ISSS2002/SA-434

Simulation Studies for Multichannel Active Vibration Control

Shashikala Prakash, R.Balasubramaniam
K.K. Praseetha

Structures Division
National Aerospace Laboratories,
PB 1779, Kodihalli, Bangalore 560017 India
E mail: shaship@css.cmmacs.ernet.in

ABSTRACT

Traditional approach to vibration control uses passive techniques, which are relatively large, costly and ineffective at low frequencies. Active Vibration Control (AVC) is used to overcome these problems & in AVC additional sources (secondary) are used to cancel vibration from primary source based on the principle of superposition theorem Since the characteristics of the vibration source and environment are time varying, the AVC system must be adaptive. Adaptive systems have the ability to track time varying disturbances and provide optimal control over a much broader range of conditions than conventional fixed control systems.

In multi channel AVC vibration fields in large dimensions are controlled & is more complicated. Therefore to actively control low frequency vibrations on large structures, multi channel AVC requires a control system that uses multiple secondary sources to control the vibration field simultaneously at multiple error sensor locations. The error criterion that can be directly measured is the sum of squares of outputs of number of sensors. The adaptive algorithm is designed to minimize this & the algorithm implemented is the "**Multiple error LMS algorithm**[1,2]". The best known applications of multiple channel FXLMS algorithm is in real time AVC and system identification. More wider applications are in the control of propeller induced noise in flight cabin interiors. In the present paper the results of simulation studies carried out in MATLAB as well as on TMS320C32 DSP processor will be brought out for a two-channel case.

Key words: Active Vibration Control, Adaptive filters, Smart Actuators, Smart Sensors, Digital Signal Processing, Least Mean Square algorithm, MATLAB, Simulation, Code compose studio.

ACRONYMS

P	Physical plant	IIR	Infinite Impulse Response
S	Secondary actuator	LMS	Least Mean Square
W	Adaptive FIR filter	PZT	Lead Zerconate Titanate
AVC	Active Vibration Control	d	Desired signal
CFRP	Carbon Fiber Reinforced Plastic	e	Error signal
FIR	Finite Impulse response	x	Primary noise/Input
FXLMS	Filtered –X LMS	y	Control filter output

y'   Secondary actuator output

INTRODUCTION

In many applications it is desirable to cancel / suppress vibration at multiple location i.e.. The simultaneous control of several rigid body modes are required in a mechanical structure Active Vibration Control is performed by introducing secondary forces to cancel the effects of undesired or primary vibration sources. The secondary forces are driven such that the super position of the primary and secondary fields destructively interfere with each other. Thus in AVC, vibration in the form of rigid body modes are suppressed or cancelled by the super position of areas in a mechanical structure.

   In this paper the simulation studies carried out in MATLAB for a single input multiple output (SIMO) control of a structure is brought out. The block diagram of the two channel AVC System is shown in Figure. 1. The algorithm used is the feed forward filtered – x LMS algorithm[3,4] to cancel broadband disturbances. It consists of a matrix of finite impulse response (FIR) control filters followed by a recursive section of FIR filters connecting each out put to each input.
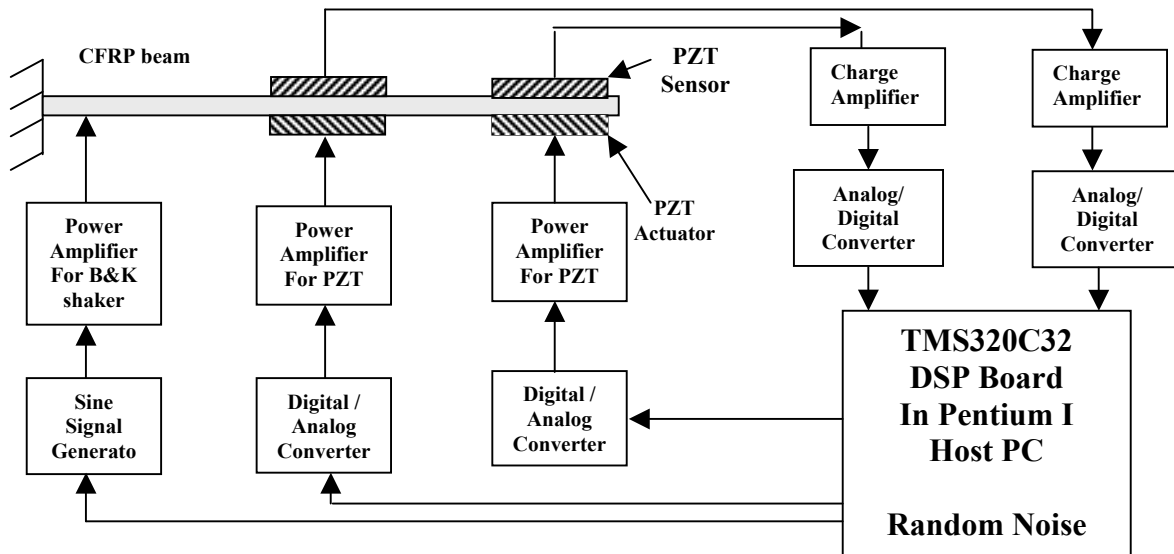

Figure 1: 1x2x2 Multicahannel Active Vibration Control system setup.

   The algorithm performs system identification where S_{lm} is defined from the input of the mth secondary source to the output of lth error sensors. This is required before the adaptation of filter parameters can begin. In this paper the results from 1 X 2 X 2 channel   is presented (one input channel, 2 error sensor and 2 control actuators). Also DSP implementation of the same has been completed successfully on a TMS320C32 DSP board. The real time testing on actual structure has to be carried out.

2. MULTIPLE ERROR LMS ALGORITHM

2.1 Derivation

Let the sampled output of the lth error sensor be $e_l(n)$, which is equal to the sum of the "desired" signal from the sensor, $d_l(n)$, due to the primary source operating alone, and an output due to each of the actuators. Let the sampled input to the mth actuator be obtained by filtering the reference signal x (n) using an adaptive FIR

2

controller whose i^{th} coefficient at the nth sample is w_{mi} (n). Let the transfer function between this input and the output of the lth sensor be modeled as the Jth – order FIR filter, whose jth coefficient is c_{lmj}, so that

$$e_l(n) = d_l(n) + \sum_{m=1}^{M} \sum_{j=0}^{J-1} c_{lmj} s_m (n-j) \tag{1}$$

where

$$S_m(n) = \sum_{i=0}^{I-1} w_{mi} x(n-i) \tag{2}$$

It is assumed that there are L sensors and M actuators, and that $L \geq M$. Let the total error J be defined as

$$J = E\left\{ \sum_{l=1}^{L} e_i^2(n) \right\} \tag{3}$$

Where E { . } denotes an expectation value. If the reference signal x (n) is at least partly correlated with each d_l(n), it is possible to reduce the value of J due to the primary source alone, by driving the secondary sources with a filtered version of the reference signal, as indicated in eqn(1).

It is clear that the total error will be a quadratic function of each of these filter coefficients. The optimum set of filter coefficients required to minimize J may thus be evaluated adaptively using gradient descent methods. The differential of the total error with respect to one coefficient is

$$\frac{\partial J}{\partial w_{mi}} = 2E\left\{ \sum_{i=1}^{L} e_l(n) \frac{\partial e_l(n)}{\partial w_{mi}} \right\} \tag{4}$$

Assuming for the moment that each w_{mi} is time invariant differentiating (1) with respect to one of these coefficients gives

$$\frac{\partial e_l(n)}{\partial w_{mi}} = \sum_{j=0}^{J-1} c_{lmj} x(n-i-j) \tag{5}$$

This sequence is the same as the one which would be obtained at the lth sensor if the reference signal, delayed by i samples, were applied to the mth actuator. Let this be equal to r_{lm} (n – i), a filtered reference.

If each coefficient is now adjusted at every sample time by an amount proportional to the negative instantaneous value of the gradient, a modified form of the well-known LMS algorithm is produced.

$$w_{mi}(n+1) = w_{mi}(n) - \mu \sum_{i=1}^{L} e_l(n) r_{lm}(n-i) \tag{6}$$

Where μ is the convergence coefficient. This is a form of "stochastic gradient" adaptive algorithm.

For a single input, single output system (L = M = 1), this corresponds exactly to the "filtered x LMS" algorithm discussed by Widrow and Stearns. The assumption of time invariance in the filter coefficients is equivalent, in practice, to assuming that the filter coefficients w_{mi} change only slowly compared to the timescale of the response of the system to be controlled. This time scale is defined by the values of the coefficients c_{lmj}.

2.2 Time Domain analysis

In order to analytically demonstrate the shape of the error surface, and so determine the optimum, wiener set of filter coefficients[5], it is convenient to consider the case in which the filter coefficients are exactly time invariant. eqn(1) may be written

$$e_l(n) = d_l(n) + \sum_{m=1}^{M} \sum_{i=0}^{I-1} w_{mi} \cdot \sum_{j=0}^{J-1} c_{lmi} x(n-i-j) \tag{7}$$

$$= d_l(n) + \sum_{m=1}^{M} \sum_{i=0}^{I-1} w_{mi} \cdot r_{lm}(n-i)$$

Where the filtered reference signal $r_{lm}(n)$ is defined as above. This equation may be written as

$$e_l(n) = d_l(n) + r_l^T w \tag{8}$$

$$r_l^T = \begin{bmatrix} r_{l1}(n), r_{l1}(n-1), \text{K } r_{l1}(n-I+1) \\ r_{l2}(n), r_{l2}(n-1), \text{K } r_{l2}(n-I+1) \\ r_{l3}(n), r_{l3}(n-1), \text{K } r_{l3}(n-I+1) \\ r_{lM}(n), r_{lM}(n-1), \text{K } r_{lM}(n-I+1) \end{bmatrix} \tag{9}$$

$$w^T = \begin{bmatrix} w_{10}, w_{11}, \text{K K } w_{1I-1} \\ w_{20}, w_{21}, \text{K K } w_{2I-1} \\ M \\ w_{M0}, w_{M1}, \text{K K } w_{MI-1} \end{bmatrix} \tag{10}$$

$$e^T = [e_1(n), e_2(n), \ldots e_L(n)] \tag{11}$$

$$d^T = [d_1(n), d_2(n), \ldots d_L(n)] \tag{12}$$

$$e = d + Rw, \quad R^T = [r_1, r_2, r_3, \ldots r_L],$$

The error criterion can now be written as

$$J = E\left\{ \sum_{l=1}^{L} e_i^2(n) \right\} = E\{ e^T e \} = E\{ d^T d \} + 2w^T E\{R^T d\} + w^T E\{R^T R\} w \tag{13}$$

4

Thus the error surface is quadratic in nature the surface has a unique global minimum. This will be known by examining the positive definiteness of $E\{R^TR\}$. By setting the differential of this expression with respect to w to zero, the optimum Wiener set of coefficients may be obtained as

$$w_{opt} \; = \; -[E\{R^TR\}]^{-1} \quad E\{R^{T\}}$$ (14)

This set of filter coefficients gives a minimum error criterion equal to

$$J_{min} = J_o \; - E\{d^TR\}[E\{R^TR\}]^{-1} \quad E\{R^Td\}$$ (15)

Where $J_o = E\{d^Td\}$ is the value of the error criterion with no control applied.

3. ALGORITHM SUMMARY

3.1. Adaptive filtering: (Figure. 2)

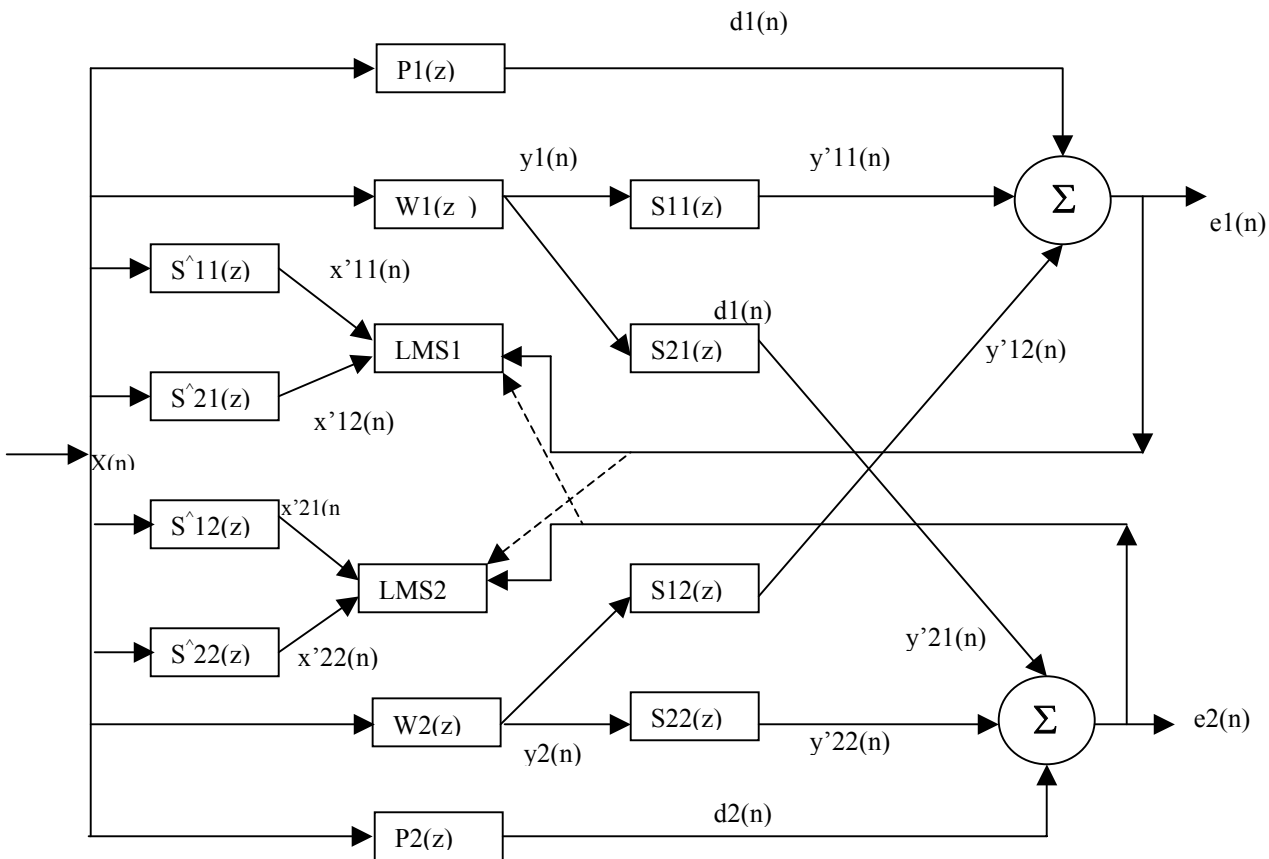$$y_1(n) = w_1^T(n)x(n) \; \& \quad y_2(n) = w_2^T(n)x(n),$$

Figure.2 Block diagram of a 1x2x2 Multiple Channel AVC System

5

3.2. Multiple-channel pre filtering: $\quad \mathbf{x'}_{lm}(n) \equiv \mathbf{s'}_{lm}(n) * \mathbf{x(n)}$

$$= \sum_{i=0}^{I-1} s_{lm,j}(n)\mathbf{x(n-i)} \quad \text{l (sensors) =1,2 \& m (Actuators) =1,2 and}$$

J is the order of the FIR filter $s'_{lm}(z)$.

3. Weight vector updates:

$$w_1(n+1) = w_1(n) + \mu[\mathbf{x'}_{11}(n)e_1(n) + \mathbf{x'}_{12}(n)e_2(n)]$$
$$w_2(n+1) = w_2(n) + \mu[\mathbf{x'}_{21}(n)e_1(n) + \mathbf{x'}_{22}(n)e_2(n)]$$

4. IMPLEMENTATION ASPECTS

4.1 MATLAB[6] Implementation: The simulation of a single input 2 error sensors, 2 control actuator case (1 X 2 X 2) case was attempted in MATLAB on a Pentium II 350 MH$_z$ PC. This needs 2 adaptive filters each having only 2 weights. Although the algorithm is valid for general broadband signals, considerable simplifications occur in the implementation if the reference signal is sinusoidal. The desired signals were generated by passing the input signal x (n) through a 2 point FIR filter. In the multi channel case there appears to be no more interference between the convergence of the coefficients of the multiple filters than between the coefficients of a single filter, even though the update terms for each filter are coupled. Once we got good convergence for the case, two other cases were tried. In one case we used random signal with a sinusoidal component as the reference signal and tried to filter out all the random components after convergence i.e., the desired signal was a sinusoid (the same that was added to the random reference signal). In the second case two sinusoids were added and used as reference signal and good convergence was obtained. These are clearly shown in figures 3, 4 & 5 respectively. We have overlapped the filter output onto the desired signal spectrum and zoomed one portion to show how good the match is.

As we go for random and complex reference signal inputs, we see that we need more number of tap weights to achieve good convergence in a reasonable time. If we use two weight vectors, then the number of samples required will be enormous and also the time taken will be too high.

4.2 DSPImplementation: The multiple error LMS algorithm has been programmed in assembly language on a Texas Instruments TMS320C32[7] DSP board mounted in a Pentium I 100 MHZ PC in assembly language. Circular buffers and the parallel instructions have been used, which considerably speeds up the computations. Initially trials were made on sinusoidal reference signal accessed as a look up table. Once confidence was gained signals were acquired from a signal generator and simulation studies were carried out. For the case of random signal added to sinusoid & for system identification, the random signal was generated using the DSP processor using the pseudo random noise generation algorithm and this was added to the acquired sinusoid. Good results have been obtained.

The robustness of the algorithm as proved by the random and complex input cases show that the convergence is largely unaffected by the introduction of considerable uncorrelated observation noise Also the programs have been put on the Texas Instruments Code Composer Studio[8] and have been meticulously tested.

We need to conduct experiments on an actual structure and we need to try to suppress the vibration levels at two modes using PZT sensors and smart PZT actuators.
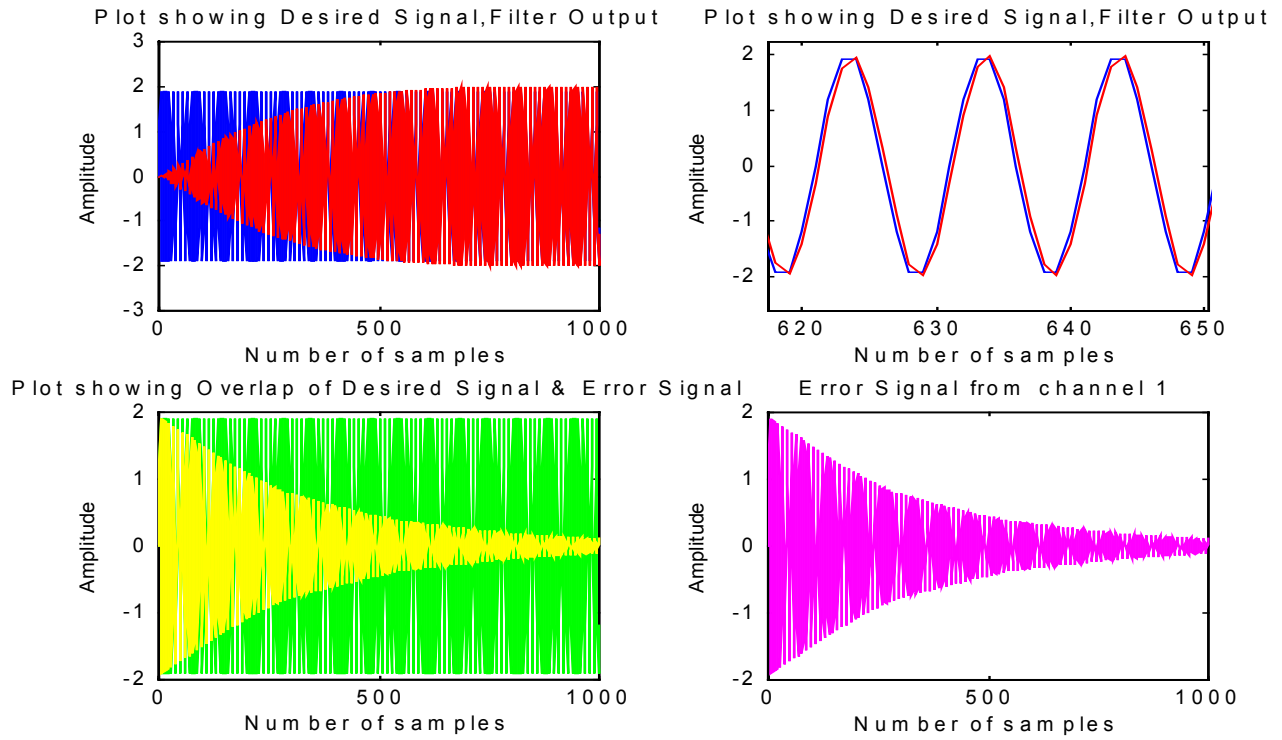
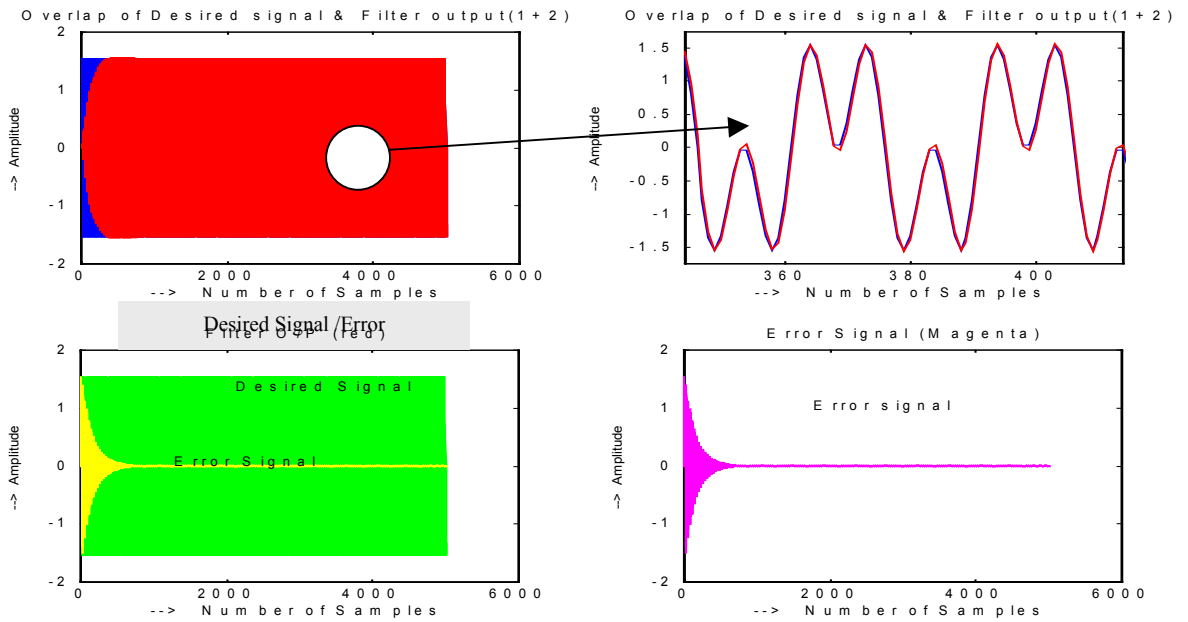**Figure. 3** Matlab Simulation results – Sine Input/Sine Desired

Figure 4. MATLAB Simulation outputs– Complex I/P Complex Desire

CONCLUSIONS

This paper describes the implementation of Multi Channel LMS algorithm for the Active Vibration Control (AVC). This implementation has been carried out by using an adaptive filter that is a digital filter composed of a tapped delay line and adjustable weights, having a finite impulse response, which is controlled by the LMS algorithm. For stationary stochastic inputs the mean square error, which is the difference between the filter outputs and an externally supplied input called desired response is a quadratic function of the weights.

This multi channel LMS control algorithm has been implemented on a Pentium II 350MHz PC using the MATLAB utility. The program has been validated for cases like sine I/P sine Desired, random input/sine desired and complex input/complex desired signals. This multi channel LMS algorithm has been implemented for the case of two adaptive filters, two error sensors and two control actuators.
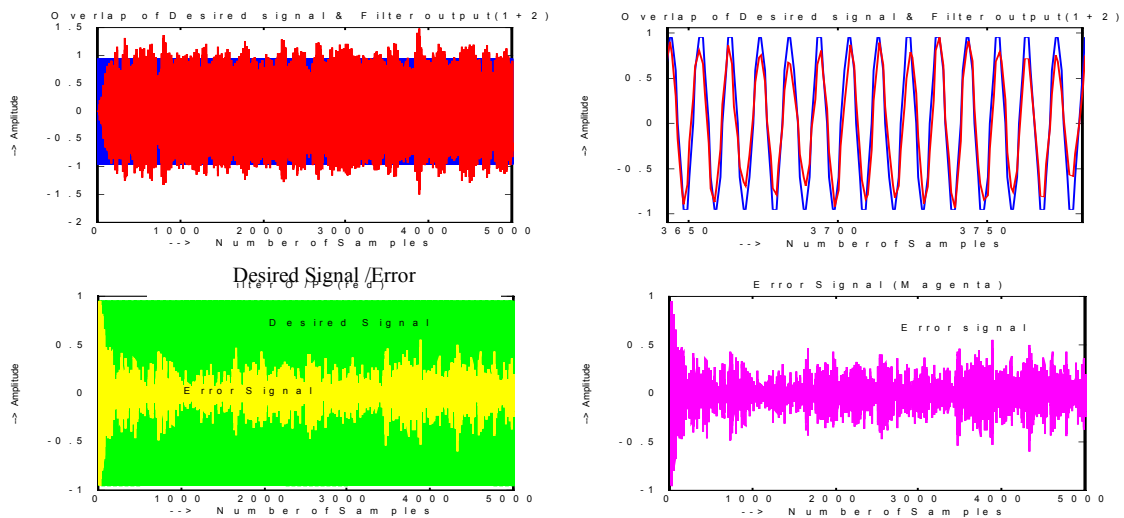

Figure 5. MATLAB Simulation outputs – RandomI/P & SineDesired

ACKNOWLEDGMENTS

REFERENCES

1.   S. J. Elliott, P. A. Nelson "*A Multiple Error LMS Algorithm & its application to the Active Control of Sound & Vibration", IEEE – ASSP 35, No.10. (Oct 1987).*

2.   S. J. Elliott, P. A. Nelson "*Algorithm For Multi Channel LMS Adaptive Filtering", Electronics Letters 1985*", 10th October 1985, Volume-21, No. 21, Pp 979-981.

3.   Bernard Widrow, Samuel D. Stearns-"*Adaptive Signal Processing*", Prentice Hall, Inc., 1985.

4.   Sen M. Kuo, Dennis R. Morgan, *Active Noise Control Systems,* Algorithms And DSP Implementation, John Wiley And Sons-Inc, 1996.

5.   John R. Treichler, C.Richard Johnson, Michael G. Larimore-"Theory And Design Of Adaptive Filters", John Wiley & Sons 1987.

6.   Matlab Reference Manual.

7.   TMS320C32 users guide

8.   Code Composer Studio users Manual