Beam Forming Algorithm Implementation using FPGA

Arathy Reghu kumar, K. P Soman, Shanmuga Sundaram G.A

Centre for Excellence in Computational Engineering and Networking
Amrita VishwaVidyapeetham, Coimbatore,TamilNadu, India
E-mail : rarathy18@gmail.com, kp_soman@amrita.edu, ga_ssundaram@cb.amrita.edu

***Abstract* – In this paper we are exploring the fundamental theory of beamforming, adaptive beamforming technique and tried to implement one of the adaptive algorithm called Least Mean Square algorithm using Xilinx system generator.The compact structure of FPGA beamformer can thus be implemented on any of the Xilinx FPGA using the generated VHDL code.**

***Keywords* – Beamforming, beamforming algorithms, LMS Algorithm, Xilinx.**

I. Introduction

Beamforming simply means transmit or receiving signals preferred in a particular direction over the other. Beamforming actually taking into account the signal coming into the array of antennas from desired location moreover it nullifies or reduce the effect of undesired signals coming from a different direction [1]. Beamforming can increase the sensitivity of the receiving part along its direction of desired signal, also decrease the sensitivity in the direction of noise. This is a method of spatial filtering[2], that it can remove the interference pattern on desired signal caused due to the unwanted signal effects. Spatial separation can be used to separate the signals from interference by the overlapping signal with same frequency by using spatial filter at the receiving end [3]. Beamforming is a signal processing technique using sensor array at both transmitting and receiving part. In this technique the coherent summation of the signals received by each one of the sensor from the spatial location is achieved [4]. It involves sending signals from each sensor at slightly different time thus all together make the effect of a single powerful beam. Time delay can be replaced by phase shifts in narrowband systems. Beamforming broadly are of two, fixed (conventional) and adaptive type. In fixed type beamformers constant weights and time delays are used. But in adaptive type beamformers it performs adaptive noise cancellation and we are getting optimal type of signals from direction without mechanically steering the array. It does not means we are getting a beam maximum to get an optimum beamforming output, it is obtained by nulling the effects of noisy signals from undesired direction [5]. Adaptive beamformer is a beamforming radar system for signal processing with an array of radar antennas for transmitting and receiving signal without any steering [6]. When compared with data independent beamformers the adaptive ones have better resolution and interference rejection capability. But if any steering vector mismatch occurs it will affect the performance of the system [7].

The main purpose of this paper is to do a discussion regarding the beamforming technique, adaptive beamforming Least Mean Square algorithm and the method to implement it in Xilinx system generator.

The paper starts with a section devoted to briefly explain the concept of beamforming, types of beamforming and in that adaptive beamforming. In the IInd section we are going to discuss adaptive beamforming and one of the commonly used 'Least Mean Square' adaptive algorithm is in detail.

In the IIIrd section we will discuss about the Xilinx system generator and give the implementation tools and problem defing blocks used in Xilinxfor the Least Mean Squarealgorithm.

IVth section explains the probable results of the Xilinx implemention of Least Mean Square algorithm

Vth section is the concluding part with some future work mentioned for radar beamforming using FPGA.

II. Adaptive Beamforming

Adaptive beamforming is one of the most significant beamforming to get the desired output, while also nullify the interference [8]. There are a lot of adaptive algorithms to get the optimum weights, only thing we consider while selection is its performance.

Commonly used adaptive algorithms are Least Mean Square, Sample Matrix Inverse and Recursive least square [9].

LEAST MEAN SQUARE ALGORITHM

Widrow proposed the least mean squares (LMS). The LMS algorithm is based on the minimum mean squares error [9]. It can be used for the purpose of adaptive interference canceling, adaptive beamforming, and adaptive control**.** LMS algorithm is a less randomness adaptive algorithm and can be used for nonstationary signal processing. The performances of the LMS algorithm have been extensively studied. If interference only exists in the output of the analyzed system, the LMS algorithm can only obtain the optimal solutions of signal processing problems [11].

Least mean square algorithm is used to improve the output by updating the weights of the system and thus converge at the optimum weights.

Fig 1 depicts the simple LMS beamforming network. LMS is using the method of steepest descent. Successive correction of the weight vector in the negative direction of the gradient term. For making use of gradient of mean square error function it does not even need squaring averaging or differentiation [12].

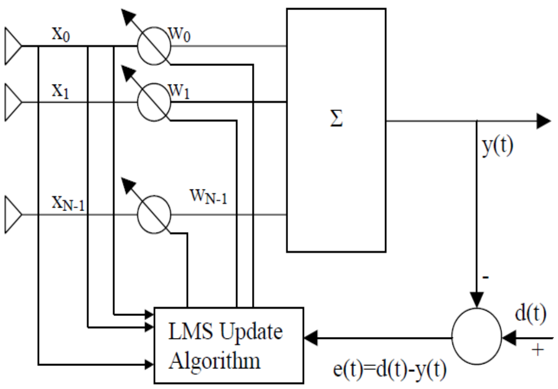

Fig. 1: adaptive beamforming network [13]

By using the steepest descent method

Where φ is the scalar constant for convergence control.

$e^2(n)$is the mean square error between d(n) and y(n)

$$y(n) = w^h x(n) \qquad (2)$$

$$e^2(n) = [\mathrm{d(n)} - w^h x(n)]^2 \qquad (3)$$

$$w(n+1) = w(n) + \frac{1}{2}\varphi[-\nabla(E\{e^2(n)\})] \qquad (1)$$

The present weight vector is termed as w(n) and the successive weight is represented as w(n+1).

Gradient vector in above weight update equation is

$$\nabla_w(E\{e^2(n)\}) = -2r + 2Rw(n) \qquad (4)$$

Unlike in steepest descent method LMS uses instantaneous values of r and R instead of its actual values

$$R(n) = x(n)x^h(n) \qquad (5)$$

$$r(n) = d^*(n)x(n) \qquad (6)$$

$$w(n+1) = w(n) + \varphi x(n)e^*(n) \qquad (7)$$

The updated weight can be written as

$$w(n+1) = w(n) + \varphi x(n)[d^*(n) - x^h(n)w(n)]$$

The algorithm starts with initial weight as zero, with successive updation of weights the mean squared error get minimized. Thus the working of the system can be summarized as:

Output is:

$$y(n) = w^h x(n) \qquad (8)$$

Error is

$$e(n) = d^*(n) - y(n) \qquad (9)$$

Tap weight is

$$w(n+1) = w(n) + \varphi x(n)e^*(n) \qquad (10)$$

If the scalar constant φ is chosen to be small then algorithm converges slowly, large value for φ offers faster convergence [13, 14, 15].

III. XILINX SYSTEM GENERATOR FOR DSP

FPGAs are now more admired than other frameworks like DSPs because of its reconfigurable nature.[16] But problem comes if the user does not have a previous knowledge about HDL coding. As the study of VHDL coding is a tedious task for a beginner, Xilinx system generator is a perfect platform. Previous experience on FPGA and coding is not needed in Xilinx, users can design blocks according to their problem defined using Xilinx blockset. It is able to generate the HDL code and it can then be processed on any Xilinx FPGA.[16,18,19]

Implementation Of The LMS Algorithm Using Xilinx System Generator

The system design diagram includes system generator block. A signal generator is used to give the input signal [16]. State control is given to enable or reset pins. Addressable shift register is there to for input and for updated weight vector. Multiply and accumulate is needed for multiplying the input with weights and accumulating. Error checking block needed to find error value between d(n) and y(n), also multiply it with the scalar constant φ. Weight update for finding out the successive weights.[15,16,18]

The implementation part of the LMS design is done using the MATLAB 2011 and the Xilinx ISE 13.1 version. The block mentioned below is created using the specific blockset in system generator. The clock rate for FPGA is set as 1 MHZ.

Delays are provided in state control. It can be set by setting the latency value. We can made necessary changes for each block by clicking and opening the blocks, then do the editing. The content of addressable registercan be read using the counters provided. Number of coefficients in the weight vector decides the counter depth. The input here is a sine wave.

.

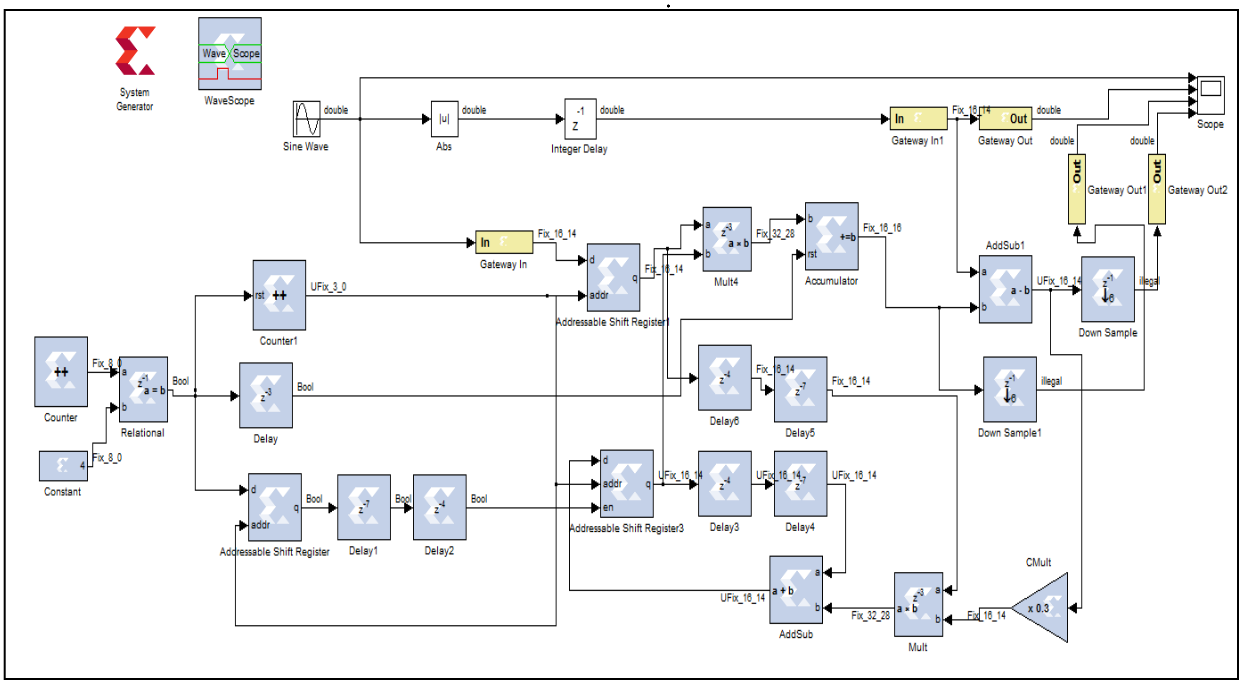

Fig. 2: Block diagram of the LMS adaptive algorithm implemented using Xilinx blockset [16]

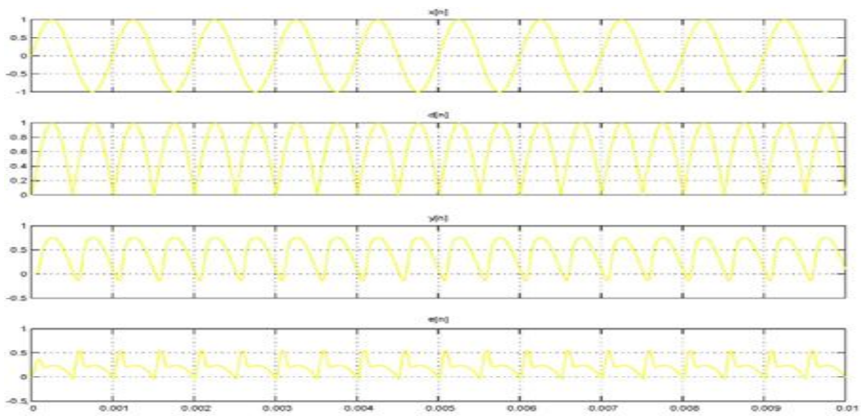

Fig 3: Represents the input signal x(n), and the desired signal response from an unknown system d(n), Out response by multiplying and accumulating y(n) and finally the error e(n). [16]

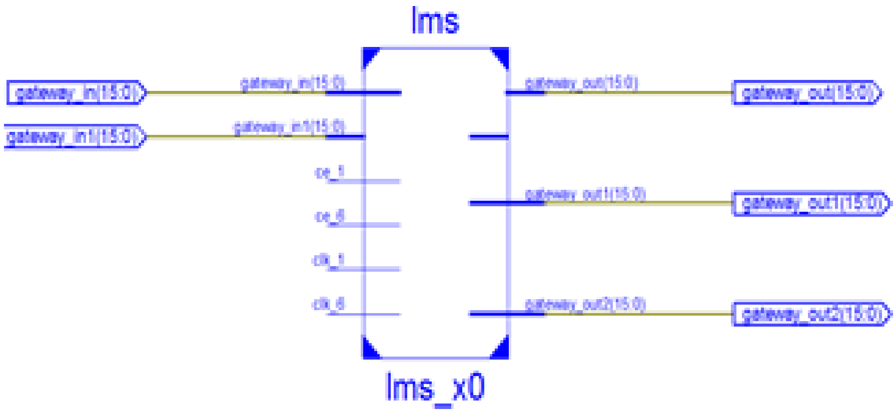

Fig. 4: Represents the top level RTL view of LM

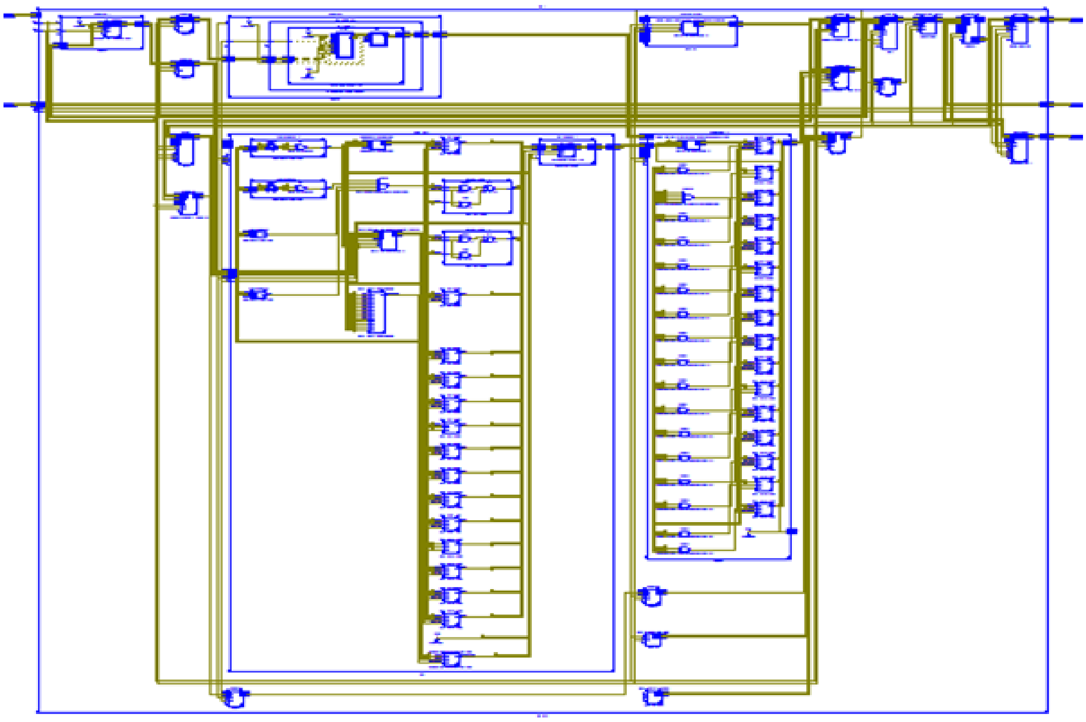

Fig. 5: Internal structure of LMS in RTL view

.

IV. Conclusion And Future Work

In this paper we have explored the theory behind the beamforming. LMS algorithm working for adaptive beamformers. Steps to implement this LMS algorithm on a Xilinx platform. Xilinx system generator is very helpful in designing real time implementations like the beamforming concept used RADAR technology.

Xilinx will automatically generate a VHDL code for the corresponding block set which we have designed in thesystem generator. This generated code can be further used to do the hardware implementation part of FPGA beamformer on any Xilinx FPGA.

Here we have discussed onlya single algorithm, but we can extend this work by creating blocksets for other recent adaptive algorithms and thus implementation on Xilinx FPGA using the generated VHDL code. Thus it will be worth while if we can do the FPGA beamforming for the RADAR technology.

V. REFERENCE

[1] A Compact FPGA Beamformer Architecture,Ioan Lie, MihailEugenTanase,2005 WSEAS Int. Conf. on DYNAMICAL systems and control (pp463-466),Venice,Italy, November 2-4, 2005

[2] Laboratory technique Spatial filtering http://cem01.ucsd.edu/~vitaliy/courses/ece182/18206files/SpatialFiltering.pdf

[3] Beamforming a versatile approach to spatial filtering, Barry D.VanVeen and Kevin M Buckley, IEEE ASSP MAGAZINE APRIL 1988

[4] Beamforming for imaging a brief overview, JanEgilKirkebo,.

[5] Antenna arrays and beamforming ch3.pdf http://scholar.lib.vt.edu/theses/available/etd-04262000-15330030/unrestricted/ch3.pdf

[6] xhttp://en.wikipedia.org/wiki/Adaptive_beamformer

[7] Kim.J.W and C K.UN An adaptive array robust to beam pointing errors, IEEE Trans Signal processing June 12, 2009

[8] "A novel digital beamformer with low angle resolution for vehicle tracking radar," Singh, A. K., P. Kumar, T. Chakravarty, G. Singh, andS. Bhooshan, Progress In Electromagnetics Research, PIER 66, 229–237, 2006

[9] A simple adaptive beamforming algorithm with Interference suppression, K.Meena alias Jeyanthi1, Dr.A.P.Kabilan2 June 12, 2009

[10] Total least mean squares algorithm Da-zheng feng, zheng bao, senior member, ieee, and li-cheng Jiao, senior member, IEEE transactions on signal processing, vol. 46, no. 8, august 1998

[11] B widrow adaptive filters in aspects of networks and system theory,N de Claris and E kalman Eds. New York: Holt, Rinehart, and Winston, 1971

[12] Interference-Normalized Least Mean Square Algorithm Jean-Marc Valin, Member, IEEE, and Iain B. Collings, Senior Member, IEEE signal processing letters, vol. 14, no. 12, December 2007

[13] Least Mean Square algorithm Ch_6lms.pdf K.K.Shetty,2004

[14] FPGA implementation of Adaptive LMS filter S.S Godbole1, P.M Paldoskar2, VP Route3, proceeding of SPIT-IEEE colloquim and International conference Mumbai, India Vol. 2,2 26

[15] Adaptive filters, Bernard Widrow. adaptive filters.pdf 1971

[16] Implementation of System Identification with Pipelined LMS Adaptive Algorithm using Xilinx System Generator for DSP Muhammad Sulman and Syed AmjadHussain Shah,2010

[17] FPGA Implementation of NLMS Algorithm for Receiver in wireless communication system, IJCSNS International Journal of Computer Science and Network Security, VOL.10 No.12, December 2010

[18] A pipelined LMS adaptive FIR filter architecture without adaptation delay, Douglas, ScottCSignal Processing, IEEE Transactions Mar1998.

[19] Virtex FPGA implementation of a pipelined adaptive LMS predictor for electronic support measures receivers Ting, Lok-Kee K. Jan. 2005

◈◈◈