FPGA Implementation of Adaptive Beamforming in Hearing Aids

Kartik Samtani, Jobin Thomas, G. Abhinav Varma, Sumam David S., *Senior Member, IEEE* and Deepu S. P.

***Abstract*— Beamforming is a spatial filtering technique used in hearing aids to improve target sound reception by reducing interference from other directions. In this paper we propose improvements in an existing architecture present for two omnidirectional microphone array based adaptive beamforming for hearing aid applications and implement the same on Xilinx Artix 7 FPGA using VHDL coding and Xilinx Vivado® 2015.2. The nulls are introduced in particular directions by combination of two fixed polar patterns. This combination can be adaptively controlled to steer the null in the direction of noise. The beamform patterns and improvements in SNR values obtained from experiments in a conference room environment are analyzed.**

I. INTRODUCTION

Beamforming is used in hearing aids to create listening sensitivity pattern which gives higher sensitivity in the direction of target sound while attenuating the interference from other directions. The array of omnidirectional microphones enhances the intelligibility of speech in the noisy environment. Other methods for improving the signal to noise ratio in hearing aids [1], rely on differences in the speech and noise either in time or in frequency domain. These methods give degraded performance in speech-like noise environments as there is very little difference between speech and noise in both time and frequency domains. Since there is difference between noise and speech signals in the spatial domain (direction of arrival of speech and noise), processing in spatial domain may overcome this drawback.

Two main approaches for directional noise cancellation are having a fixed beamform pattern cancelling sounds from a particular direction all the time [2], [3] or having an adaptively varying beamform pattern based on the direction of noise [4], [5]. Increasing the number of microphones in the array improves directional noise cancellation, but it increases the hardware complexity. Hence we implement the beamformer using an array of two omnidirectional microphones to keep the hardware relatively simple. The implemented architecture introduces null in the direction of noise using adaptively controlled combination of two fixed polar patterns.

Most of the hardware implementation of beamforming algorithms for hearing aids use Digital Signal Processors [5] and not much literature is available for Field Programmable Gate Array (FPGA) based implementation of such algorithms. The FPGA implementation helps in prototyping the hardware and at the same time allows to explore hardware level improvements. In this paper we propose modifications to relax the computational speed without affecting the performance and the real time operation. This helps in reducing the dynamic power of the system. This paper also addresses some practical issues in implementing a two omnidirectional microphone array beamformer and also tries to equalize the frequency response introduced by the beamformer using a simple Infinite Impulse Response (IIR) filter. The paper is organized as four sections. Section II aims at analyzing the proposed system theoretically, section III gives the implementation details of the architecture and section IV presents the results obtained from the tests conducted.

Authors are with the Department of Electronics and Communication Engineering, National Institute of Technology Karnataka, Surathkal, Mangalore 575025, INDIA. (corresponding author e-mail: samtanikm@gmail.com).

II. PROPOSED SYSTEM

The proposed adaptive beamforming architecture based on the system proposed by Luo et al. [5] is shown in Fig. 1. The incoming sound signal from direction θ is represented as $s(n)$. Signals $a(n)$ and $b(n)$ correspond to the sound inputs to the two omnidirectional front and back microphones respectively in the array. The direction directly in front of the hearing aid user is represented as $0°$, whereas $180°$ represents the direction directly behind the user. Nulls are obtained in different directions (for different values of θ) by varying the gain $G(n)$, by the adaptive logic block. The distance between the front and back microphones is denoted as d and speed of sound is denoted as c. The delay T is taken as d/c. The polar pattern of $x_1(n)$ is cardioid with a null at $180°$ and that of $x_2(n)$ is cardioid with null at $0°$. The relationship of the null of the system output $y(n)$ with the gain $G(n)$ is given as [5]

$$G(n) = \frac{sin(\pi f \frac{d}{c}(1 + cos\theta_{null}))}{sin(\pi f \frac{d}{c}(1 - cos\theta_{null}))} \tag{1}$$

where f is the frequency of the signal and θ_{null} is the angle of the null along the line between the two microphones. It

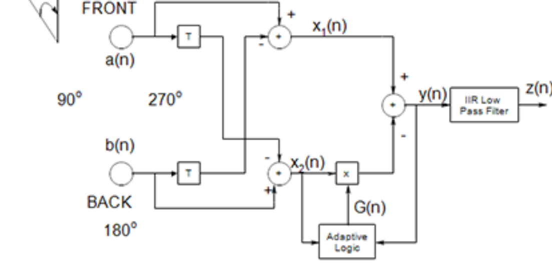

Fig. 1. Model of the Adaptive Beamformer

978-1-5090-2809-2/17/$31.00 ©2017 IEEE

can be seen for all frequencies if the direction of incoming sound $\theta = 180°$, then the null is located at $180°$ by setting $G(n)$ = 0. Similarly, null is placed at $\theta = 90°$ or $270°$ by setting $G(n)$ = 1. With the approximation $sin\theta \approx \theta$, for the frequency range of interest, Luo et al. [5] has shown that $G(n)$ can be approximated to

$$G(n) = \frac{1 + cos\theta_{null}}{1 - cos\theta_{null}} \tag{2}$$

Therefore the relationship between the null and the gain will be independent of the frequency. Varying $G(n)$ in the range from 0 to 1 places nulls in region corresponding to $90° \leq \theta \leq 270°$. It is assumed that the noise is received only from this region and that the target signal is received from the region $0° \leq \theta \leq 90°$ and $270° \leq \theta \leq 360°$ (i.e. from the direction in which the front microphone is facing). With this limit on $G(n)$, if $G(n)$ is varied to minimize the power of $y(n)$, we place nulls in direction of dominant sources of noise without attenuation of the target signal. Least Mean Squares (LMS) (stochastic gradient descent) algorithm [5], [6] is used to update the value of $G(n)$ to minimize the power in array output $y(n)$ and hence steering the null adaptively in the direction of noise. Using LMS algorithm the adaptive gain can be obtained as

$$G(n + 1) = G(n) + 2\mu y(n)x_2(n) \tag{3}$$

where μ is the step parameter also known as learning rate. This is implemented in the adaptive logic block. For sound coming from direction $\theta = 0°$, there is no directional cancellation of sound. Under this condition the relation between the desired sound source and the array output can be expressed as [4]

$$\left|\frac{Y(\omega, \theta)}{S(\omega)}\right| = 2\left|sin(\frac{\omega d}{c})\right| \tag{4}$$

In most of the hearing aid applications d is small when compared to the wavelength of the sound. Under these conditions, the argument of the sine function is close to 0. Hence first term of Taylor series expansion can be used to approximate $sin\theta = \theta$. In this case the output of directional microphone is proportional to ω which gives a magnitude response with slope of 6 dB/octave [6]. Also, the output is proportional to d, which means that reducing the separation between the microphones reduces the output level. A first order IIR low pass filter (5) is used for equalizing this high pass response from 100 Hz to 2 kHz (low frequencies).

$$H(z) = \frac{C_1 + C_2.z^{-1}}{1 - C_3.z^{-1}} \tag{5}$$

With reduction in the value of d, the range of this band to be equalized increases. The output of this filter $z(n)$ is the final output as shown in Fig. 1.

III. Implementation

The inputs of the two microphones have to be delayed by $T = d/c$ as shown in Fig. 1. Hence the minimum sampling frequency is chosen as $F_s = c/d$ so that the incoming signal can be delayed by one sample to obtain the required delay. We use 16 bit, 2's complement fixed point arithmetic (14 bit fractional part) for doing the processing in hardware. The architecture as shown in Fig. 2 consists of a fixed beamformer module, an LMS module and a low pass equalizer filter. The output of fixed beamformer is denoted as $y(n)$, output of low pass filter as $z(n)$ and the output of LMS module as $G(n)$. Fixed beamformer and LMS modules are each implemented as four state FSM and run in parallel synchronously to do adaptive beamforming. Low pass filter is implemented as an IIR filter. Using (5),

$$z(n) = C_3.z(n-1) + C_1.y(n) + C_2.y(n-1) \tag{6}$$

It consists of multipliers, adders and a register to store previous value of the output $z(n-1)$ for computing $z(n)$. The IIR filter was designed using Matlab and the coefficient values used are C_1 = 0.2759, C_2 = 0.2759 and C_3 = 0.9758.

The processing has been done in such a way that the output is delayed by one cycle of the ADC clock with respect to input. Since our microphones are about 1.2 cm apart and with $c = 340\ m/s$, the minimum sampling frequency we can have is about 28 kHz. This is a very high sampling frequency and does not give us much information as typical range of voice frequencies is from 200 Hz to 4 kHz. Moreover this requires the computations and the arithmetic operations to run at a higher rate. Hence we downsample the signals $a(n)$ and $b(n)$ by 2 after the required delay. This helps to reduce the speed of computation without any loss of information. This also brings down the dynamic power dissipation of the hardware. If the downsampler was not introduced in the architecture, both the FSM, LMS module and the fixed beamformer module would have required a clock of frequency which is 4 times the ADC sampling frequency or $4c/d$ for the entire system to operate in real-time. This is because all four states of the FSM must be executed before the arrival of new sample i.e., within a single cycle of the ADC clock. In presence of the downsampler, this frequency reduces to

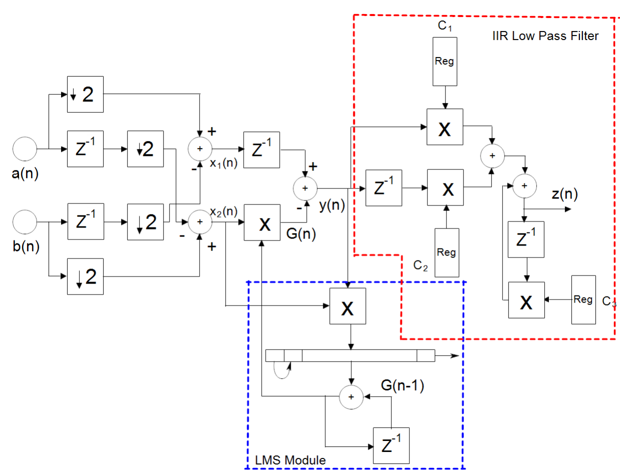

Fig. 2. Hardware Realisation of the model to be Implemented

$2c/d$ (4 times $c/2d$).

Radix-2 Booth's multiplier was used for signed multiplication. It takes around 16 cycles to multiply two 16 bit fixed point numbers. Since each multiplication needs to happen within one state of the FSM, the multipliers need to run at a clock with frequency 16 times FSM clock or $32c/d$. The clock dividers are used to provide the desired clocks to different units. The LMS module is responsible for adaptively varying the gain $G(n)$. The inputs are two 16-bit numbers, namely $x_2(n)$ and $y(n)$. It uses a Booth's multiplier to multiply $x_2(n)$ and $y(n)$, the product is then shifted to the right by 2 bits, to multiply with $2\mu = 0.25$ and is added to the previous value of gain, $G(n-1)$ stored in a register to get $G(n)$ (3).

IV. Experimental Testing and Results

Experiments were conducted in a conference room environment with the distance between the microphones as 1.2 cm. The microphones used in the experiment were ADMP401 MEMS omnidirectional microphones, with a flat frequency response from 100 Hz to 15 kHz. It gives a 200 mV peak-peak output when recorded at arms-length, so the distance at which the source of sound was placed for the experiments was around 50 cm. The sound data, $a(n)$ and $b(n)$, are acquired from the microphones using an ADC with 16 bit resolution at 28 kHz sampling rate.

The architecture described in section III was implemented on Xilinx Artix 7 FPGA (Xilinx Nexys 4 FPGA board with xc7a100tcsg324-1) using Xilinx Vivado® 2015.2. All the analysis on the output acquired from the system was carried out using MATLAB® 2015.

The high pass nature of the beamformer was tested with 900 Hz and 1.8 kHz monotones. It can be seen from Table I that the input to output gain increases by 6 dB on changing from 900 Hz to 1.8 kHz when the source is located at 0° azimuth. The corresponding equalized gain is also shown in Table I and the difference between the input to output power gain, introduced due to the high pass frequency response of the beamformer, with 900 Hz and 1.8 kHz monotones has considerably reduced from about 6 dB to about 0.47 dB with the gains being positive.

Since the microphones are at a distance of 1.2 cm, when the sound comes from 180° azimuth, the back microphone records the sound with higher power than the front microphone as the sound attenuates over the distance. Effect of distance on the input signal can be observed in Table II, here the same signal (1.8 kHz sinusoid) is given from two different directions (90° and 180°). When sound is incident at 90°, both the microphones record the sound with almost same power, but this does not happen when sound is arriving from 180°. Due to this, we had to pre-process the acquired data by making their powers equal. To remove this effect, d must be reduced.

TABLE I
Frequency Response of Beamformer

Frequency (Hz)	Gain Without Equalizer	Gain With Equaliser
900	-8.9757	0.5690
1800	-2.2512	1.0425

TABLE II
Signal Power Recorded by Microphone Dependent on the Direction of Reception of Sound

Direction of Incoming Signal (Degrees)	Power from front Microphone	Power from back Microphone
90	0.0131	0.0128
180	0.0105	0.0127

In order to check the convergence of the adaptive gain to the desired value, monotone of frequency 1.8 kHz was played from different directions ($90° \leq \theta \leq 270°$) with respect to the microphones. Fig. 3 shows the convergence of the gain $G(n)$ for 1.8 kHz monotone arriving from 180°. We see that the adaptive gain tries to converge to 0 as expected for the noise from 180°. Table III shows the convergence of the adaptive gain for a 1.8 kHz monotone arriving from other angles to the beamformer.

The polar pattern with nulls at 90°, 120° and 180° were obtained from the system implemented and are shown in Fig. 4. The radius of polar plots in Fig. 4 represents the ratio of output power to input power. Patterns obtained do match the expected patterns [5].

An experiment was conducted, in which the desired speech signal was arriving from a source at 0° azimuth, while noise was arriving from different directions. Two kinds of noise were used to test the algorithm, a 1.8 kHz monotone and

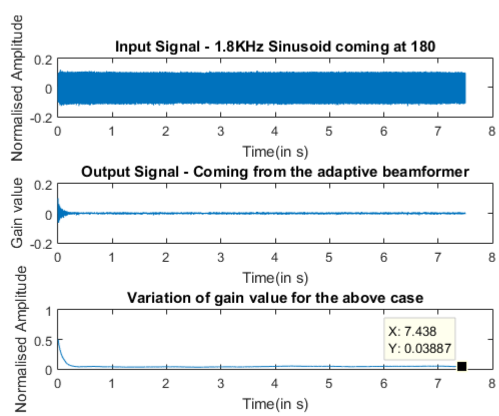

Fig. 3. Convergence of G(n) for 1.8 kHz monotone from 180^0

TABLE III
Convergence of G to Expected Values

Direction of Incoming Signal (Degrees)	Value of G	Noise Attenuation (dB)
90	1.0000	-23.10
120	0.3612	-22.95
180	0.03887	-21.81

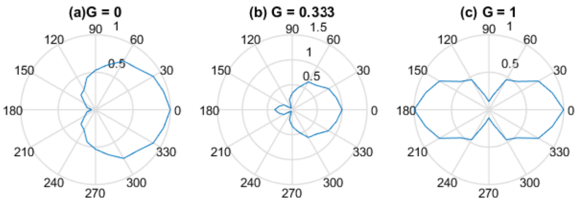

Fig. 4. Directional Response with values of G being (a)0 (b) 0.333 (c)1

another speech signal. The Table IV also shows the SNRe values obtained in each case. Here,

$$SNRe = SNR_{Output} - SNR_{Input} \tag{7}$$

It can be seen that the output coming from the beamformer has significant reduction in the noise and that the noise reduction is better in case of stationary noise.

The architecture was implemented on Xilinx Artix 7 FPGA (xc7a100tcsg324-1) using Xilinx Vivado® 2015.2. The resource utilization is shown in Table V. Xilinx Vivado® 2015.2 was used for analyzing and estimating the reduction in dynamic power consumed by the hardware after incorporating the downsamplers in the architecture. The dynamic power consumption reduced by a factor of 1.57.

V. Conclusion

Adaptive beamforming was implemented on Xilinx Artix7 FPGA. The high pass nature of the beamformer was compensated by using an IIR low pass filter. The practical issues due to the distance between the microphones were accounted for and taken care by preprocessing the input before giving it to the beamformer. Hence the results were similar to the theoretically expected ones. The downsampler was used and was successfully able to reduce the computations in the algorithm and reduce the dynamic power. The distance between the microphones must be reduced for avoiding any preprocessing. The future work may involve splitting $x_1(n)$ and $x_2(n)$ into separate frequency bands with $G(n)$ adapting independently in each frequency band.

TABLE IV
Improvement in SNR Values

Type of Noise Signal	Direction of Incoming Noise (Degrees)	SNRe (dB)
Monotone Signal	180	20.81
	150	22.10
	105	21.21
Speech Signal	180	15.08
	150	15.12
	105	14.92

TABLE V
Resoource Utilisation

Resource	Utilisation	Available	Utilization Percentage
FF	130	126800	0.10
LUT	552	63400	0.87
I/O	48	210	22.86
BUFG	8	32	25

Acknowledgment

The authors would like to thank Centre for System Design, National Institute of Technology Karnataka, Surathkal for the support provided in data acquisition.

References

[1] B. Widrow and F.-L. Luo, “Microphone arrays for hearing aids: An overview,” *Speech Communication*, vol. 39, no. 1, pp. 139–146, 2003.
[2] J. G. Desloge, W. M. Rabinowitz, and P. M. Zurek, “Microphone-array hearing aids with binaural output. i. fixed-processing systems,” *IEEE Transactions on Speech and Audio Processing*, vol. 5, no. 6, pp. 529–542, 1997.
[3] W. Soede, A. J. Berkhout, and F. A. Bilsen, “Development of a directional hearing instrument based on array technology,” *The Journal of the Acoustical Society of America*, vol. 94, no. 2, pp. 785–798, 1993.
[4] G. W. Elko and A.-T. N. Pong, “A simple adaptive first-order differential microphone,” in *Applications of Signal Processing to Audio and Acoustics, 1995., IEEE ASSP Workshop on*. IEEE, 1995, pp. 169–172.
[5] F.-L. Luo, J. Yang, C. Pavlovic, and A. Nehorai, “Adaptive null-forming scheme in digital hearing aids,” *IEEE Transactions on signal processing*, vol. 50, no. 7, pp. 1583–1590, 2002.
[6] J. M. Kates, *Digital hearing aids*. Plural publishing, 2008.