International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Index Copernicus Value (2013): 6.14 | Impact Factor (2013): 4.438

ASIC Design of Sample Rate Convertor

Roopa M[1], H. Sudha[2]

[1]Post Graduate Scholar, Department of ECE, Bangalore Institute of Technology, Bangalore India
[2]Associate Professor, Department of ECE, Bangalore Institute of Technology, Bangalore, India

Abstract: ***At present, video and audio are mostly based on complete digital processing of a signal. Therefore more problems come if the equipment is based on different sampling rate. Before it was only signal digital processor and in particular signal chain and we had Analog to Digital or Digital to Analog converters used in those chains. Since the era now uses almost all digital we need a digital interfacing between different digital equipment and it's called as Sample Rate Converter. Therefore sample rate converter is a device used for converting a signal frequency from one to another.***

Keywords: ASIC Design, FIR form polyphase filter implementation, physical design

1. Introduction

We are in the age of technology and innovations in various fields such as communication, VLSI and digital signal processing. Always there has been strive for innovation in the field of VLSI especially to better three parameters which are the bread butter for a VLSI engineer i.e. delay, power and area. As we know the electronics field is always shrinking the world in terms of area occupied by any electronic component as well in terms of communication. The device length has ranged from µm to nm nowadays. Number of devices on chip have ranged from few thousands to billions of devices. That's how the technology is moving forward by compacting most of the electronic components into a single chip which has highly unimaginable few decades ago.

Taking an example of a mobile phone, its size and features have varied considerably. In the earlier times the size was big enough so that people would be frustrated to carry along with them, but now the size has shrunk effectively of about palm size. Also the features of the mobile were initially just to make calls whereas nowadays mobile has been used for multiple tasks such as video calling, internet, camera, video recording, remote, Wi-Fi etc. All these innovations are possible due to mainly two factors in the VLSI field, that is, the performance of the device has increased tremendously as well as the size of the device has reduced.

The performance of any system depends on how fast it can compute the given task and produce the result. Hence a Sample Rate Converter which provide a better performance with reasonable timing and area would be highly appreciated. Hence different architectures are defined and used to fulfill our requirements.

A sample rate converter is used for conversions of frequencies. These converters are used for many applications like broadcasting, digital compact disc, digital audio tape, digital communication, biomedical, speech processing etc. There are many implementation of sample rate converter. Our main criterion is to achieve a design which has higher speed, lesser timing and area compared to all. Hence several research papers have been reviewed and are presented in this section.

2. Interpolation Implementation

Interpolation is used to increase the sampling rate. There are many implementation techniques for interpolation design. It can be design with FIR or IIR filter, and also cascade of multiple stages. We will see the polyphase implementation. Output sample rate is first made more by factor I by appending I-1 zeros in between inputs, then in time domain the samples are separated by Ts/I = 1/(I*Fs), Ts and Fs are sampling period and sample frequency. Reflections of the original signal are filtered out using low pass filter.

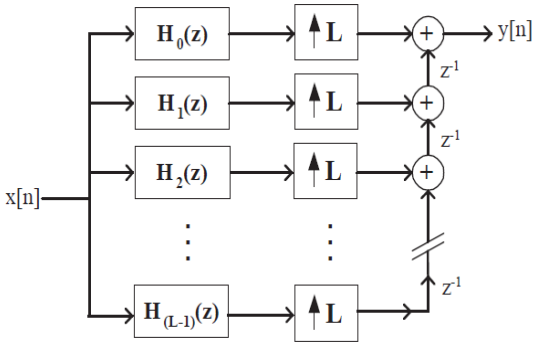


Figure 1: polyphase Implementation

With the case of expander with value L followed by an FIR filter of length N, each output would have come directly from the filter without subfilters technique and therefore needing N multiplies to compute. This would have needed N multiplies/output sample for FIR filter. With polyphase we have decomposition into L component, each of length N/L. with a single input sample to the system. It is processed by L component filters, each of length N/L, and hence L · N/L = N multiplies are required. With this single input sample, every element filter produce a single output point. every output point passes to expander of value L, generating output points of order L. later these output points pass through delay elements and an adder, to produce the final sequence of L output samples. For computation we required N total multiplies to produce L output points, and hence the computation cost is reduced to N/L multiplies/output sample for FIR polyphase filter.

Licensed Under Creative Commons Attribution CC BY

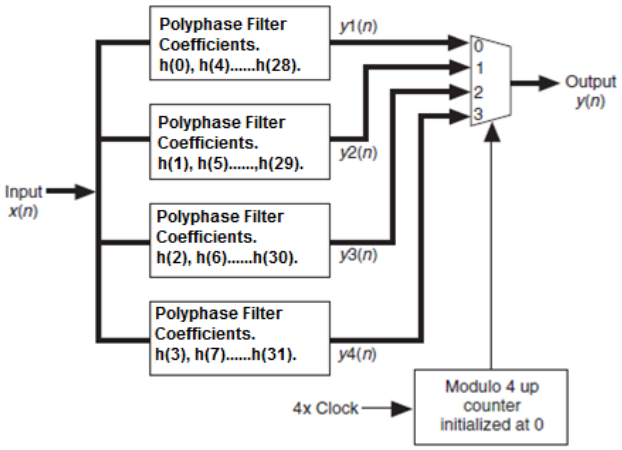


Figure 2: Polyphase Representation of Interpolation Filter

3. Design of Interpolator

The below figures shows the implementation of the direct form polyphase interpolation filter and transposed form polyphase interpolation filter.

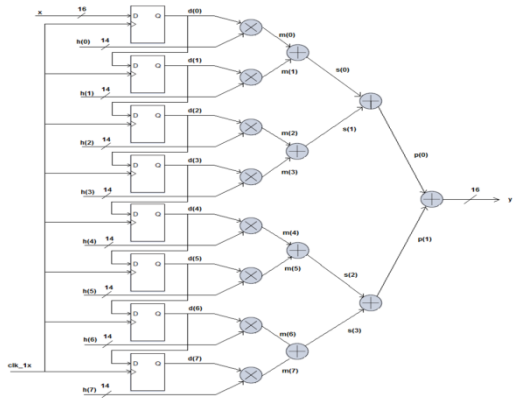


Figure 3: Polyphase Architecture Using Direct Form Structure

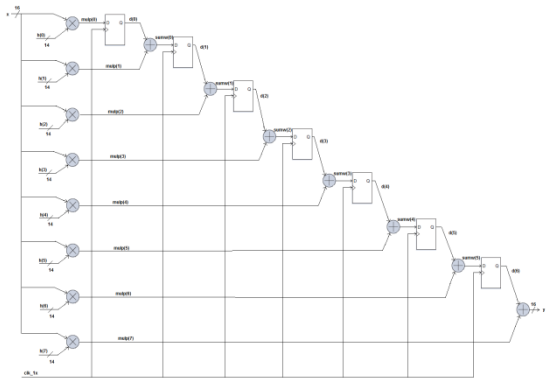


Figure 3: Polyphase Architecture Using Direct Form Structure

Direct form has huge addition at the output, so implementation in hardware is difficult even though maps well to MAC operation on DSP processor. Transposed-form has small additions separated by delay elements it is good for FPGA or ASIC. Direct form needs more pipelined registers to reduce the delay and get high throughput. Transpose form have registers between the adder and can achieve high throughput without any extra pipelined registers.

4. Introduction to ASIC Design

ICs are design on circular silicon wafer. There are three categories, first is the full custom, next is semicustom and it is further classified into standard cell ASIC and gate array based ASIC and the last is, programmable ASIC which has programmable logic device and field programmable gate array.

In full-custom design the, the circuit designed is specific for particular application only and cannot be used for others. This won't use already designed cells for design. Only under the absence of required cell libraries this approach is taken up. The causes may be that the library cells are not at the required speed or it may be big in size or might not consume lot of power.

This form uses already designed cells know as standard cells. Cell based ASIC means or CBIC means a standard cell based ASIC design. Rows of standard cells are the area for the standard cells. The area may use already designed cells, microprocessor,

In this form the transistors are designed before only on the silicon wafer. The transistor on a gate array is base array, and the smallest component to build the base array is called the base cell. Only the mental layers used for interconnect are defined using custom masks. There are different types of gate array based ASICs; they are channeled gate arrays, channel-less gate array and structured gate array.

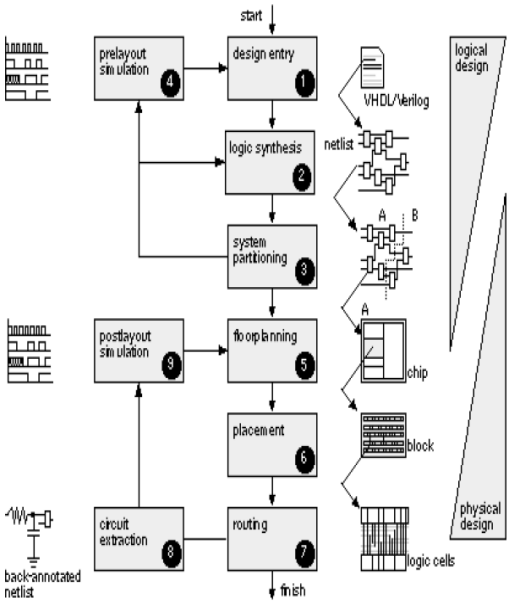


Figure 4: ASIC design flow

Simulation is used to check the functionality of the design, that is to check if the output is got as needed, behavioral, functional, static, gate-level, switch level, transistor level or circuit level are the variations in simulations. The list is from higher level to low level simulation so accurate results also are more. But its complex and take more amount of time too. This step is also called as verification. We should first take in the design and the test bench written in VHDL or Verilog. The next step is the compilation step. Later we should

Licensed Under Creative Commons Attribution CC BY

perform elaboration that is construct design hierarchy and connects signals when the elaboration is complete then we do simulation that is checking the functionality. The next step in the simulation topic is the code coverage; code coverage is done to know how well the HDL code is exercised by the test bench. There are different type of coverage, they are block, expression, FSM, toggle. This is done to see if all possible conditions are satisfied. The necessary inputs to do synthesis are RTL, standard cell library and constraints. We need to first set the library path for both the libraries and the RTL. The libraries used are fast, slow; typical which uses are different from each other when compared with timing, area, power. After reading in the library files and RTL the next is to construct design hierarchy and connect signals. We need to give some timing constraints to the clock which we using in our design to estimate the timing. Then we need to map the design to the best effort which may be medium. And synthesis to generic can also be done. The constraint file we talking about is the SDC file. Using the SDC file we are going to define period, pulse width, rise and fall time, uncertainty and also the input and output delays for different signals. After the synthesis is done we can look into the schematic of our design. Dump out the power, timing, area reports to judge the performance and write out he netlist and then the SDC and SDF files for timing constraints. The dumped files are needed to perform the physical design. This is the backend design, physical design is also known as place and route.

5. Results

We are going to use cadence tool for implementing our design. In this chapter we will be discussing the results which are obtained by implementing the sample rate convertor for audio applications. The language used for coding is Verilog HDL. After the simulation is done to check for functionality we go for synthesis. This is done to get the gate level netlist. We will compare both the form of sample rate convertors for area, timing and power.

For a given input in the test bench we verify the waveforms. The input we give is the sine wave. This sine wave is dumped into a file by coding for a sine wave in matlab. The snapshot of the waveforms for direct form and transposed form polyphase sample rate convertor is shown below.

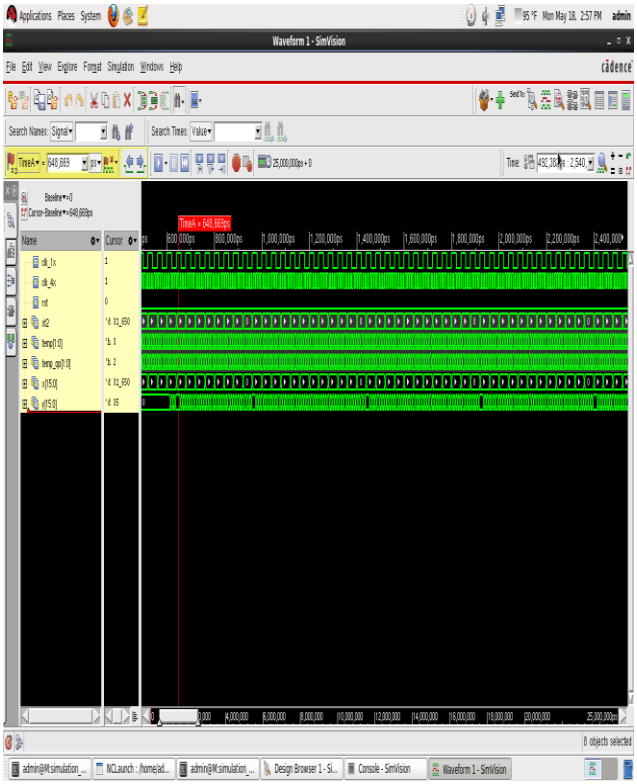


Figure 5: Waveform for direct form polyphase interpolator

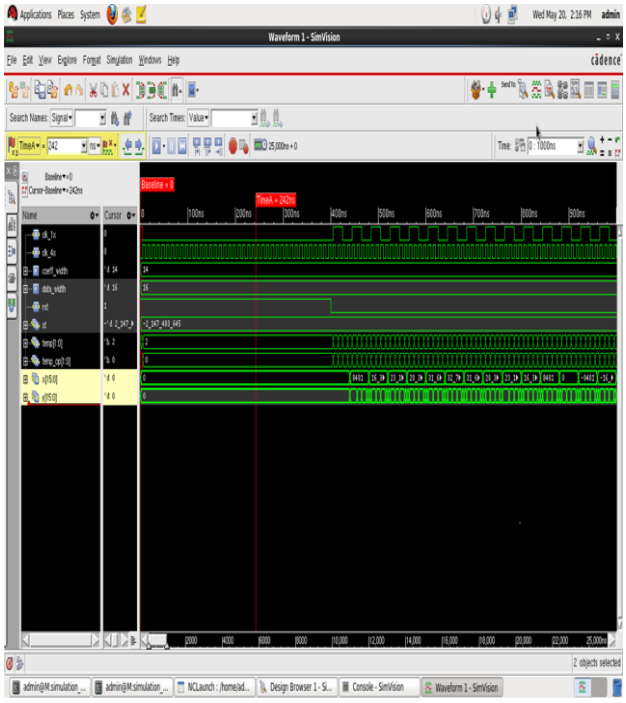


Figure 6: Waveform for transposed form polyphase interpolator

Table 1: Parameter comparisons

Sample rate convertor	Delay(ps)	Area(μm²)	Power(nw)
Direct form	9002	129996	7168028.043
Transpose form	767	99915	8538846.799

Licensed Under Creative Commons Attribution CC BY

Our goal of this project is to implement the sample rate convertor in two methods. That is the direct form implementation and the direct form implementation. And we are proving that the transposed form has more advantage in timing and area compared to direct form implementation. The table below shows the comparison for each of the parameters.

Libraries or data sets are needed for the physical design of an application specific integrated circuit. The libraries have the physical layout, abstract view, timing models, simulation or functional models, and transistor level circuit description. Success of the fabricated ASIC design depends on accuracy of the libraries. Standard cell libraries, input output pads, memories and custom libraries are mainly used. Custom libraries which are also known as intellectual property (IP) libraries has PLL, ADC, DAC, voltage regulators.

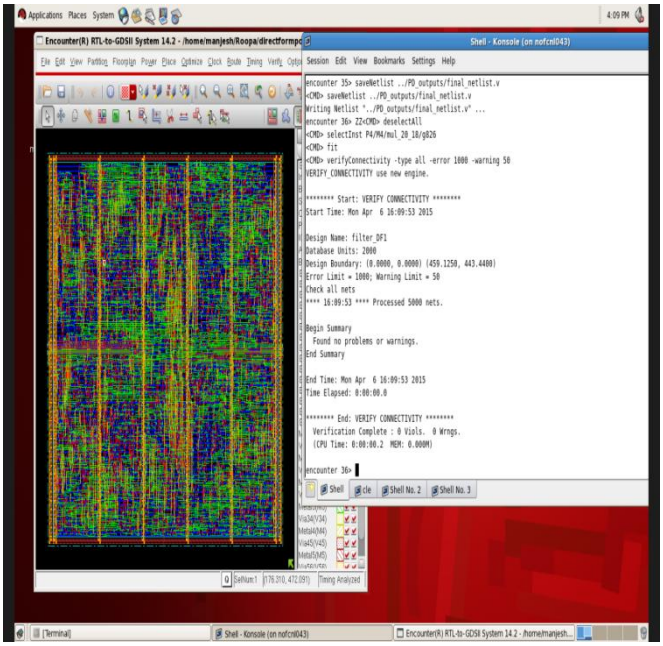


Figure 7: The routing for the design

6. Conclusion

Since we know that this era have many components working at varying frequencies we need to have sample rate converter to get them together. Polyphase structure is designed in both direct form and transposed form. Its architecture was implemented and realized in Hardware description language (Verilog) using Cadence tools.

The simulation outputs of HDL design was verified. This design is then synthesized to obtain gate level schematic and its area, timing and power report was generated. Later the design is taken to the backend and successfully routed. The applications are, Digital audio player, Digital audio recorder, Digital mixing desk, Digital audio interface for computers, Digital audio routers and distribution systems, Broadcast studio equipment, DVD/CD recorders, Surround sound decoders, Car audio system

References

[1] Altera Corporation “Implementing High Performance DSP Functions in Stratix & Stratix GX Devices Handbook”.
[2] John G. Proakis, Dimitris G. Manolakis “Digital Signal Processing: Principles, Algorithms, and Applications”, Prentice Hall
[3] Fedor Merkelov, Yaroslav Kodess, LiTH-ISY-EX-3520-2004 Linköping, 2004 “Design and Implementation of Sampling Rate Converters for Conversions between Arbitrary Sampling Rates”
[4] Daniel B. Turek “Design of Efficient Digital Interpolation Filters for Integer Upsampling”
[5] Ivar Løkken, 12/4-05 “The ups and downs of arbitrary sample rate conversion”
[6] Dave Rossum E-mu systems, inc. Santa Cruz, CA USA “some aspects of sample rate conversion”
[7] Thomas A. Baran, “Design and Implementation of Discrete-Time Filters for Efficient Sampling Rate Conversion”
[8] Sanjit K. Mitra, “Digital Signal Processing: A Computer-Based Approach” 3rd International Edition.
[9] Nazeih M. Batros, “HDL Programming VHDL and Verilog”, Dreamtech
[10] Samir Palnitkar , “Verilog HDL A guide to Digital Design and Synthesis”, SunSoft Press 1996.
[11] V. Kamakotiand Shankar Balachandran “CAD for VLSI Design –II”
[12] Khosrow Golshan, “PHYSICAL DESIGN ESSENTIALS, An ASIC Design Implementation Perspective”
[13] J. Bhasker, Rakesh Chadha “Static Timing Analysis for Nanometer Designs A Practical Approach”
[14] Wang Lijun, Zhao Yun “Signal Generation Techniques based on Arbitrary Sample Rate Conversion” 2014 IEEE International Conference
[15] Yu Huijun “Design Of A Sample-Rate Converter Based On Least-Square Method” 2012 IEEE International Conference.
[16] Wang Lijun, zhao Yun” Signal generation techniques based on arbitrary sample rate conversion” 2014 IEEE International conference.
[17] Haipeng Kuang, Dejiang Wang, Gang Zhou, Zhengping Xu” A Multi-Channel, area-efficient, audio sampling rate interpolator “ 2009 IEEE International Conference.
[18] Andreas Franck and Karlheinz Brandenburg “An overall optimization method for arbitrary sample rate converters based on integer rate SRC and large range interpolation” 2009 IEEE International conference
[19] Yukio Mori, Kenji Nagasawa, and Naoyuki Aikawa “A design method and structure of kernel for sampling rate converter” IEEE international conference.
[20] Noura Ben Ameur, Mourad Loulou “Design of efficient Digital interpolation filters and sigma-delta modulator for Audio DAC” IEEE International conference

Author Profile

Roopa M received his B.E in Electronics and communication from Visvesvaraya Technological University (VTU) in 2010. She has worked at Appsconnect technologies, Bangalore for 1 year in the field of backend. Currently, she is pursing M.Tech in VLSI Design and Embedded systems from Bangalore Institute of Technology, VTU.

Licensed Under Creative Commons Attribution CC BY