Analysis of Adaptive Direct sequence Spread Spectrum using Least mean squares Algorithm

Mainak Chakraborty*, Moloy Kumar Chowdhury**

*(Department of Electronics & Communication Engg, Heritage Institute of Technology, Kolkata)

** (Department of Electronics & Communication Engg, Heritage Institute of Technology, Kolkata)

ABSTRACT

Spread spectrum forms an important aspect of digital communication technology where the message signal is transmitted over a channel having bandwidth much greater than required. Direct Sequence Spread Spectrum (DSSS) belongs to the category of spread spectrum methods and forms an important part in mobile communications. The objective of this paper is to investigate the FIR optimal, Wiener and least mean square (LMS) algorithms in the design of traversal tap delay line filters for the purpose of compensating the effect of the quasi-static communication channel in mobile communication using DSSS. The designed equalizers remove the distortion caused by the channel from the transmitted signal with the help of an adaptive, recursive Weiner filtering technique using the concept of Weiner-Hopf equation using LMS algorithm and to aid in the equalization process, a training sequence is first transmitted to adapt the filter coefficients. The performance metrics are studied for DSSS system with and without adaptive equalization. Further, the convergence of LMS algorithm is also verified as well as the impact of different step-sizes on the speed of the conversion and the accuracy of the overall algorithm. Exhaustive Monte Carlo simulative analysis have been performed to investigate the scheme under varying distortion levels and Signal to noise ratio (SNR) values via impulse response, frequency response and uncoded Bit Error Rate (BER) simulations.

Keywords - *Direct Sequence Spread Spectrum, Least Mean Square, Weiner-Hopf equation, Channel Equalization, Adaptive filter, Transversal tap delay line filters, Bit error rate.*

I. INTRODUCTION

In digital communication, spread spectrum technique is used to increase the transmitted signal bandwidth to a higher value than required for transmitting the information digits. Spread spectrum technology was developed for military and intelligence purpose, and radar engineers employed the method of spectrum spreading for jamming. Most recent spread spectrum techniques (2.4GHz) are associated with IEEE 802.11 standard. There are a few number of spread spectrum methods among which two are approved by FCC, namely Direct sequence spread spectrum and Frequency Hopping spread spectrum. IEEE 802.11 direct sequence spread spectrum technology encompasses differential binary phase shift keying for 1Mbps and differential quadrature phase shift keying at 2Mbps. Both these digital modulation techniques used are based on phase shift keying modulation.

Sometimes signals are at a low probability to intercept, and as a result they are difficult to be detected or to be demodulated after detection. To ensure the security, a signal is needed to be developed which is very difficult to interfere with or to “jam”. Employment of spread spectrum techniques eliminates the problems and also fulfils the question of security[1].

Direct sequence spread spectrum techniques offers several advantages like easy implementation and high data rate. At present almost all Wireless LANs use this technology. Here both the transmitter and receiver are set with frequency at 22 MHz. In DSSS, chipping code is used to combine with data signal. The RF carrier is then modulated by the combined signal so that the transmitted signal is spread over a wide bandwidth. Another name of chipping code is processing gain which is implied to protect high signal from interference. Sum of chips in the code will determine how big the spreading occurs, and sum of chip per bit and code rate will determine the data rate.[2]

We may consider frequency translation of a message signal of B Hz by DSBSC modulation technique. The modulated signal (of bandwidth 2B Hz) .It would typically override the noise occupying the same part of the spectrum. So it would be easy to detect by spectrum analyzer giving very high probability of intercept. Moreover, to achieve synchronous demodulation, a carrier synchronized with transmitter is required at the receiver [19].The recovered signal to noise ratio comes as 3 dB better than measured one and it occurs due to coherent addition from each of sidebands where noise is not added. This 3 dB improvement is called as processing gain. In practice, transmission bandwidth and message bandwidth are in the ratio of 2:1 and thousand of different carriers are implemented in order to generate thousands of DSBSC signals from same message. As the carriers are spread over a large bandwidth (much greater than 2B Hz) all of these DSBSC signals are spread over the same bandwidth. Here the power of an individual DSBSC in spread spectrum phenomenon is thousand times less as compared to the single DSBSC case. It would be hard to find out by spectrum analyzer, as over the entire bandwidth occupied by one of these DSBSC signals, it would be in literally 'buried in the noise' state. Instead of the total transmitted power being concentrated in a band of width 2B Hz, the multiple carriers have spread it over a very wide bandwidth. We observe that the signal-to-noise ratio for each DSBSC is very low (well below 0 dB). At the receiver in order. to recover the message from the transmitted spread spectrum signal , thousands of local carriers are needed, at the same frequency and of the same relative phase, as all those at the transmitter. All these carriers come from a pseudo random binary sequence (PRBS) generator. [9]

With the help of a stable clock, and a long sequence, it can be proved that the spectrum of a pseudo random binary sequence generator is a good source of these carriers. Similarly, a second PRBS generator, at the same rate, and correctly aligned, is enough to regenerate all the local carriers at the receiver demodulator for perfect synchronization. Alternatively the PRBS signal is termed as PN – pseudo noise - signal, since its spectrum approaches that of random noise [7].

In other words, correct sequence at the receiver means that the message received from each of the thousands of minute DSBSC signals are combined in phase, coherently and are added up to produce a finite message output. Otherwise they add with random phases, which in turn results weak power level noise-like output [14].

II. ADAPTIVE FILTERING

In direct sequence code division multiple access(DS/CDMA) system, to recover the transmitted information, the received signal should first despread using a locally generated pseudo noise (PN) code sequence. The code acquisition performs initial code timing alignment between the locally generated and received signals. Code acquisition method is usually conducted using a correlator to serially search the code phase, which is related to the channel propagation delay. This approach performs well for the adaptive filtering approaches have high acquisition-based capacity and drawback of these schemes is high computational complexity [8]. Due to down-sampling operations, the computational complexity of the adaptive filters can be effectively reduced. Code synchronization is a very essential and important part of any spread spectrum (SS) system in order to remove the spreading effect induced by the transmitter and to exploit the processing gain of the spread signal [12].The receiver must be able to estimate the delay offset between the spreading code in the received signal and the locally generated replica of the code before data demodulation is started. Code synchronization is usually developed over two steps, namely acquisition and tracking [9].In this application of multirate processing, not only the computational complexity, but also the mean acquisition time can be effectively reduced. Here the basic idea of the scheme used in multirate adaptive filtering for DS/CDMA code acquisition is discussed [10].

Figure 1 shows a generic adaptive signal processing system. The system consists of three parts processor, adaptation algorithm, performance function. The processor is a part of the system that is actual responsible for processing of input signal x and generating output signal y. The processor is however to be a FIR digital filter. The performance function however takes the signals x and y as inputs along with other data d that may affect the performance of the system from X to y.The performance function is the quality measure of the adaptive system. To the objective function and in control theory it corresponds to the cost function. The output ε is the performance function of the quality signal illustrating the processor at its present state and indicating whether it is performing well taking into account the input signal, output signal and other relevant parameters[18]. The quality signal ε is finally fed to the adaptation algorithm. The task of adaptation algorithm is to change of the parameters of the processor in such a way performance is maximized. In the example of adaptive filters this would require change in tap weights. Close loop adaptation has advantage of being usable in many situations where no analytic synthesis procedure is either exist or is known and/or when the input signal characteristics varies considerably. Further in cases where physical system components values are variable or inaccurately known closed loop adaptation will be the best choice of parameters. In the event of partial system failure the adaptation mechanism may even be able to readjust the processor [21].

In a way that the system will still achieve an acceptable performance. System reliability can often be improved using performance feedback. There are however some inherent problems as well.The processor model and the way the adjustable parameters of the processor model is chosen affects the possibilities of building a good adaptive system[6]. We must be sure that the desired response of the processor can really be achieved by adjusting the parameters within the allowed range.Using too complicated a processor model would on the other handmake analytic analysis of the system cumbersome or even impossible.The performance function also has to be chosen carefully [17]. If the performance function does not have a unique extremum, the behaviour of the adaptation process may be uncertain. Further, in many cases it is desirable(but not necessary)that the performance function is differentiable. Choosing "other data" carefully may also afffect the usefulness of the performance function of the system [20].Finally the adaptation algorithm must be able to adjust the parametersof the processorin a way to improve the performance as fast as possible and to eventually convergeto the optimum solution(in the sense of the performance function).Too fast an adatation algorithm may, however, resultsin instability or undesiredoscillatory behaviour. These problems are well known from optimization and control theory[5].

The processor and the performance function:- In this section a common processor model, th eadaptive linear combiner and a common performance function, the mean square error wull be presented.Throughout the text,vectornotation will be used where all the vectors are assumed to be column vectors. And are usually indicated as a transpose of a row vector unless stated otherwise [13].

The adptive linear combiner:- One of the most common processor model is the so called adaptive linear combiner shown in figure 3.2. In the most general form the combiner is assumed to have *L+1* inputs denoted

$$X_k = [x_{0k} x_{1k} \ldots\ldots x_{Lk}]^T \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(1)$$

Where the subscript *k* is used as the time index . Similarly the gainsor weights of the different branches are denoted

$$W_k = [w_{0k} w_{1k} \ldots\ldots w_{Lk}]^T \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(2)$$

Since we are now discussing with an adaptive system the weights will also vary in time and hence have a time subscript *k*. The output *y* of the adaptive multiple input linear combiner can be expressed as

$$y_k = \sum_{l=0}^{L} w_{lk} x_{lk} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(3)$$

As can be seen from equation (3.3) this is nothing but a dot product between the input vector and the weight vector ,hence

$$y_k = X_k^T W_k = W_K^T X_K \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(4)$$

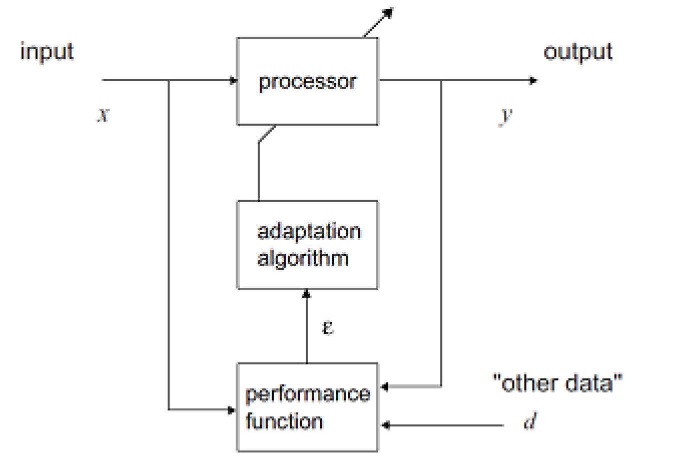

Figure 1 Closed Loop Generic Adaptive System

III. LMS ALGORITHM

The After Least mean squares (LMS) algorithms are a class of adaptive filter used to mimic a desired filter by finding the filter coefficients that relate to producing the least mean squares of the error signal (difference between the desired and the actual signal). It is a stochastic gradient descent method in that the filter is only adapted based on the error at the current time [4]. The LMS algorithm uses a special estimate of the gradient that is valid for the adaptive linear combiner. Thus, the LMS algorithm is more restricted in its use than the other methods. On the other hand, LMS algorithm is important because of its ease of computation and because of it does not require the repetitive gradient estimation[16]. The LMS algorithm is quite often the best choice for many different adaptive signal processing applications in processing domain. The LMS algorithm is a widely used algorithm for adaptive filtering. The algorithm is described by the following equations [11].

$$y(n) = \sum_{i=0}^{M-1} W_i(n) * x(n-i) \ldots\ldots\ldots\ldots\ldots(5)$$

$$e(n) = d(n) - y(n) \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(6)$$

$$w_i(n+1) = w_i(n) + 2ue(n) * x(n-i) \ldots.(7)$$

In these equations, the tap inputs x(n),x(n-1),……,x(n-M+1) form the elements of the reference signal x(n), where M-1 is the number of delay elements. d(n) denotes the primary input signal, e(n) denotes the error signal and constitutes the overall system output. $w_i(n)$ denotes the tap weight at the nth iteration. In equation (3), the tap weights update in accordance with the estimation error. And the scaling factor u is the step-size parameter u controls the stability and convergence speed of the LMS algorithm. The LMS algorithm is convergent in the mean square if and only if u satisfies the condition: 0 < u < 2 / tap-input power M where tap input power is: $\sum_{k=0}^{M-1} E\{|u(n-k)^2|\}$. Here $\lambda_{\max}$ is the largest Eigen value of the correlation matrix of the input data. The stability would be confirmed if the following relation is satisfied:

$$0 < \mu < \frac{2}{inputsignalpower} \quad \ldots\ldots\ldots\ldots\ldots(8)$$

If step size increases, it would increase adaptation rate as well as residual mean squared errorThe LMS algorithm is most commonly used adaptive algorithm because of its simplicity and a reasonable performance. Since it is an iterative algorithm it can be used in a highly time-varying signal environment. It has a stable and robust performance against different signal conditions. However it may not have a really fast convergence speed compared other complicated algorithms like the Recursive Least Square (RLS). It converges with slow speeds when the environment yields a correlation matrix R possessing a large Eigen spread. Usually traffic conditions are not static, the user and interferer locations and the signal environment are varying with time, in which case the weights will not have enough time to

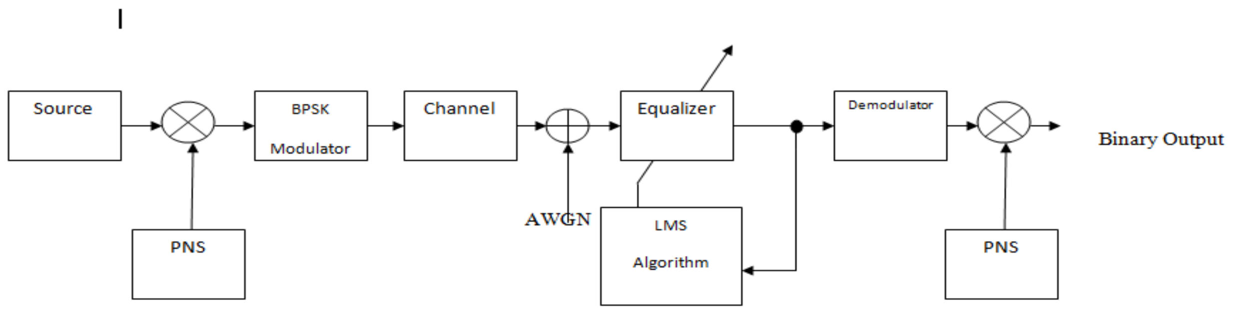

Figure 2 Block Diagram of DSSS-LMS

converge when adapted at an identical rate.The,µ is the step-size needs to be varied in accordance with the varying traffic conditions. There are several variants of the LMS algorithm that deal with the shortcomings of its basic form. The Normalized LMS (NLMS) introduces a variable adaptation rate [15]. It improves the convergence speed in a non- static environment. In another version, the Newton LMS, the weight update equation includes whitening in order to achieve a single mode of convergence [3]. For long adaptation processes the Block LMS is used to make the LMS faster. In block LMS, the input signal is divided into blocks and weights are updated block wise. A simple version of LMS is called the Sign LMS. It uses the sign of the error to update the weights. Also, LMS is not a blind algorithm i.e. it requires a priori information for the reference signal [7].

In the block diagram (Fig. 2), the source contains digital information. The PNS is pseudo random sequence which is itself semi random in nature. It is multiplied and applied to BPSK modulation input. The modulated output is given to a time varying channel. The channel is quasi static in nature. At the receiver adaptive equalizer is adjusted depending on coefficients that are determined by LMS algorithm. The equalized output is applied to the BPSK demodulator where it is further multiplied by PNS to obtain binary output which is an estimate of the original source signal.

Table:- Various Parameters and Values

Parameter	Values
Total Iteration	**700**
Step Size	**0.125**
Taps	**7**
SNR	**30dB**

IV. RESULTS OF SIMULATION

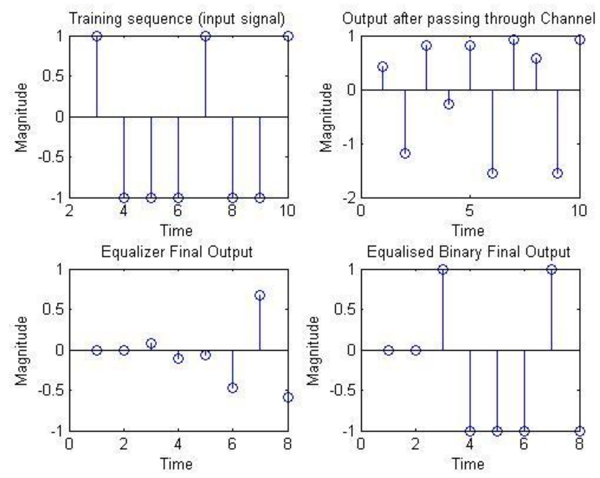

Fig. 3 Training Sequence and Equalized Output

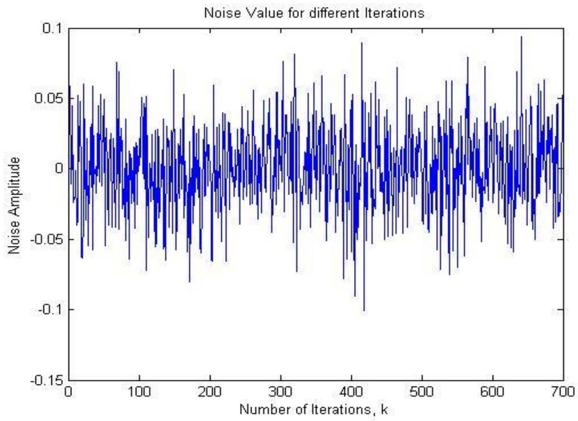

Fig.4 Noise Value for different Iteration

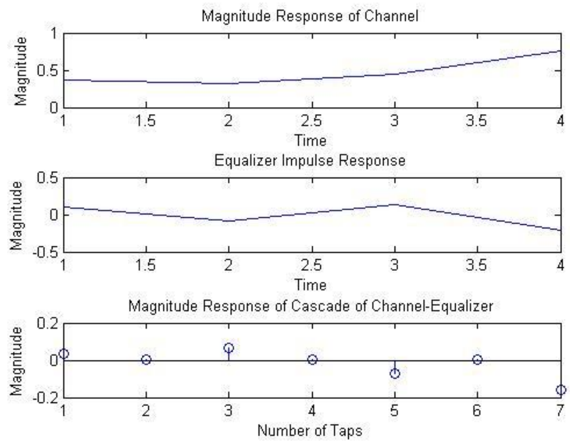

Fig.7 Channel and Equalizer Impulse Response

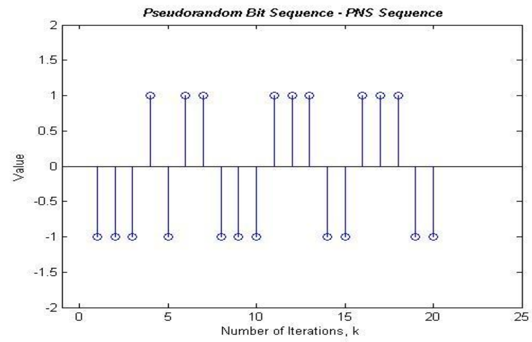

Fig.5 PNS Bit Sequence

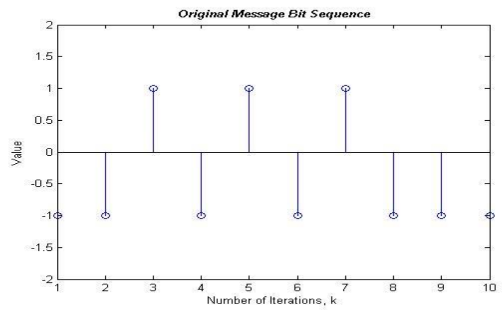

Fig.6 Original Message Bit Sequence

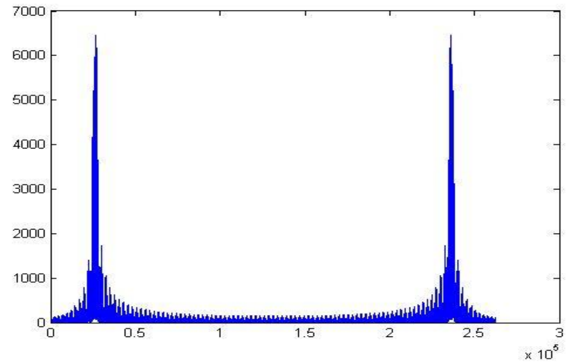

Fig.8 Dsss Frequency Spectrum

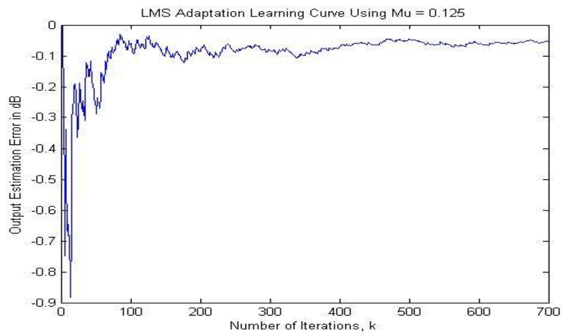

Fig.9 LMS Adaptation Learning Curve

Fig 3 shows four discrete time signals The first sequence is training sequence for equalizer training. The training sequence is convolved to the channel. This is shwon in output after channel. Final output of equalizer shows non zero values at only two time indices. The next picture shows equalized binary final output and equals the training sequence. In Fig 4, we observe the noise amplitude for different iterations. The total no. of iteration are set at k=700. In Fig. 5, PNS sequence is shown which is pseudorandom bit sequence only for 25 iteration. In Fig. 6, original message bit sequence generated by the source for 10 iteration only. In Fig. 7, the imulse response of channel and equalizer is observed. Here we see that equalizer impulse response is inverse of channel response and cascade combination of channel and equalizer againstno. Of taps. In Fig.8 it exhibits the direct sequence spread spectrum frequencey spectrum. In Fig. 9, we observe LMS adaptation curve for step size0.125and output estimation error (in dB) is plotted against total no. of iteration. Here we see that untill 100 iteration, LMS curve exhibits fluctuations and after it becomes constant at -0.1dB.

.

VI. CONCLUSION

Here we implemented the adaptive direct sequence spread spectrum communication system based on LMS algorithm. Before the actual data is transmitted training sequence is initially sent to adjust the equalizer to nullify the channel distortion. Then the data is transmitted at the end of each packet transmission and the channel profile changes. Thus the channel is modeled as quasi static in nature. The output sequence at DSSS demodulator equals the original digital information sent by source. We also observed that LMS adaptation learning curve gradually approaches a constant value. Based on LMS algorithm, the equalizer impulse response is observed to be inverse of channel response.

REFERENCES

[1] G. R. Ansari and B. Liu, "Multirate signal processing," in Sanjit. K. Mitra and James F. Kaiser (ed.), Handbook for Digital Signal Processin, 981-1084, John Wiley, 1993.

[2] M.G. Bellanger, G. Bonnerot, and M. Coudreuse, "Digital filtering by polyphase network: application to sample-rate alteration and filter banks," IEEE Transactions on Acoustics, Speech, and Signal Processing, vol. 24, pp. 109-114, April, 1976.

[3] M. Bellanger, Digital processing of signals: Theory and practice, New York: John Wiley & Sons, 1989.

[4] Crochiere, R.E., and Rabiner, L.R., "Interpolation and decimation of digital signals - A Tutorial Review," Proceedings of the IEEE, vol.78, pp. 56-93, March 1981.

[5] R.E., Crochiere and L.R. Rabiner, Multirate digital signal processing Englewood Clifs, Prentice-Hall, 1983.

[6] A.G. Dempster and M.D. Macleod, "General algorithms for reduced adder integer multiplier design," Electron. Letters, vol. 31, pp. 261-264, Oct. 1995.

[7] W. Drews and L. Gaszi, "A new desig method for polyphase filters using all-pass sections," IEEE Transactions on Circuits and Systems, vol. 33, pp. 346-348, March 1986.

[8] N. J. Fliege, Multirate digital signal processing, New York: John Wiley & Sons, 1994.

[9] F. J. Harris, Multirate Signal Processing, Upper Saddle River: Prentice Hall, 2004.

[10] T. Hentchel, Sample Rate Conversion in Software Configurable Radios,Morwood, MA: Artech House, 2002.

[11] A Antonieu, Digital Filter Analysis and Application,McGraw Hill New York

[12] J W Adams FIR Digital Filters , IEEE,1991

[13] J W Adams and A N Wilson,A new Approach to Digital FIRfilter,IEEE,1983.

[14] J Babts, Digital Signal Processing Applications, Prentice Hall 1995.

[15] S. K. Mitra, Digital Signal Processing, McGrawHill, 1999.

[16] Andrea Goldsmith, Wireless Communications. Cambridge University Press,2004.

[17] Bernard Sklar, Digital Communications. Pearson Education Press,2005

[18] Chanpreet Kaur et al. / International Journal of Engineering Science and Technology (IJEST) “High Speed and Cost Efficient Adaptive Equalizer for ISI Removal”Vol 3, No 1, Jan 2011.

[19] J.G. Proakis, “Adaptive equalization for tdma digital mobile radio,”

[20] IEEE Trans. Vehic. Technol., vol. 40, no. 2, pp. 333–341, 1991.

[21] B. Widrow and S. D. Stearns, Adaptive Signal Processing. Prentice Hall Signal Processing Series, 1985.