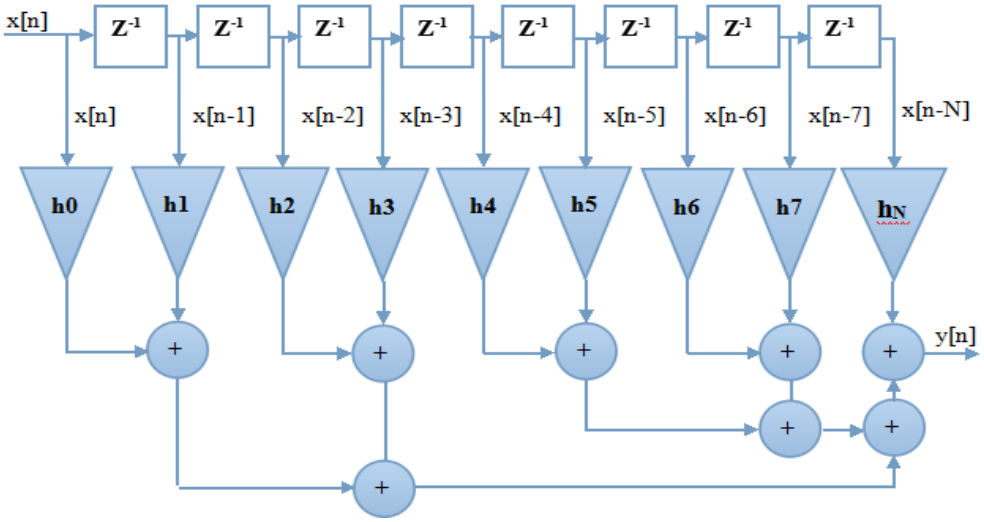

Figure 2. FIR filter used for data smoothing

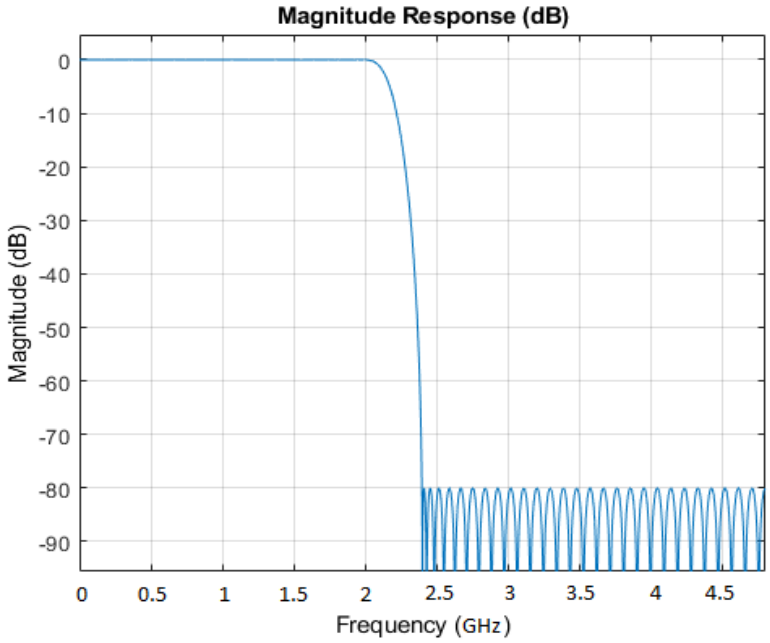

Figure 3. Frequency response of 2.5G Hz low pass filter

Table 1. FIR filter specification

Sl.No	Specification	Value
1	Passband frequency	2.1 GHz
2	Stopband frequency	2.4 GHz
3	Stopband attenuation	-80 dB
4	Passband gain	0 dB
5	Passband ripple	0.1
6	Sampling frequency	4.8 GHz
7	Filter order	16

2.2. Relations between input and output samples in time domain

The time-domain relationships between the four sequences included in this signal may be expressed sequentially using a typical sampling-rate converter. Where d_n is the coefficient of the transfer function $H(z)$. The input sample $l[m]$ will exist only at the integer-valued time instances k. Only these terms (Mk-n)/L contribute to the output value y[x] in the preceding equation. As demonstrated in this diagram, sampling rate conversion by a rational factor L/M entails upsampling the signal by a factor of L, filtering the resultant up the sampled sequence with a transfer function H(z), and lastly downsampling the filtered signal by a factor of M.

$$w[\beta] = \sum_{n=0}^{K} d_n l[\beta - n] \quad (1)$$

$$Y[x] = w[Mk] \quad (2)$$

$$y[x] = \sum_{n=0}^{K} d_n l[\frac{Mk-n}{L}] \quad (3)$$

2.3. Up sampling

Up sampling is the process of adding zero-valued samples between original samples to increase the sampling rate. (It's sometimes called "zero-stuffing.") This method of up sampling inserts undesired spectral images that are focused on multiples of the sampling rate into the original signal. The down sampler is comparable to the up-structure sampler. Master control keeps an eye on the FIFO's reading to avoid underflow. Using an "up sampling map," master control ensures that the ROM indices are incremented and the FIFO is read at the correct times. Of course, the binary map for up-sampling and down-sampling is not the same. For down sampling, the map is synchronized on the input timings; for up sampling, it is synchronized on the output timings. For up-sampling, ROM indices must be increased when a map element is set to 1. It behaves identically to a simple D Flip-Flop register in all other respects, which is useful when two or more output samples are computed from the same input samples [18].

2.4. Down sampling

Down sampling is the process of lowering a signal's sampling rate. By selecting one out of N samples, the down sample decreases the sampling rate of the input AOs by an integer factor. The raw data is not subjected to any anti-aliasing filtering [19]. A down sampler cab is used to make communication between different communications standards such as ZigBee, and Bluetooth. Wi-Fi uses a bandwidth of 20 to 40 MHz frequency. That can be down sampled to 8 MHz which is the sampling frequency of ZigBee. Similarly, ZigBee can be down sampled to 8-96 KHz.

2.5. FPGA implementation

A Xilinx FPGA chip was used to test the proposed SRC architecture. Each block of the proposed SRC design is coded in hardware description language in the Xilinx simulator with ISE Architecture Suite 14.7. (HDL). The synthesized bitstream was downloaded at speed grade 3 on the XC6VCX240t-2FF484 target device. This configuration limits the possible area for low-cost implementation at a higher rate than other comparable hardware techniques.

Table 2 shows the implementation specification for the sample rate converter. The conversion order of the down converter is from Wi-Fi to Bluetooth and for the upconverter, it is vice versa. In this work, an 8-bit input is used to represent the inputs and output which can reduce the hardware burden and computational cost. A lowpass FIR filter is used to perform smoothing for up sampled output. 16 tap filter is used to perform the low-pass filtering process. A 32 kB buffer is used on the input side to perform the signal-catching process.

Table 2. Sample rate convertor specification

Sl.No	Specification	Value
1	Down conversion order	1. Wi-Fi 2. ZigBee 3. Bluetooth
2	Up-conversion order	1. Bluetooth 2. ZigBee 3. Wi-Fi
3	Input size	8-Bit
4	Data smoothing	Low pass FIR filter
5	Buffer size	32kB
6	Filter order	16-Tap
7	Filter type	Cascaded FIR structure
8	Conversions	Up sampling and down sampling
9	Conversion mode	Single

3. RESULT AND DISCUSSION

3.1. Resource utilization on FPGA device

A fixed number of programmable logic, routing, I/O, and memory resources are available on each FPGA. These resources are used by the compiler to implement code on the FPGA. On an FPGA, programmable resources are spaced at regular intervals over the device, allowing for short pathways between them [20]-[23]. An FPGA's programmable resources are spaced at regular intervals over the device, allowing for short pathways between them. Short routes between resources lower the amount of time it takes for programs to run on the FPGA target [24]-[26]. Many concurrent processes can operate on the same device at the same time, reducing resource conflicts, thanks to the multiple instances of each resource on an FPGA.

Table 3 displays the device utilization table with various logic such as the number of slice registers, LUTs, IOBs, Block RAM, and DSP. In this design, slice registers account for 9% of the total, and LUTs account for 48%. This design uses 14279 LUTs out of a total of 28,800, and 2,759 slice registers out of a total of 28,800. All the resources available with their numerical count and percentage of utilization summary are represented in the graph. The resource utilization comparison with previous techniques, as presented in Table 4, reveals distinctive characteristics among different methods. In the context of slice registers, method [22] does not provide specific data, while [14] and [25] demonstrate 3,337 and 3,278 registers, respectively. The number of LUTs for [22] is unspecified, whereas [14] and [25] show 5,898 and 4,594 LUTs, respectively. Regarding I/O blocks, [22] does not disclose information, [14] has no IOBs, and [25] similarly lacks details. In contrast, the proposed work utilizes 20 IOBs, indicating a specific allocation for input and output functionality. The number of Block RAM/FIFO instances is specified only for method [22], with 4 instances. [14] and [25] do not provide data in this category. The proposed work employs 2 Block RAM/FIFO instances, suggesting an efficient utilization of this resource. In terms of DSP units, method [22] utilizes 82 DSPs, while [14] and [25] have no specified information. The proposed work employs 16 DSP units, showcasing a balanced and optimized use of digital signal processing resources.

Table 3. Resource utilization

Logic utilization	Used	Available	Utilization
Number of slice registers	2759	28800	9%
Number of LUT	14279	28800	48%
Number of IOB’s	20	485	4%
Number of block RAM/FIFO	2	32	6%
Number of DSP	16	48	33%

Table 4. Resource utilization with previous techniques

Method	Number of slice registers	Number of LUT	Number of IOB	Number of block RAM/FIFO	Number of DSP
[22]	-	-	-	4	82
[14]	3337	5898	-	-	-
[25]	3278	4594	-	-	-
Proposed work	2759	14279	20	2	16

3.2. FPGA prototype

Using the Xilinx device setup tool, the sample rate conversion architecture is confirmed on a Xilinx Virtex-6, XC6VCX240t-2FF484 FPGA board. A reliable technique to check that an SoC design is functionally accurate is to run it on an FPGA prototype. This is a huge advance over designers who depend entirely on software simulations to guarantee that their hardware design is sound. During the initial silicon pass, around a third of all current SoC designs are fault-free, with functional logic errors accounting for about half of all re-spins. A single prototype platform may evaluate hardware, firmware, and application software design capabilities before the initial silicon pass [23].

As circuit complexity rises and time-to-market decreases, the need for verification of application-specific integrated circuit (ASIC) and system-on-chip (SoC) designs grows. Hardware platforms are gaining traction among verification engineers due to their ability to test system designs at high speeds utilizing on-chip bus clocks, rather than simulation clocks, which may not accurately reflect system behaviour [24]. Because these multi-million gate designs cannot be fully implemented on a single FPGA, they are usually implemented on a multi-FPGA prototype platform with six or more FPGAs [25]. The fewer FPGAs that must be partitioned, the less effort the design engineer must expend [26]. An image of an FPGA-based prototype platform with a dual-FPGA setup is shown to the right [27].

3.3. ASIC performance

Figure 4 shows the RTL schematic in the cadence RTL compiler. Table 5 illustrates a comprehensive comparison of device utilization among various ASIC-implemented techniques. The examined methods, including [5] and [9], are evaluated based on core area, power consumption, delay, and technology specifications. Method [5] is characterized by a core area of 16.02 mm², with power consumption ranging from 0.43 to 1.99 mW. Unfortunately, no specific delay information is provided for this method, and it is implemented using 180 nm technology.

Method [9] exhibits a considerably smaller core area of 0.747 mm², accompanied by a low power consumption of 0.063 mW. Similar to [5], there is no specified delay for this method, and it utilizes 90 nm technology. For the 180 nm technology, the proposed work has a core area of 4.825 mm², a power consumption of 0.582 mW, and a delay of 0.402 ns. For the 45 nm technology, the core area further reduces to 1.487 mm², with a power consumption of 0.062 mW and a reduced delay of 0.391 ns.

Table 5. Comparison of device utilization with other ASIC-implemented techniques

Methods	Core Area (mm)2	Power (mW)	Delay(ns)	Technology
[5]	16.02	0.43~1.99	-	180 nm
[9]	0.747	0.063	-	90 nm
Proposed work	4.825	0.582	0.402	180 nm
	1.487	0.062	0.391	45 nm

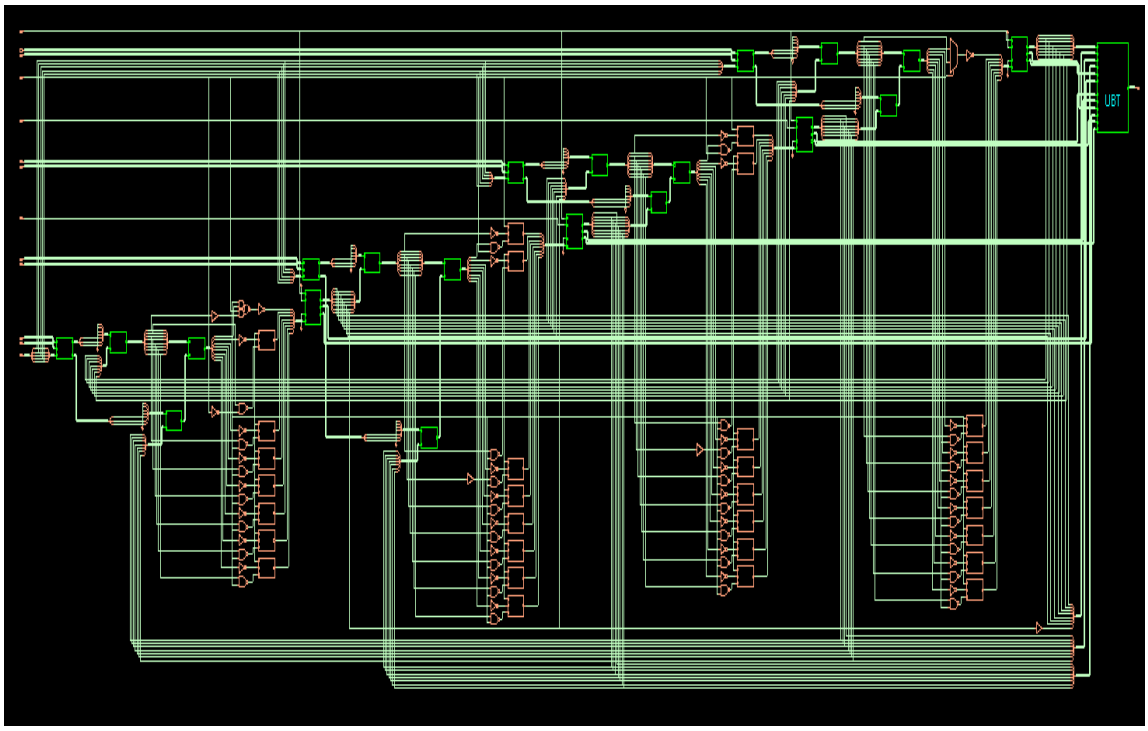

Figure 4. RTL schematic in cadence RTL compiler

4. CONCLUSION

This work implements an energy-efficient design and optimized architecture for multi-standard radio sample rate conversion on the Virtex-6 XC6VCX240t-2FF484 FPGA. A lowpass FIR filter of 16 taps is used to avoid the aliasing effect in the down sampler. A selection MUX and DEMUX are used to control the bidirectional signal conversion process. Decimators and interpolators are used parallel to perform the signal conversion process controlled by a centralized control logic. The usage of sample rate converter design simplifies the implementation of analog rate conversion for up and down sampling, as well as the same for digital signals for efficient processing, resulting in a reduction in hardware resources. The proposed sample rate converter achieved a 155 MHz operating frequency to perform the conversion as well as the filtering process. The suggested architecture has a considerable decrease in hardware resources of roughly 20%,

according to the comparison. This work achieved a core area of 1.487 mm² and a power of 0.062 mW delay of 0.391 in 45 nm technology nodes. Staying updated with the latest FPGA advancements will be essential for optimizing performance and resource utilization. The design uses a lowpass filter with a minimal number of taps to reduce area consumption. However, this simplification might compromise the filter's performance in certain scenarios, particularly with signals having more complex spectra. This could result in inadequate suppression of aliasing and signal distortion, affecting the quality of the converted signal. More advanced filter designs that balance complexity and performance can be investigated to mitigate this issue. Future studies could explore more sophisticated filter designs that offer better performance while maintaining low power consumption and minimal area usage. Investigating adaptive filters that dynamically adjust to varying signal conditions could enhance the converter's robustness.

REFERENCES

[1] C. singh, M. Singh, and S. Sharma, "Efficient implementation of sample rate converter," *International Journal of Advanced Computer Science and Applications*, vol. 1, no. 6, 2010, doi: 10.14569/ijacsa.2010.010606.

[2] R. Bregovic, Y. J. Yu, T. Saramaki, and Y. C. Lim, "Implementation of linear-phase FIR filters for a rational sampling-rate conversion utilizing the coefficient symmetry," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 58, no. 3, pp. 548–561, 2011, doi: 10.1109/TCSI.2010.2072350.

[3] F. Sheikh and S. Masud, "Sample rate conversion filter design for multi-standard software radios," *Digital Signal Processing: A Review Journal*, vol. 20, no. 1, pp. 3–12, 2010, doi: 10.1016/j.dsp.2009.04.014.

[4] S. L. Chen *et al.*, "A power-efficient mixed-signal smart ADC design with adaptive resolution and variable sampling rate for low-power applications," *IEEE Sensors Journal*, vol. 17, no. 11, pp. 3461–3469, 2017, doi: 10.1109/JSEN.2017.2680472.

[5] X. Liu, X. X. Yan, Z. K. Wang, and Q. X. Deng, "Design and FPGA implementation of a reconfigurable digital down converter for wideband applications," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 25, no. 12, pp. 3548–3552, 2017, doi: 10.1109/TVLSI.2017.2748603.

[6] W. Tianqi and L. Cheng, "Sample rate conversion technology in software defined radio," in *Canadian Conference on Electrical and Computer Engineering*, 2006, pp. 1355–1358, doi: 10.1109/CCECE.2006.277567.

[7] K. J. Cho, J. G. Chung, J. S. Park, B. K. Kim, and K. K. Parhi, "Design of a sample-rate converter from CD to DAT using fractional delay allpass filter," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 54, no. 1, pp. 19–23, 2007, doi: 10.1109/TCSII.2006.885067.

[8] C. Wu, C. Kong, S. Xie, and H. Cai, "Design and FPGA implementation of flexible and efficiency digital down converter," in *International Conference on Signal Processing Proceedings, ICSP*, 2010, pp. 438–441, doi: 10.1109/ICOSP.2010.5654948.

[9] L. Sahukar and M. Latha M, "Area efficient fractional sample rate conversion architecture for software defined radios," *ICTACT Journal on Communication Technology*, vol. 05, no. 03, pp. 977–986, 2014, doi: 10.21917/ijct.2014.0140.

[10] W. D. Richard, "Efficient parallel real-time upsampling with Xilinx FPGAs," *Xcell Journal*, vol. 89, pp. 38–44, 2014.

[11] Q. Zhang and X. Su, "The design of digital down converter based on FPGA," 2012, doi: 10.1109/WiCOM.2012.6478707.

[12] X. Zhang and J. Zhu, "High-throughput interpolation architecture for algebraic soft-decision reedSolomon decoding," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 57, no. 3, pp. 581–591, 2010, doi: 10.1109/TCSI.2009.2023935.

[13] B. H. Tietche, O. Romain, and B. Denby, "A practical FPGA-based architecture for arbitrary-ratio sample rate conversion," *Journal of Signal Processing Systems*, vol. 78, no. 2, pp. 147–154, 2015, doi: 10.1007/s11265-013-0840-5.

[14] G. J. Dolecek and F. J. T. Torres, "Multiplierless multiband filter for fractional sample rate conversion," 2008, doi: 10.1109/MICCPE.2008.4555746.

[15] S. Boukhtache, B. Blaysat, M. Grediac, and F. Berry, "Alternatives to bicubic interpolation considering FPGA hardware resource consumption," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 29, no. 2, pp. 247–258, 2021, doi: 10.1109/TVLSI.2020.3032888.

[16] T. O. Otunniyi and H. C. Myburgh, "Low-complexity filter for software-defined radio by modulated interpolated coefficient decimated filter in a hybrid Farrow," *Sensors*, vol. 22, no. 3, 2022, doi: 10.3390/s22031164.

[17] D. Sinha and S. Kumar, "CIC filter for sample rate conversion in software defined radio," 2016, doi: 10.1109/STARTUP.2016.7583972.

[18] S. Li and C. X. Yu, "Ultrahigh sampling rate photonic time stretch analog-to-digital converter employing phase modulation," *Optik*, vol. 124, no. 20, pp. 4539–4543, 2013, doi: 10.1016/j.ijleo.2013.02.011.

[19] E. Martos-Naya, J. López-Fernández, L. Díez del Río, and J. T. E. Muñoz, "Structure of a sampling rate converter based on elementary signal processing blocks," *Digital Signal Processing: A Review Journal*, vol. 18, no. 4, pp. 480–487, 2008, doi: 10.1016/j.dsp.2007.06.001.

[20] K. Rajamani, Yhean-Sen Lai, and C. W. Furrow, "An efficient algorithm for sample rate conversion from CD to DAT," *IEEE Signal Processing Letters*, vol. 7, no. 10, pp. 288–290, Oct. 2000, doi: 10.1109/97.870683.

[21] A. Franck, "Arbitrary sample rate conversion with resampling filters optimized for combination with oversampling," *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, pp. 149–152, 2011, doi: 10.1109/ASPAA.2011.6082271.

[22] N. Michael and A. P. Vinod, "Reconfigurable architecture for arbitrary sample rate conversion in software defined radios," 2008, doi: 10.1109/PIMRC.2008.4699572.

[23] L. Wang and Y. Zhao, "Signal generation techniques based on arbitrary sample rate conversion," in *Proceedings - 2014 IEEE International Conference on Computer and Information Technology, CIT 2014*, 2014, pp. 446–449, doi: 10.1109/CIT.2014.121.

[24] C. Schmidt-Knorreck, R. Knopp, and R. Pacalet, "Hardware optimized sample rate conversion for software defined radio," *Frequenz*, vol. 64, no. 11–12, pp. 204–209, 2010, doi: 10.1515/FREQ.2010.64.11-12.204.

[25] B. Guoan and S. K. Mitra, "Sampling rate conversion in the frequency domain [DSP Tips and Tricks]," *IEEE Signal Processing Magazine*, vol. 28, no. 3, pp. 140–144, 2011.

[26] G. Bi, S. K. Mitra, and S. Li, "Sampling rate conversion based on DFT and DCT," *Signal Processing*, vol. 93, no. 2, pp. 476–486, 2013, doi: 10.1016/j.sigpro.2012.08.023.

[27] A. Agarwal, L. Boppana, and R. K. Kodali, "A factorization method for FPGA implementation of sample rate converter for a multi-standard radio communications," in *IEEE 2013 Tencon - Spring, TENCONSpring 2013 - Conference Proceedings*, 2013, pp. 530–534, doi: 10.1109/TENCONSpring.2013.6584501.

BIOGRAPHIES OF AUTHORS

Swetha Pinjerla is a assistant professor, ECE, MRCET has done B.Tech and M.Tech from JNTUH, Hyderabad. Pursuing Ph.D. in JNTUH, Hyderabad. Has teaching experience of 14 years. Published more than 12 papers in reputed international and national journals/conferences in the area of wireless communications, image and signal processing. She can be contacted at email: pswetha0823@gmail.com.

Surampudi Srinivasa Rao is a professor of ECE, Principal, MRCET has done B.Tech from Madras Institute of Technology, Chennai, M.Tech from JNTUH, Hyderabad and Ph.D. from JNTUH, Hyderabad. Has 27 years of teaching experience which include 8 years of research and 2 years of industrial experience. He is an author of more than 25 papers in reputed international and national journals/conferences in the area of wireless communication, speech processing and wireless networks. He can be contacted at email: ssrao.atri@gmail.com.

Puttha Chandrasekhar Reddy is a professor in ECE, JNTU, Hyderabad. He has completed B.Tech (ECE) from JNTU Anantapur and M.Tech in Applied Electronics from Baratiar University. Has done Ph.D. from JNTU Anantapur on “Routing in Adhoc networks”. Has 30+ years of teaching experience and an author of numerous technical papers in renowned international and national journals/conferences. He can be contacted at email: drpcsreddy@gmail.com.