Design, Simulation and Implementation of an Active Sound-Noise Cancellation System for Use in a Cockpit Intercommunication System

O. Sharifi-Tehrani[1]

Department of Electrical Engineering,
Najafabad Branch,
Islamic Azad University, Isfahan, Iran.
omidsht@sel.iaun.ac.ir

ABSTRACT

In this paper, an active noise control system for denoising the intercommunication signal of an airplane cockpit is proposed. Noise sources such as engines degrade the quality of the intercommunication signal, especially in the case of the pilot and copilot headsets. A two-microphone active adaptive noise controller is designed by using an adaptive FIR filter in an active structure. The designed system is simulated and also implemented in real environment using real speech signals, periodic noise and AWGN noise. Also, an FPGA-based hardware implementation utilizing a novel method is provided. The whole design is considered an FPGA hardware core with low resource utilizations which is suitable for HW/SW codesign and System-on-Programmable-Chip (SoPC) applications. The codes have been written by using the VHDL93 hardware description language, the XilKernel embedded operating system and a finite state machine. The obtained results showed competent functionality and performance of the proposed system. This ICS noise removal architecture can be used on any cargo, civil or fighter platform (such as C-130, IR-AN 140 and F5-F) and also in radar and electronic warfare (EW) systems (for clutter/interference compensation) with minimum hardware or software changes.

Keywords: active noise control, sound denoising, adaptive FIR filter, intercommunication system, airplane cockpit, FPGA, electronic warfare.

1. Introduction

The advances in adaptive signal processing/transmission and the problem of environmental noise/interference have made signal enhancement an important subject. Environmental noise and interference increase the probability of error and degrade accuracy in applications that include tasks such as speech identification, thus decreasing functionality [1]. It is possible to optimize the input speech signal by means of special techniques with the goal of expediting further reduction in noise and interference. This is usually referred to as *preprocessing* [2]. Reduction or elimination of noise is one of the most important applications of adaptive active filters. There are many situations in which old filtering techniques for noise reduction are not applicable. Old filtering systems are bulky, have hundreds of coefficients and are not efficient in lower frequencies [3]. Adaptive-active filters, on the other hand, are not dependent on the characteristic of the noise source or interference once their parameters are adjusted for a dedicated task. These filters generate an equal but opposite phase signal which acts on the noise signal to reduce or cancel it [4]. Using the optimized Wiener filter theory is an old and fundamental way to reduce noise. Just as those based on adaptive algorithms such as least mean squares (LMS), this technique can be adapted to dynamic noise environments [5]. The employment of spectral subtraction has become a common digital method for speech signal noise reduction. Nonlinear spectral subtraction is an effective and popular method for speech enhancement [6]. Using arrays of microphones is another way to reduce noise [7]. Here we discuss the noise reduction effect of an adaptive filter based on the least mean square algorithm on the Persian language speech signal. A noisy Persian speech signal is given to the system as input; then the system tries to identify and simulate parameters of the noise source and to generate a similar signal with an opposite phase by employing the LMS algorithm. This generated signal is subtracted from the noisy signal and the latter is thus denoised. The functionality of the system is evaluated by the number of weights, speed of convergence and training time.

2. Adaptive FIR filter and LMS algorithm

As Figure 1 shows, delayed inputs are multiplied by their respective coefficients to form the output. The real output is compared with the desired output and the error signal is calculated. The error signal is used with the learning rate so that the system weights are updated and thus the error is minimized (in fact the real output signal is intended to be as close as possible to the desired output signal). The minimization of error is done by updating the system weights and getting them as close as possible to an optimal combination. Choosing an appropriate value for the adaptive learning rate and choosing a proper number of system weights are in fact of prime importance, as well as tapped delay lines. The latter can be a determining factor for the functionality of the system. If the number of coefficients is too small, the noise reduction functionality could be seriously compromised. From Figure 1, the output can be calculated as

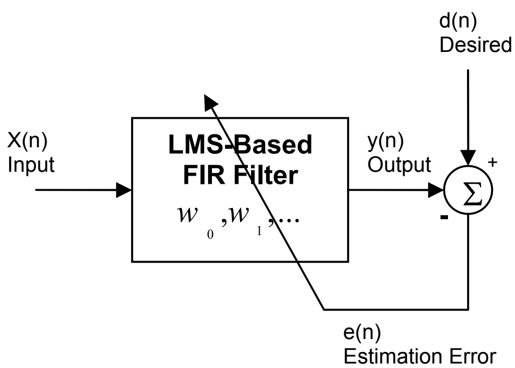

Figure 1. Adaptive FIR filter architecture.

$$Y = V = \sum w_i . x_i = W.X \tag{1}$$

where X is the input vector and W is the coefficient vector. Based on the LMS training algorithm:

$$W(m+1) = W(m) - \beta \frac{\partial F}{\partial W(m)} \tag{2}$$

$$F = E^2 = (d-y)^2 \tag{3}$$

where F is the performance index, d is the desired output, and y is the real output [8, 9]:

$$W(m+1) = W(m) - \beta \nabla F |_W \tag{4}$$

In Equation (4), $\nabla F |_W$ is the performance index gradient vector.
Assuming the P-dimension input, the error (E) can be calculated as

$$E = (d-y) = d - \sum_{j=1}^{p} w_j . x_j \tag{5}$$

Thus,

$$\frac{\partial E}{\partial W_i} = -X_i \tag{6}$$

Then:

$$\frac{\partial F}{\partial W_i} = -2E.X_i \tag{7}$$

$$W_i(m+1) = W_i(m) + 2\beta.E.X_i(m) \tag{8}$$

By replacing $\eta = 2\beta$, the learning rate, and $\delta = E$, the error, in Equation (8):

$$W_i(m+1) = W_i(m) + \eta.\delta.X_i(m) \tag{9}$$

After passing through several epochs, the architecture tunes the coefficients for an optimum combination for noise/interference reduction. The adaptation formula for tuning the coefficients is as that in Equation (9). The leaky LMS algorithm can be implemented by incorporating coefficient α into Equation (9). This improves the architecture stability and expedites its movement toward an optimum point [10, 11]:

$$W_i(m+1) = \alpha W_i(m) + \eta.\delta.X_i(m) \tag{10}$$

Analytical algebra shows that the maximum value of α, which allows a stable behavior of the LMS algorithm, is

$$\alpha \leq \frac{1}{\lambda_{Max}} \quad (11)$$

Here λ_{Max} is the largest eigenvalue of the data correlation matrix:

$$\lambda_{Max} = max\,(eigenvalue(X^T X)) \quad (12)$$

The topology used in this paper is shown in Figure 2. The initial input includes the desired speech signal $s(k)$ and the noise $n(k)$. The reference input, however, includes only the $n_R(k)$ noise. The $n(k)$ noise is not exactly the same as the $n_R(k)$ noise (it has been filtered and attenuated by the noise path). There is also a slight delay due to acoustic propagation.

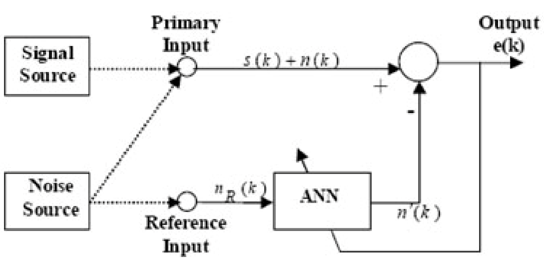

Figure 2. Proposed system topology.

To achieve maximum reduction, a smart adaptive and active topology should be used with the goal of making the $n_R(k)$ noise and the $n(k)$ noise converge as much as possible. From the LMS algorithm [12] we have

$$E[e^2(k)] = E[(s(k)+n(k)-n'(k))^2] \quad (13)$$

Assuming $s(k)$, $n(k)$ and $n_R(k)$ to be signals with zero mean and assuming $s(k)$ to be independent of $n(k)$ and $n_R(k)$:

$$E[s(k)(n(k)-n'(k))] = 0 \quad (14)$$

And

$$E[e^2(k)] = E[s^2(k)] + E[(n(k)-n'(k))^2] \quad (15)$$

Considering the fact that the reference input signal has no information about the $s(k)$ signal, the minimization of the value of $E[e^2(k)]$ will only affect the value of its second term. In other words, minimizing $E[e^2(k)]$ is the same as minimizing the difference between $n(k)$ and $n'(k)$, resulting in the reduction or cancellation of the noise signal and leaving the speech signal (the desired signal) as the error $e(k)$ at the output [13-15]. The following delay-attenuation model is used for modeling the acoustic propagation path:

$$Y(n)_{\text{Propagated}} = (1-A)Y(n-\Delta)_{Source} \quad (16),$$

where $Y(n)_{\text{Propagated}}$ is the signal propagated through the acoustic path, $Y(n)_{Source}$ is the main, unpropagated signal, A is the attenuation factor for the propagation path, and Δ is the propagation path delay.

3. Hardware synthesis

The system was designed to be assembled and tested on a C-130, IR-AN 140, or F5-F platform but, due to some restrictions and difficulties (by aviation industries), it was synthesized and tested in a real environment with noise propagated through the acoustic path. Figure 3 shows the situation used for experimenting. The number of coefficients selected was 64 and the learning rate was 0.0002. The results obtained for the sine noise (at different frequencies) are depicted in Figure 4 to Figure 8 and Table 1.

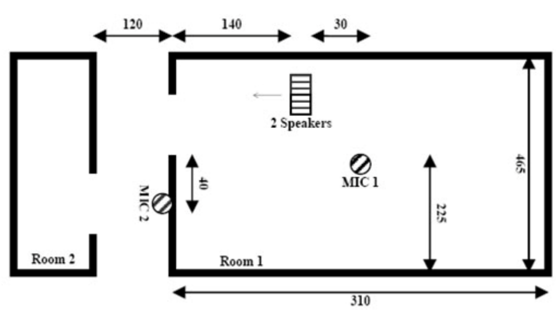

Figure 3a. Test location dimensions (in centimeters).

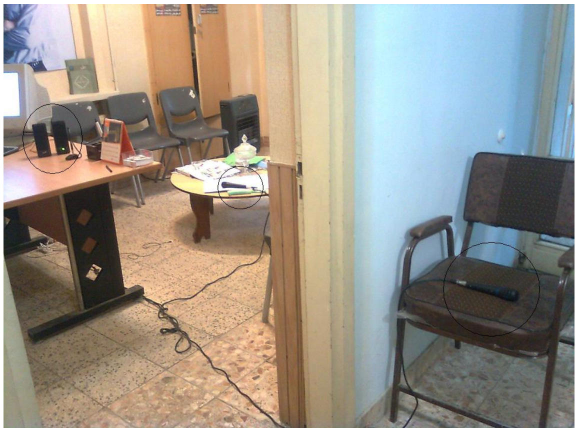

Figure 3b. View of test location.

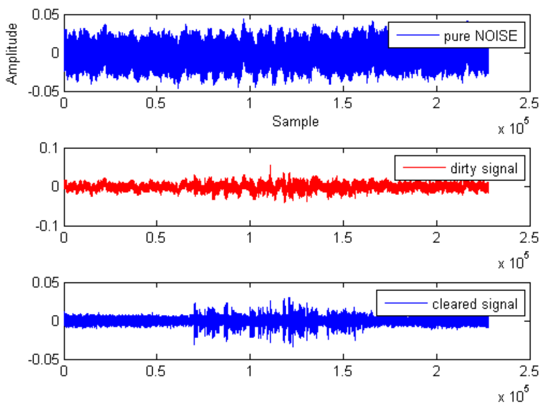

Figure 4. Noise frequency is 50 Hz.
Amplitude in terms of the sample.

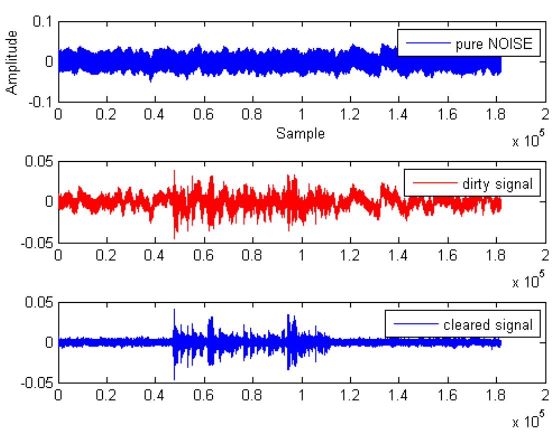

Figure 5. Noise frequency is 100 Hz.
Amplitude in terms of the sample.

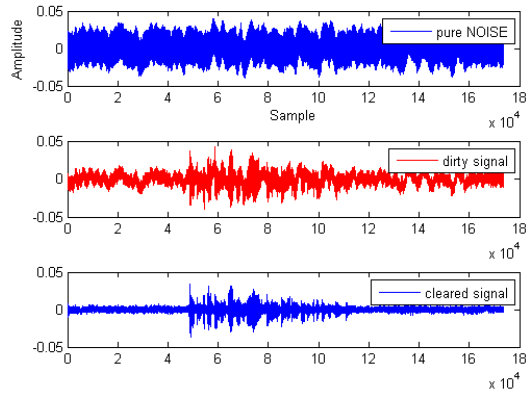

Figure 6. Noise frequency is 500 Hz.
Amplitude in terms of the sample.

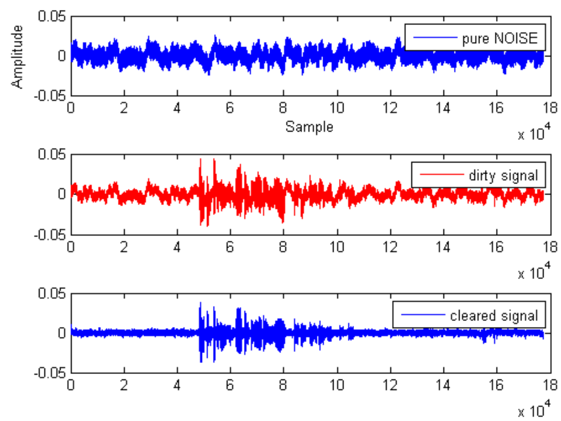

Figure 7. Noise frequency is 1000 Hz.
Amplitude in terms of the sample.

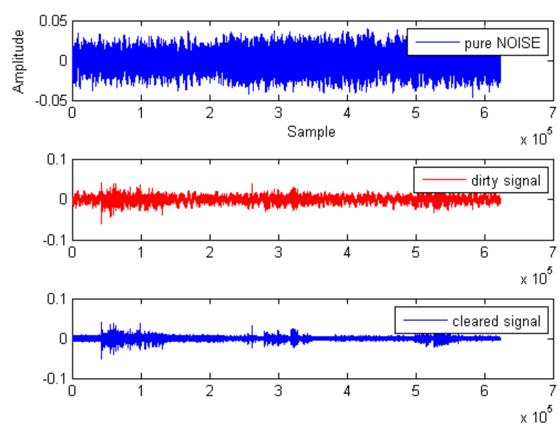

Figure 8. Varying noise.
Amplitude in terms of the sample.

Sine Noise Frequency (Hz)	SNR Improvement (dB)
50	42
100	42
500	43
1000	44
15, 100 and 710 consecutively	43

Table 1. SNR improvement.

Based on the SNR improvements in Table 1, time-domain graphs and also by hearing the output of the system, it is concluded that the proposed architecture introduces appropriate functionality and performance. The above experiments were conducted in real noisy situation and in two ways; using a P4-computer sound card, and using a FPGA starter kit individually.

4. FPGA development

The proposed system was successfully developed and synthesized on an XC3S1600E Spartan 3E FPGA starter kit. The adaptive FIR filter core employed was designed by Sharifi-Tehrani [1] in pure hardware using VHDL 93 hardware description language; the I/O controller unit of the system (ADC/DAC, audio, etc.) was designed using the XilKernel embedded operating system on an FPGA starter board. The resource utilization of the adaptive FIR filter, which is hardware efficient, is presented in Table 2. The entity of the adaptive FIR filter core employed is depicted in Figure 9 [1].

	Proposed Method	Method in Ref [2]
Slice Registers	1%	9%
4 input LUTs	1%	11%
Occupied Slices	1%	21%
Bonded IOBs	21%	21%
RAMB16s	5%	0
MULT18X18 SIOs	5%	5%
BUFGMUXs	4%	4%
Maximum Frequency (MHz)	186	50

Table 2. Resource utilization on XC3S1600E-5fg320

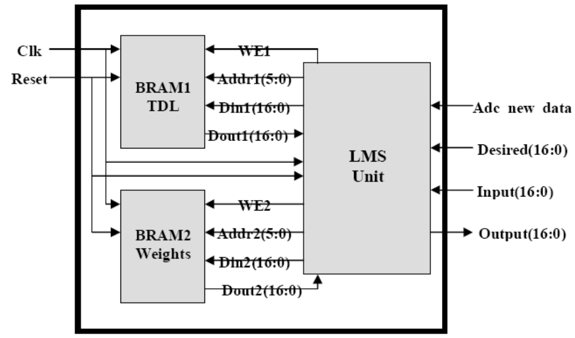

Figure 9. Entity of the adaptive FIR filter core employed.

5. Conclusion

An active sound-noise removal system based on a dual-microphone method to be used in a cockpit intercommunication system (ICS) was suggested. The system was experimented in a real noisy situation and its feasibility for real-time adaptive noise elimination on commercial and military platforms (such as C-130, IR-AN 140 and F5-F) was certified. This architecture can be synthesized by using a PC, DSP or FPGA for stand-alone applications. Low resource utilization and relatively low calculations are main benefits of the proposed architecture. Because the structure is adaptive, it is capable of estimating and converging to new acoustic path characteristics in a short period when the locations of sensors or propagation path are varied.

Acknowledgement

This project was supported by HESA Aviation Industries.

References

[1] O. Sharifi-Tehrani, "Novel Hardware-Efficient Design of LMS-based Adaptive FIR Filter Utilizing Finite State Machine and Block-RAM," *PRZEGLĄD ELEKTROTECHNICZNY (Electrical Review)*, Vol. 87, No. 7, pp. 240-244, August 2011.

[2] O. Sharifi-Tehrani and M. Ashourian, "An FPGA-Based Implementation of ADALINE Neural Network with Low Resource Utilization and Fast Convergence," *PRZEGLĄD ELEKTROTECHNICZNY (Electrical Review)*, Vol. 86, No. 12, pp. 288-292, December 2010.

[3] O. Sharifi-Tehrani et al., "An FPGA-Based Implementation of Fixed-Point Standard-LMS Algorithm with Low Resource Utilization and Fast Convergence," *Inter. Rev. on Comp. and Soft. (IReCOS)*, Vol. 5, No. 4, pp. 436-444, July 2010.

[4] A. DiStefano et al., "Efficient FPGA implementation of an adaptive noise canceller", *7th Inter. Workshop on Computer Architecture for Machine Perception*, pp. 87-89, Italy, 2005.

[5] M. Bahoura and H. Ezzaidi, "FPGA-implementation of a sequential adaptive noise canceller using Xilinx system generator", *Inter. Conf. on Microelectronics - ICM*, pp.213-216, Marrakech, 2009.

[6] V. Rodellar et al., "FPGA implementation of an adaptive noise canceller for robust speech enhancement interfaces", *4th Southern Conf. on Programmable Logic*, pp. 13-18, San Carlos de Bariloche, 2008.

[7] H. Zheng-wei and X. Zhi-yuan, "Modification of theoretical fixed-point LMS algorithm for implementation in hardware", *2nd Inter. Sym. on Electrical Commerce and Security - ISECS*, pp. 174-178, Vol. 2 China, IEEE Computer Society, 2009.

[8] A. A. Vega-Ramírez and J. L. Pérez-Silva, "Digital Implementation of a Logical Functor on a PLD," *Journal of Applied Research and Technology (JART)*, Vol. 9, No. 3, pp. 291-301, December 2011.

[9] F. M. Casco-Sánchez et al., "A New Variable Step-Size NLMS Algorithm and its Performance Evaluation in Echo Cancelling Applications," *Journal of Applied Research and Technology (JART)*, Vol. 9, No. 3, pp. 302-313, December 2011.

[10] A. Jalili et al., "Design and implementation of a fast active noise control system on FPGA", *Mediterranean Conf. on Control and Automation*, pp. 1-4, Athens, 2007.

[11] T. Adali and S. Haykin, "Adaptive Signal Processing-Next Generation Solutions", New York, Wiley-IEEE Publication, 2010.

[12] P. P. Chu, "FPGA Prototyping by VHDL Examples", New York, Wiley Publication, 2008.