Complex Demodulator: A VLSI Architecture and FPGA Implementation

Neema Rajan, Jayaraj U K
Dept. of Electronics & Communication, Calicut University

***Abstract*—In this paper we developed Complex demodulator with application in mind as Ultrasound Imaging. The main objective is to model in MATLAB™, hardware architecture implementation in Verilog HDL adopting 1.15 fixed point data representation and synthesizing using XILINX ISE® Spartan6 FPGA as target. The developed architecture consists of input memory, LUTs, Mixers and FIR filters which convert complex modulated signal stored in 1.15 format into in-phase (I), and quadrature-phase (Q) channel outputs each of signed 32 bit. The hardware based results have been compared and validated with entire demodulator modeled in MATLAB® and finally the error values are calculated. The maximum working frequency of the design can be as high as 151.777MHz with minimum period of 6.589ns utilizing 8 clock cycles per symbol and proved to be simple, convenient as well as hardware resource saving than other conventional implementation methods.**

***Keywords*—Complex Demodulator, IP Core, IQ demodulator, CosLUT, Sine LUT.**

I. INTRODUCTION

Ultrasound imaging systems are used in medical field to explore interior areas of a patient's body. It displays images in real time and is considered safe, noninvasive and less costly than other cross-sectional imaging modalities, e.g., X-ray CT, MRI and PET [3]. In various embodiments the probe does not include the demodulator rather it is performed by the complex demodulator within the host system. More specifically the complex demodulator performs digital demodulation, and optional filtering is adopted. The demodulated ultrasound data may be stored in a memory such as temporarily to perform one or more embodiments. The complex demodulator demodulates the RF signal to form IQ data pairs representative of the echo signals, which in various embodiments have a reduced data transfer rate than the transfer rate of the ADC. The RF or IQ element data may then be routed directly to the memory. The processor then processes the output of the RF processor and prepares frames of ultrasound information for display. The display includes monitors that include patient information and ultrasound images for diagnostics and analysis.

Any band pass signal can be represented by a sum of in-phase and quadrature components. The in-phase and quadrature signals represent a slowly varying complex envelope which is applied to the high-frequency carrier. A complex demodulation is the standard approach used to obtain quadrature samples. The input signal is multiplied by the output of a quadrature oscillator running at the demodulation frequency. The sine and cosine products are low-pass filtered to remove the sum frequencies. The down shifted difference frequencies are digitized at a rate greater than signal bandwidth, yielding a pair of in-phase and quadrature samples.

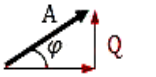
II. THEORY

A. IQ Demodulation

Here, the baseband digital information is separated into two independent components: I (*In-phase*) & Q (*Quadrature*) components [2]. These components are then combined to form the baseband modulating signal.

$$\cos(\alpha+\beta)=\cos(\alpha)\cos(\beta)+\sin(\alpha)\sin(\beta)$$
$$A\cos(2\pi f_c t+\varphi)=A\cos(2\pi f_c t)\cos(\varphi)-A\sin(2\pi f_c t)\sin(\varphi) \quad (1)$$

The conversions to band pass signal increases the performance of the complex demodulator, since its sampling frequency increases. In communication system [7] signal is demodulated in its in-phase and quadrature component. Basically, it consists of splitting the received signal and multiplying each arm by *cos* and *sin* functions. This procedure provides I & Q components after low pass filtering. FIR LPF



$$I = A\cos(\varphi)$$
$$Q = A\sin(\varphi)$$
$$A\cos(2\pi f_c t+\varphi)=I\cos(2\pi f_c t)-Q\sin(2\pi f_c t)$$

passes only the lower frequencies.

Given the low-pass filter which has an impulse response $h_{LP}(n)$ of order N, the output of the in-phase and quadrature channels [1] can be expressed as,

$$I(n)=h_{LP}(n)*\left[x(n).\cos\left(\frac{\pi}{2}n\right)\right]$$
$$=\sum_{m=0}^{N-1}h_{LP}(m).x(n-m).\cos\left(\frac{\pi}{2}(n-m)\right) \quad (2)$$

$$Q(n) = h_{LP}(n) * \left[x(n).\sin\left(\frac{\pi}{2}n\right) \right]$$

$$= \sum_{m=0}^{N-1} h_{LP}(m).x(n-m).\sin\left(\frac{\pi}{2}(n-m)\right) \quad (3)$$

III. ARCHITECTURE

In the proposed architecture ADC output is given to the input memory and its output is fed as the input of two multipliers. These inputs multiplied with the cos and sine LUT generates *I* & *Q* components respectively. It is then filtered out by using filter, whose filter order is 128. The outputs of the filter are stored to output memory. The whole process is controlled by the controller. Fig 1 depicts the implemented architecture.

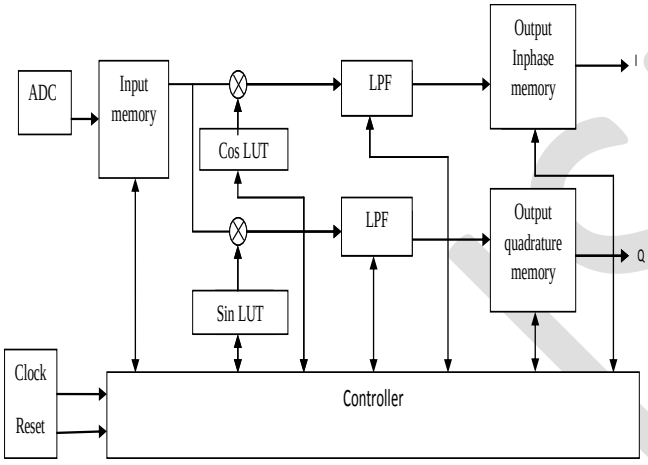


Fig. 1 Architecture of complex demodulator

Xilinx CORE [4] generates and delivers parameterized cores optimized for FPGAs. It is mainly used to create high density, high performance designs in Xilinx FPGAs in less time. Here we are generating different IP core for demodulation process.

The demodulator consist of 16 bit input data and two 32 bit output which are I & Q. Also validated bits are given to both input and output. Start and stop demodulated bit controls the top module initialization and stop.

A. Input Memory

Input memory as in fig 2 is a true dual port ram i.e., ping pong for real time processing which has two independent access ports that permit shared access to a central pool of memory.

Here, inputs are of 1.15 format. It stores and provides input band pass signal for demodulator. Band pass signal is generated by using chirp function in MATLAB™.

The MATLAB generated COE file is then loaded to the block memory generator as input of size 1024*16. The 16 bit

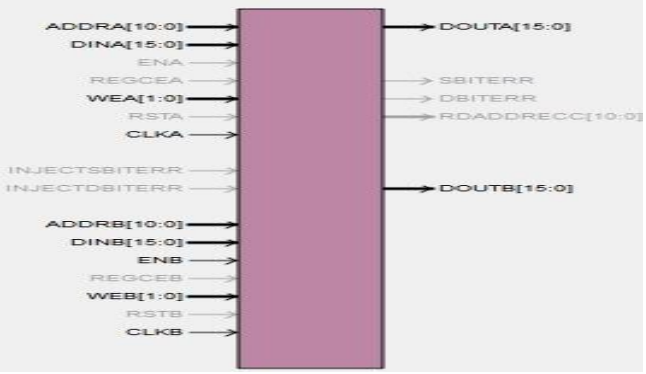


Fig. 2 Input memory block

output is driven by the controller section.

B. Look Up Table (LUT)

The LUT of cosine and sine function by forming a COE file, which stands for coefficient files that contains the contents of the block memory for the specified read depth and read width values of the image. These are then loaded to single port ROM IP core. The LUT consist of a 4 bit data input and a 16 bit output line and clock as shown in the fig 3.

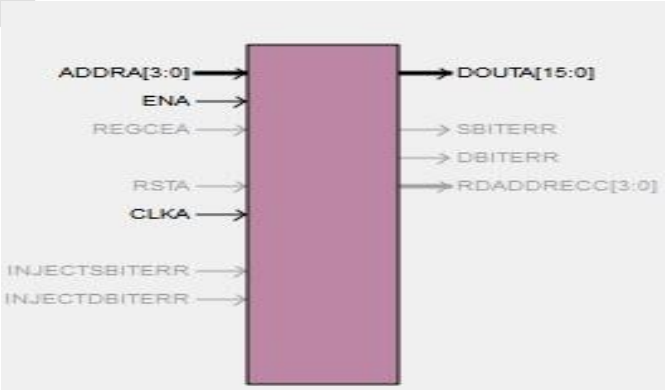


Fig. 3 Single port ROM

C. Multiplier

The output of input memory along with output taken from sin/cos LUT is multiplied which provides 2.30 output format which can be scaled down to 1.15 using right shift arithmetic operation.

D. FIR Filter

Here we are using FIR Low Pass Filter, which filters out higher frequencies at the output and passes only the low frequencies. The filter coefficients are loaded as COE file.

Fig 4 shows the filter response and the basic parameters used here. Using this filter extracted the in-phase and quadrature parts. The filter out obtained in this manner is of 2.30 format.

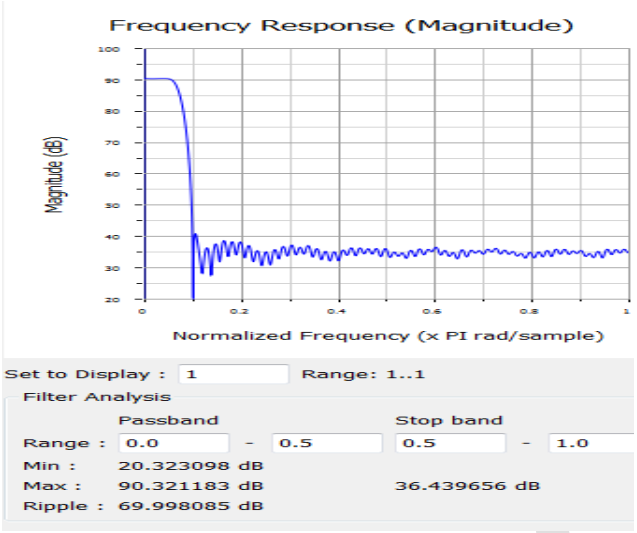


Fig. 4 FIR frequency response

E. Output Memory

It resembles the input memory section which is discussed earlier. The filtered in-phase and quadrature components which are of 2.30 format, are stored in their corresponding in-phase & quadrature output memory. These memory values are finally plotted in MATLAB™ for validating with MATLAB™ model.

F. Controller

Now let us see how the controller controls from start to end of the complex demodulation process based on the structured state machine. The output of the controller decides the functionality of each block.

1) Contoller FSM

State diagram helps in coding the different blocks of the system. Here we are using Mealy diagram for this purpose as shown in fig 5. It consists of different states where functions of different blocks are taken into consideration. Our system can be divided into five different levels and works based on the predicted timing diagram as depicted in fig 6.

Firstly the system will be in initial state before starting the process. Initialization is by clock (Clk), reset (Rst) and start_demod which is provided as a trigger pulse.

Input memory values are read when input mem_web is at logic 0 followed by enabling cos and sin LUT in the read state, else it will remain in the same state itself.

The whole process of multiplication is done in multiplier state. When valid signal corresponding to each multiplier block is active then input memory dout values are fed to the input of multiplier block and then multiplication of LUT data and input memory values takes place in order to produce in-phase and quadrature output .The quadrature output will be 90^{o} out of phase with in-phase.

The fourth state will be initiated when control unit fetches the in-phase and quadrature values to low pass filter. Here, when new data (ND) and ready for data (RFD) are active then these data will be directed to the LPF. After completing the process we get the required in-phase and quadrature filtered output.

Next, in write state Ready (RDY) signal is continuously monitored, when these become active high the filtered in-phase and quadrature values are written to their respective in-phase and quadrature output memory.

Finally when address count (Addr-count) reaches 1023 the demodulation process is stopped and is returned to the initial state.

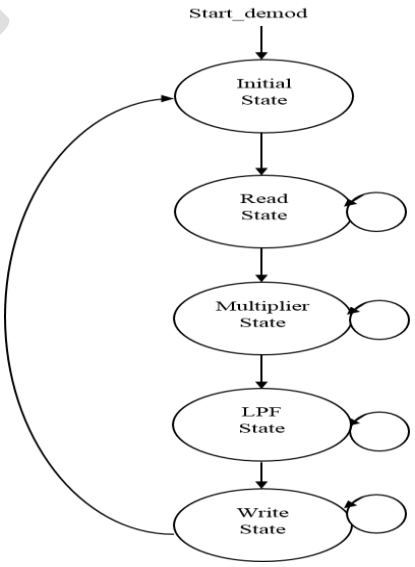


Fig. 5 FSM state machine

IV. SIMULATION

In order to reduce the complexity of translating model to HDL the entire demodulator have been coded in MATLAB™ in order to validate the hardware results by considering the below Table I specification in MATLAB™.

TABLE I: SPECIFICATION IN MATLAB™

Parameters	Symbol	Value
Center frequency	fc	4MHz
Sampling frequency	fs	40MHz
Bandwidth	BW	50% fc
Sampling interval	Ts	25ns
Block size for processing	N	1024samples
Pulse width	PW	0.256us
Filter order	M	128

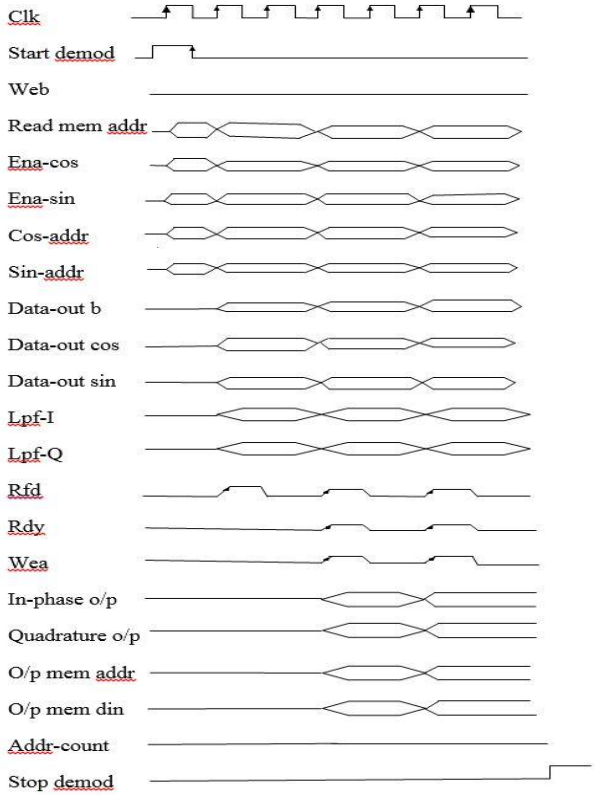


Fig. 6 Predicted timing diagram

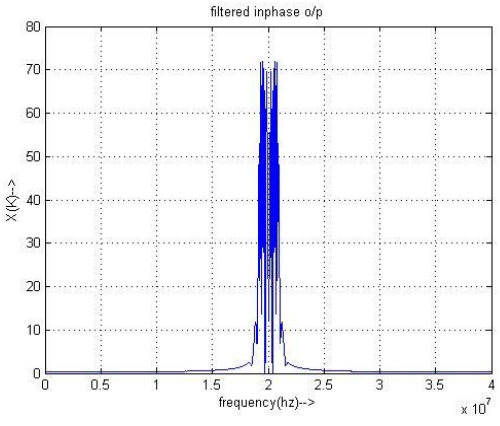


Fig. 7 MATLAB™ in-phase output

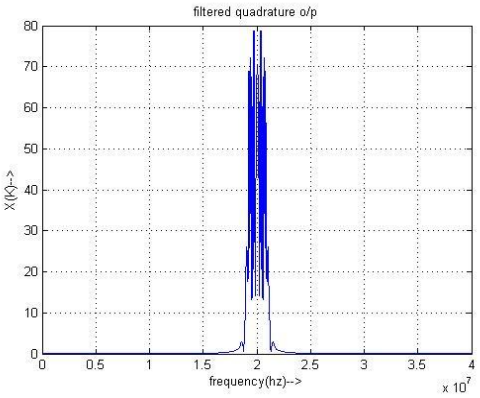


Fig. 8 MATLAB™ quadrature output

In-phase and Quadrature output modeled from MATLAB™ is depicted in fig 7 and fig 8 respectively and fig 9 shows Verilog output.

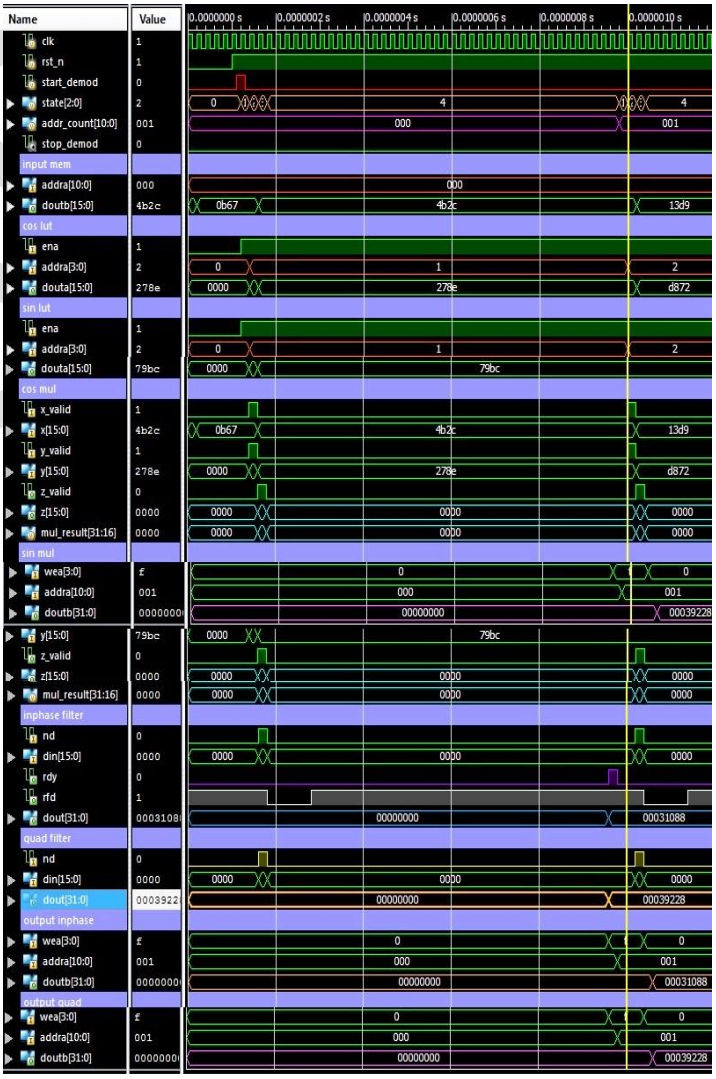


Fig. 9 Verilog output

V. RESULTS

MATLAB™ is used to check algorithm of full system. Each of the output memory values corresponding to in-phase and quadrature phase obtained in Verilog HDL are plotted in MATLAB™ as shown in Fig10 and fig11 respectively.

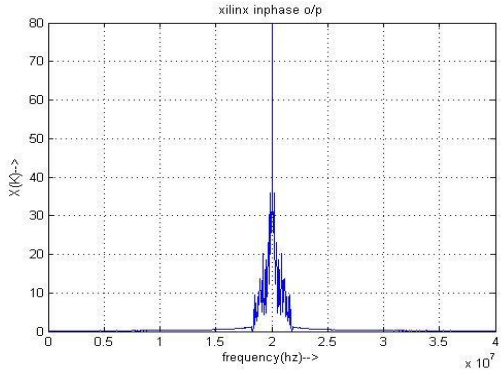


Fig. 10 Verilog HDL in-phase output plotted in MATLAB™

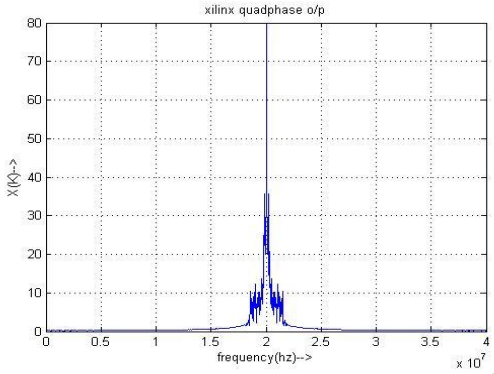


Fig. 11 Verilog HDL quadrature output plotted in MATLAB™

In order to verify the functionality the waveforms are compared with the entire model modeled in MATLAB™ and finally error values are calculated using the following equation,

$$Error = std(X) / sqrt(length(X)) \quad (4)$$

where X is the in-phase or quadrature values.

Calculated error values are indicated in Table II. It is found that error values are smaller for in-phase and quadrature obtained using XILINX ISE®.

TABLE II: CALCULATED ERROR VALUES

Tool	Error values
MATLAB™	In-phase = 0.0104 Quadrature = 0.0107
XILINX ISE®	In-phase = 0.0042 Quadrature = 0.0036

The described architecture has been implemented on Xilinx Spartan-6 SP605 evaluation platform. Table III shows the device utility.

TABLE III: DEVICE UTILIZATION SUMMARY

Slice LUTs	1,399 out of 27,288
Slice Registers	2,814 out of 54,576
RAMB 16B	13 out of 116
RAMB 8K	5 out of 232
IOBs	84 out of 296
Adders/Subtractors	6
Multiplexers	136

VI. CONCLUSION AND FUTURE SCOPE

This paper has demonstrated a novel architecture for complex demodulation. The different modules of demodulator was simulated and finally synthesized and ported the RTL implementation in Xilinx Spartan6 ML605 FPGA platform. The results were proved to be simple, convenient and hardware resource saving with maximum working frequency of 151.777MHz with minimum period of 6.589ns utilizing 8 clock cycles per symbol compared to other conventional implementation methods.

The future work will include increasing the speed of the system by adding pipeline registers in critical path in the filter design.

REFERENCES

[1]. J.M.P. Langlois and D. Al-Khalili, "Polyphase Filter Approach For High Performance, Fpga-Based Quadrature Demodulation" in *Journal of VLSI Signal Processing*, vol, 32, 237– 254, 2002.

[2]. Agilent Vector Signal Analysis Basics Application Note 150-15 literature number 5989-1121EN Palo Alto, CA, July 14, 2004.

[3]. Kerem Karadayi, Cheoljin Lee and Yongmin Kim, "SoftwareBased Ultrasound Beam forming on Multi-core DSPs" *Image Computing Systems Laboratory Departments of Electrical Engineering and Bioengineering University of Washington Seattle*, WA 98195.

[4]. B.Muralikrishna, K.Gnana Deepika, B.Raghu Kanth3 "Image Processing using IP Core Generator through FPGA" in *International Journal of Computer Applications (0975 – 8887)* Vol, 46– No.23, May 2012.

[5]. Bomar B.W."finite wordlength effects" digital signal processing Hand booked. Vijay k. madisetti and Douglas b. Williamsboca raton: crc press llc, 1999.

[6]. Basem Ahmed Hassan, "Low-Cost Digital Ultrasound Beam Former Design using Field Programmable Gate Arrays" *Cairo University* Giza, Egypt 2012.

[7]. David Bernal, Pau Closas, Juan A. Fernandez Rubio, "Digital I&Q demodulation In array Processing: Theory And Implementation*" 16th European Signal Processing Conference (Eusipco 2008)*, Lausanne, Switzerland, August 25-29, 2008.

[8]. H. Samueli and B. C. Wong, "A VLSI Architecture for a High Speed All-Digital Quadrature Modulator and Demodulator for Digital Radio Application," *IEEE J. Select. Areas Commun.*,vol. 53, no.8, pp. 1512–1519, October 1990.

[9]. I. Janiszwski, B.H. Meuth "Numerically controlled oscillators With Hybrid function generators," *IEEE transactions on Ultrasonic's, vol, 49, No.7, July2002*.
[10]. Weinstein, C. and Oppenheim, A.V., "A comparison of roundoff noise in floating-point and fixed-point digital filter realizations," Proc. *IEEE* vol, 57, 1181–1183, June 1969.
[11]. J.Bhasker, B.S.Publications, "AVerilog HDL Primer", 2nd Edition, 2001 ISBN 10: 8178000121 ISBN 13: 9788178000121.
[12]. M .Morrismano, Michael d. Ciletti, "Digital Design" 4th Edition Pearson Prentice Hall 2009 ISBN978-81-317-1450-8.