ISSN (Print) : 2319-5940
ISSN (Online) : 2278-1021

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 2, Issue 10, October 2013

FPGA Implementation of Sigma Delta ADC on Spartan 2

Simranpreet Kaur[1], Dr. Charanjit Singh[2]

Department of Electronics and Communication, Punjabi University, Patiala, India[1, 2]

Abstract: Analog-to-Digital Converters plays an important role in our daily life. Sigma Delta (∑Δ) Converter is an attractive ADC for future communication systems due to its high resolution rate. This paper presents Sigma delta ADC which is suitable for embedded FPGA applications. Designing of Sigma Delta ADC has done in MATLAB simulink and implemented it on Spartan2 FPGA kit.

Keywords: Analog-to-Digital converters (ADC), Field programmable gate arrays (FPGA), Sigma Delta (∑Δ), VLSI, User Constraint File (UCF).

I. INTRODUCTION

Analog-to-digital converters play a important role in modern audio and communication design. Nyquist converters are used efficiently only for medium resolutions and require analog components that are precise and highly immune to noise and interference. On the other hand oversampling converters are used to achieve high resolutions (>20bits). In conventional oversampled modulators, negative feedback is applied, to control the dynamic behaviour of a system and to realize the attenuation of the quantization noise in the signal band due to noise shaping. With the increasing demand of electronics product, complexity of designing the circuits is growing high. Due to competitive pressures available time to design products is shrinking. Field programmable gate arrays (FPGA) provides best solution to address all these issues. Field programmable gate arrays are able to offer advantages over traditional VLSI technology.

II. SIGMA DELTA MODULATOR

Analog-to-Digital converters are classified into two classes: (1) Nyquist rate ADC and (2) Oversampled (Sigma Delta) ADC. In Nyquist-rate ADC the input analog signal samples at the sampling frequency (f_s) is equal to the twice of the highest frequency component of the input signal. It is necessary for the input signal to be band-limited, to prevent aliasing; an anti-aliasing filter must be used before the converter. This filter must have a very narrow transition band to ensure that the filtered signal does not contain any frequency component above $f_s/2$. But to realize narrow transition band anti-aliasing filter is very difficult. Also Nyquist rate converters have low resolution which is not suitable for a very low voltage signal conversion. Oversampled ADCs are preferred over Nyquist rate ADC due to their high signal to noise ratio and high resolution [4]. In case of Sigma-Delta ADC the modulator portion behaves like a noise shaper and digital filter removes the out of band quantization noise thus ensure much higher SNR which is impossible to achieved in Nyquist rate ADC [5]. In addition to improve SNR, oversampling possesses inherently the motivation for prediction. When the signal is oversampled then it does not change significantly in the interval between successive samples [6]. Since the values of these samples are very close, they are highly correlated and therefore future samples could be predicted from the past one. The Linear Delta modulator is the simplest predictive modulator. By pushing most of the in-band noise outside the signal frequency band more improvement in the SNR can be achieved. This is attainable if signal transfer function is a low pass whereas and Noise transfer function is high pass. This technique is called noise shaping and can be easily and efficiently implemented by modifying the delta modulator [7]. Here the integral of the input signal is encoded rather than the input signal directly. Clearly integration being a linear function does not affect system transfer function. The demodulation integrator at output can be placed at the input of Delta modulator. This modification of Delta modulation system results to the new system called Sigma Delta Modulator [1] as shown in Figure1.

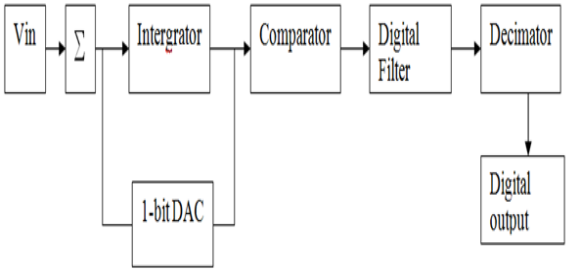

Figure 1 Block diagram of Delta Sigma modulator

Sigma Delta analog to digital converter consists of analog part and digital part. Analog part contains modulator which samples the input signal at high rate and thus produces output in the form of binary sequence. Digital

ISSN (Print) : 2319-5940
ISSN (Online) : 2278-1021

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 2, Issue 10, October 2013

portion of the Sigma Delta ADC consists of digital decimation filter. Decimation process can be done by using single-stage as well as multi-stage decimation.

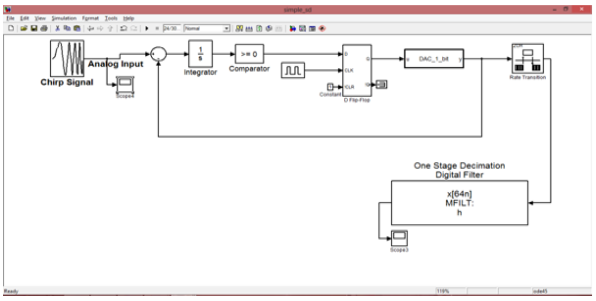

Figure 2. Delta Sigma modulator in MATLAB simulink

Here, in the first stage, decimation is done by using single-stage FIR decimator filter. The function of this filter is to calculate average value of output sequence by using low-pass decimating FIR filter.

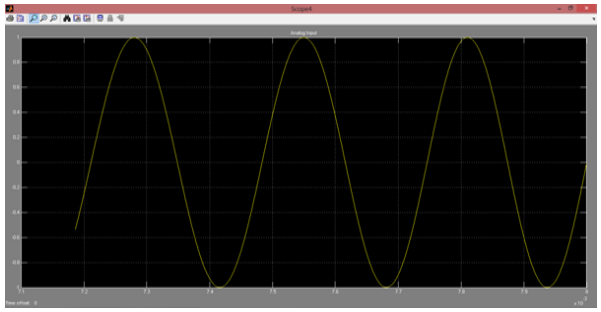

Figure 3 Waveform of the analog chirp signal.

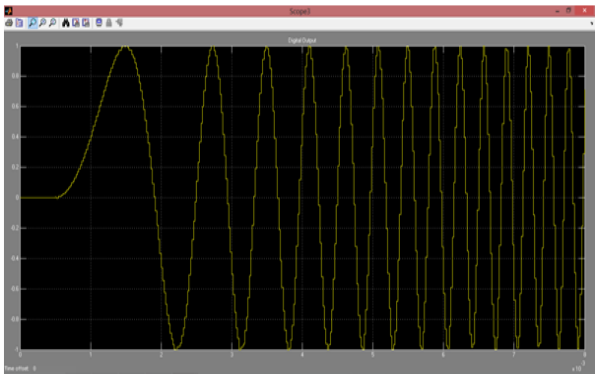

Figure 4 Waveform of the output digital signal.

The designed Sigma-Delta ADC model is not an efficient model because filter size is very large. It is not easy to implement such a large size filter on FPGA chip. Hence to overcome this problem, we can break the large decimator filter into three-stage decimation filter. The very first stage decimates the signal by the factor of 8, second stage by 2 and last stage by 4.

By using decimation process which consists more than one decimator filter, computational difficulties are reduced but this process provides a greater filter delay in the output.

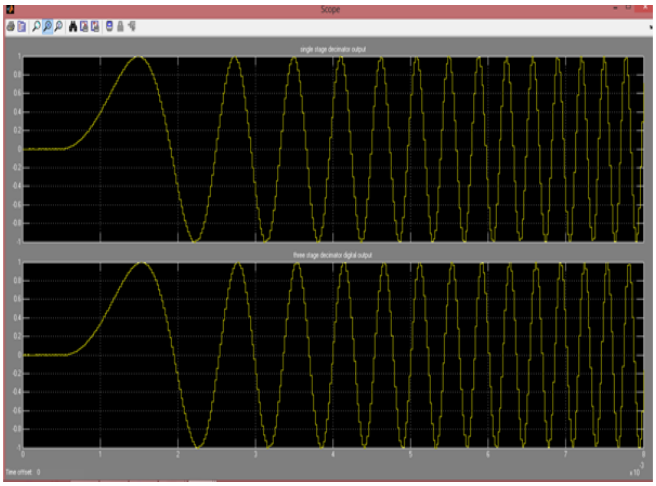

Figure 5 Waveform shows the delay in digital output signals of single-stage and multi-stage decimator filter.

For the implementation of FPGA, designed model should be in fixed point. But our all designed models are not in fixed point; hence we convert designed Sigma-Delta ADC in fixed point.

III. IMPLEMENTATION RESULTS

In our proposed model, there is a difficultly to convert the model into fixed point. To resolve this problem we can simply designed the Sigma Delta ADC using system generator tool of Xilinx. This tool automatically generates the VHDL code.

Steps for Implementation of Sigma Delta ADC on FPGA:

(1) Design Sigma Delta ADC using system generator tool of Xilinx blockset.
(2) Generate VHDL code on Xilinx ISE software.
(3) Assign pin locations for all the ports of Sigma Delta ADC.
(4) Generate UCF file.
(5) Generate Package Pins and Device Architecture for Spartan xc2s50-5pq208 FPGA kit.
(6) Implement the Sigma Delta ADC on FPGA using Configure Device (iMPACT) process.

Block diagram of ADC is shown in Figure 6. ADC generates binary bit stream which becomes input to the digital portion The proposed Sigma Delta ADC has been implemented on Spartan2 xc2s50-5pq208 Device. Top level RTL schematic of the proposed ∑Δ ADC is shown in Figure 7, its internal view is shown in Figure 8 and its technology schematic is shown in Figure. 9.

ISSN (Print) : 2319-5940
ISSN (Online) : 2278-1021

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 2, Issue 10, October 2013

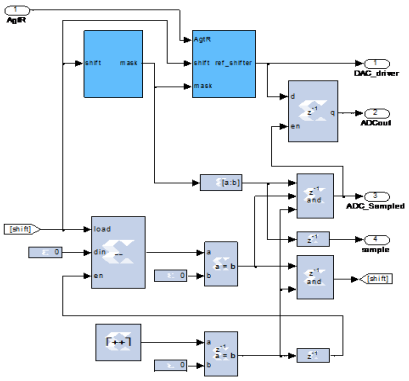

Figure 6 Block Diagram of ADC using system generator toolbox.

Figure 7 Top Level RTL Schematic of Proposed ∑Δ ADC

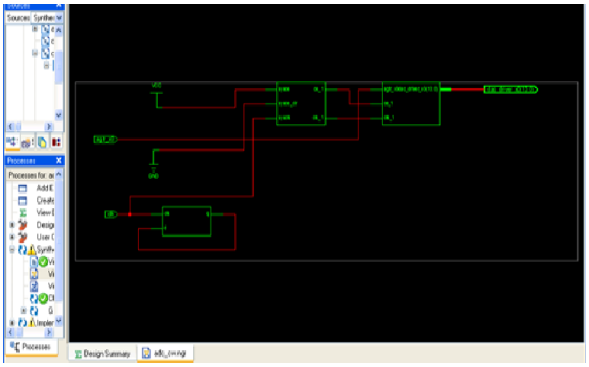

Figure 8 Internal structure of Proposed ∑Δ ADC

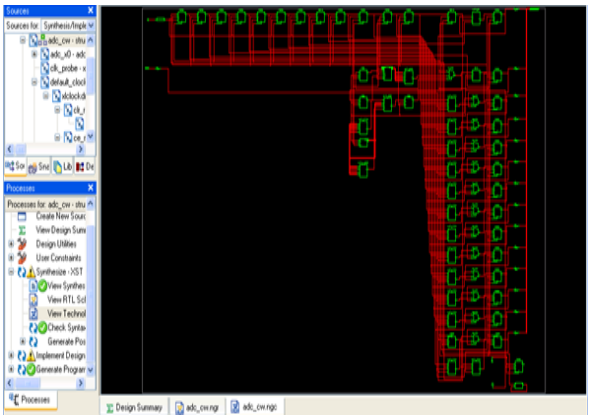

Figure 9 Technology Schematic of Proposed ∑Δ ADC

Device utilization summary gives whole information regarding the used devices from available devices in proposed ∑Δ ADC as shown in Table 1.

Table 1: Resource Utilization for Spartan2 xc2s50-5pq208 Device

Device Utilization Summary				
Logic Utilization	**Used**	**Available**	**Utilization**	**Note(s)**
Number of Slice Flip Flops	46	1,536	2%	
Number of 4 input LUTs	24	1,536	1%	
Logic Distribution				
Number of occupied Slices	32	768	4%	
Number of Slices containing only related logic	32	32	100%	
Number of Slices containing unrelated logic	0	32	0%	
Total Number of 4 input LUTs	45	1,536	2%	
Number used as logic	24			
Number used as a route-thru	21			
Number of bonded IOBs	15	140	10%	
Number of GCLKs	1	4	25%	
Number of GCLKIOBs	1	4	25%	
Total equivalent gate count for design	635			
Additional JTAG gate count for IOBs	768			

Device Utilization Summary				
Logic Utilization	**Used**	**Available**	**Utilization**	**Note(s)**
Number of Slice Flip Flops	46	1,536	2%	
Number of 4 input LUTs	24	1,536	1%	
Logic Distribution				
Number of occupied Slices	32	768	4%	
Number of Slices containing only related logic	32	32	100%	
Number of Slices containing unrelated logic	0	32	0%	
Total Number of 4 input LUTs	45	1,536	2%	
Number used as logic	24			
Number used as a route-thru	21			
Number of bonded IOBs	15	140	10%	
Number of GCLKs	1	4	25%	
Number of GCLKIOBs	1	4	25%	
Total equivalent gate count for design	635			
Additional JTAG gate count for IOBs	768			

A. Assigning Pin Location Constraints

Pin locations for all the ports of proposed ∑Δ ADC are specified so that all available pins are connected on FPGA kit efficiently.

All the information related to the pin locations are saved in UCF file.

UCF FILE

#PACE: Start of PACE I/O Pin Assignments

NET "agtr_x0" LOC = "p99" ;

NET "clk" LOC = "p80" ;

NET "dac_driver_x0<0>" LOC = "p45" ;

NET "dac_driver_x0<10>" LOC = "p71" ;

NET "dac_driver_x0<11>" LOC = "p59" ;

NET "dac_driver_x0<12>" LOC = "p58" ;

NET "dac_driver_x0<13>" LOC = "p57" ;

NET "dac_driver_x0<1>" LOC = "p46" ;

NET "dac_driver_x0<2>" LOC = "p47" ;

NET "dac_driver_x0<3>" LOC = "p61" ;

NET "dac_driver_x0<4>" LOC = "p62" ;

NET "dac_driver_x0<5>" LOC = "p63" ;

NET "dac_driver_x0<6>" LOC = "p67" ;

NET "dac_driver_x0<7>" LOC = "p68" ;

NET "dac_driver_x0<8>" LOC = "p69" ;

NET "dac_driver_x0<9>" LOC = "p70" ;

ISSN (Print) : 2319-5940
ISSN (Online) : 2278-1021

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 2, Issue 10, October 2013

Xilinx PACE window contains all information related to Package view and Architecture view as shown in Figure 10 and Figure 11 respectively.

All the assigned pin locations in both package view and architecture view are shown in dark blue color.

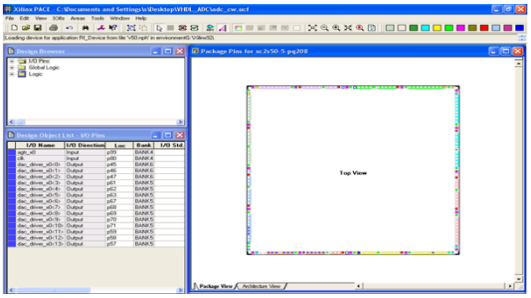

Figure 10 Package Pins for xc2s50-5pq208

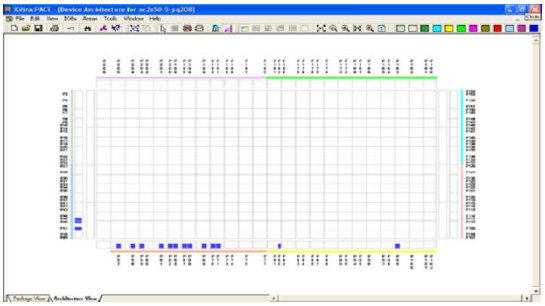

Figure 11 Device Architecture for xc2s50-5pq208.

B. *Implementation on FPGA kit*

This is the last step for the implementation of ∑Δ ADC on FPGA. Connect the power cable to the FPGA kit power input. After that download cable is connected between the FPGA kit and PC. Then run the Configure Device (iMACT) process.

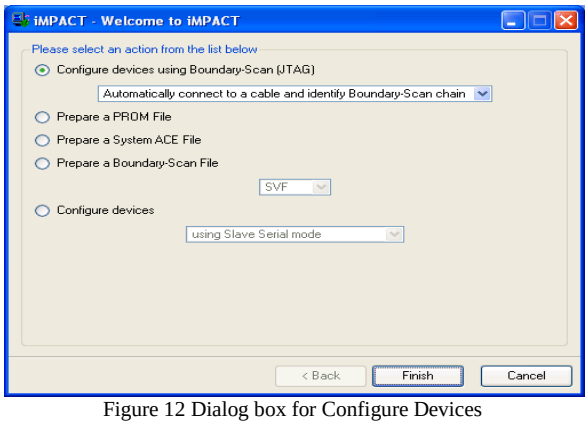

Figure 12 Dialog box for Configure Devices

When programming is complete, the program succeeded message is displayed as shown in Figure 13.

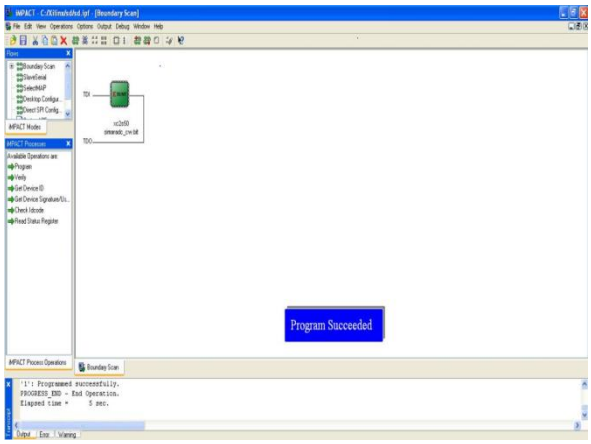

Figure 13 iMPACT boundary scan dialog box

After that the results are shown on FPGA kit. Apply input through DIP switches, i.e. input is applied on pin no. p99 and clock input is applied at pin no. p80. And output is displayed on LEDs as shown in Figure 14.

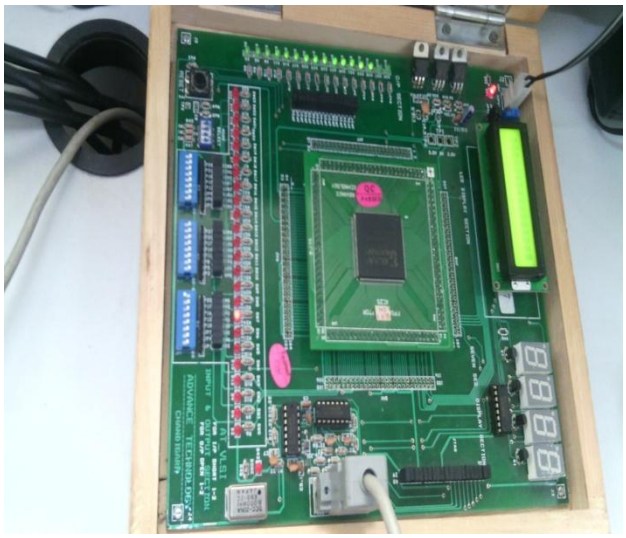

Figure 14 FPGA spartan2 xc2s50-5pq208 Kit

IV. CONCLUSION

In this paper, we have presented design of Sigma Delta ADC and its implementation on Spartan2 FPGA kit. It consumes only 49 number of slices out of available 768, 70 Slice Flip Flops out of 1536, 48 number of 4 input LUTs out of 1536, 3

number of bonded IOBs out of 128 and only 1 number of GCLKs out of available 8 respectively. Results show that the proposed design is quite accurate.

ACKNOWLEDGMENT

Simranpreet kaur Author wishes to express her sincere gratitude to Dr. Charanjit Singh, Assitant Professor, Punjabi University, Patiala for guiding her throughout the current research work.

ISSN (Print) : 2319-5940
ISSN (Online) : 2278-1021

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 2, Issue 10, October 2013

REFERENCES

[1] Eric T. King, Aria Eshraghi, Ian Galton, and Terri S. Fiez, "A Nyquist-Rate Delta–Sigma A/D Converter," IEEE Journal of Solid-State Circuits, Vol. 33, no. 1, pp. 45-52, January 1998.
[2] Memon T.D., Beckett P., Sadik A.Z., "Power-Area Performance Characteristics of FPGA-based Sigma Delta FIRfilters", J Sign Process System, Vol 70, no 3, pp 275-288. March 2013.
[3] Pascal Witte, John G. Kauffma, Joachim Becker, Maurits Ortmanns. "A Correlation-Based Background Error Estimation Technique for Bandpass Delta–Sigma ADC DACs", IEEE Trans. on Circuits & Systems, Vol. 58, no. 11, Nov. 2011.
[4] George I Bourdopoulos, Aristodemos Pnevmatikakis, Vassilis Anastassopoulos, Theodore L Deliyannis: Delta sigma Modulators, Modeling, Design & simulation. Imperial college press, 2006.
[5] Yuanyuan Zhang; Yanfeng Jiang, "Noise shaping sigma delta modulation techniques and models simulation," Solid-State and Integrated Circuit Technology, 2006. ICSICT '06. 8th International Conferenc, pp. 23-26, October 2006.
[6] Sangil Park, "Principles of Sigma Delta Modulation for Analog to Digital Converters", Motorola Digital Signal Processors.
[7] B.P. Lathi, " Modern Digital and Analog Communication Systems," oxford press , Third edition.
[8] Gabor C. Temes and Richard Schreier, Understanding Delta-Sigma Data Converters, IEEE Press.
[9] RI W. Stewart and E. Pfann, "Oversampling and sigma-delta strategies for data conversion," Electronics & Communication Engineering Journal, pp.37-47, February 1998.