FPGA Implementation of NLMS Algorithm for Receiver in wireless communication system

K. R. Rekha *, **Dr B. S. Nagabushan** ** **And Dr K.R..Nataraj** ***

*E&C Dept, M. G. R. University, Chennai, India

** Sanlab Technologies, Bangalore, 560038, India

*** SJB Institute of Technology, Bangalore, 560060, India

Abstract

This paper proposes a verilog implementation of a normalised Least Mean Square (NLMS) adaptive algorithm. The envisaged application in the wireless communication identification system. The good convergence of NLMS algorithm has made us to choose it. It also has good stability. Adaptive filtering constitutes one of the core technologies in digital signal processing and finds numerous application areas in science as well as in industry. In this paper NLMS algorithm is used to reduce the error at the output of the receiver in wire less communication system. A verilog implementation is developed for a 5th order NLMS adaptive filter. As compared conventional LMS it has been proven that NLMS Algorithm has good behaviour. Model Sim simulations results altogether with plots obtained in Mat lab prove the same.

Keywords:

Adaptive filters , Digital signal processing,FPGA NLMS, LMS.

1. Introduction

Even though many interesting adaptive algorithms are present around us. The applications with
limited precision and processing power, the Normalized Least-Mean- Square (NLMS) algorithm [3] and some versions of it (e.g., frequency-domain or sub-band versions [1]) are usually used. The step-size parameter will control the algorithm, in terms of convergence rate, maladjustment, and stability. Within the stability conditions, the choice of this parameter reflects a trade-off between fast convergence rate and good tracking ability on the one hand and low maladjustment on the other hand. To meet these conflicting requirements, the step size needs to be controlled. This algorithm gives good performance.

Reducing an error in receiver system has been a central issue in wireless communication networks and teleconferencing etc. reducing an error in the procedure of specifying the receiver model in terms of the available experimental evidence, that is, a set of measurements of the input output desired response signals and an appropriately error that is optimized with respect to Receiver parameters. Adaptive identification refers to a particular procedure where we learn more about the model as each new pair of measurements is received and we update the knowledge to incorporate the newly received information.

2. Need for NLMS

In this paper, we will be describing the more stabled adaptive algorithm Normalized Least Mean Square (NLMS) algorithm. The paramount reasons for this decision are as follows:

1. The LMS adaptive FIR filter is the most popular adaptive estimation technique and is likely to remain so in the foreseeable future [9]. Despite speculations that the LMS algorithm is losing its established status as the workhorse for the design of linear adaptive systems, there are still numerous ongoing researches and state-of-the-art advances in this algorithm [7]. A forthcoming new book titled —Least-Mean-Square Adaptive Filters, edited by Bernard Widrow (originator of LMS) and Simon Haykin,is a good representative of the devoted interests in this adaptive algorithm shown by researchers around the world.

2. The LMS algorithm can be easily modified to a normalized step-size version known as the Normalized LMS (NLMS) algorithm. NLMS, not only provides a potentially faster adaptive algorithm, but also guarantees a more stable convergence in response to variations of input signal power.

2.1 Following this approach, the main objectives of this paper are:

1. Study of LMS /NLMS adaptive algorithms.

Manuscript received December 5, 2010
Manuscript revised December 20, 2010

2. Implement an NLMS-based adaptive algorithm for receiver system in wireless communication to show its good convergence rate and reduction in error.

3. Design of modified LMSalgorithm

In this section the derivation of modified LMS has been given and it is named as normalised least mean square algorithm (NLMS).
In many adaptive filter algorithms Normalized least mean square algorithm (NLMS) is also derived from conventional LMS algorithm. The objective of the alternative LMS-based algorithms is either to reduce computational complexity or convergence time. The normalized LMS, (NLMS),
algorithm utilizes a variable convergence factor that minimizes the instantaneous error. Such a convergence factor usually reduces the convergence time but increases the misadjustment. In order to improve the convergence rate the updating equation of the conventional LMS algorithm can be employed variable convergence factor $\tilde{\mu}$. it is derived as below.
μσ2x directly affects the convergence rate and stability of the LMS adaptive filter. As the name may imply, the NLMS algorithm is an effective approach to overcome this dependence, particularly when the variation of input signal power is large, by normalizing the update step-size with an estimate of the input signal variance, σ2x(n) [10]. In practice, the correction term applied to the estimated tap-weight vector w(n) at the n-th iteration is 'normalized' with respect to the squared Euclidean norm of the tap input x(n) at the (n-1)-th iteration [8],

$$w(n+1) = w(n) + \frac{\tilde{\mu}}{\|x(n)\|2} e(n)x(n) \qquad 3.1$$

Apparently, the convergence rate of the NLMS algorithm is directly proportional to the NLMS adaptation constant $\tilde{\mu}$, i.e. the NLMS algorithm is independent of the input signal power. Theoretically, by choosing $\tilde{\mu}$ so as to optimize the convergence rates of the algorithms, the NLMS algorithm converges more quickly than the LMS algorithm [10].
Indeed as reported in [11], by taking into account the variation of signal level at the filter input and selecting a normalized correction term, we get a stable as well as a potentially faster converging adaptation algorithm for both uncorrelated and correlated input signal. It has also been stated that the NLMS is convergent in the mean square if the adaptation constant û (note that it is no longer called the step size) satisfies the following condition [12]:

$$0 < \tilde{\mu} < 2 \qquad \ldots\ldots\ldots \qquad 3.2$$

Despite this particular edge that NLMS exhibits, it does have a slight problem of its own. Consider the case when the input vector x(n) is small. Instability may occur since we are trying to perform numerical division by a small value of the Euclidean Norm ||x(n)||2.
However, this can be easily overcome [8] by appending a positive constant to the denominator in W(n+1) = w(n) +μe(n)x(n) such that

$$W(n+1) = w(n) + \frac{\tilde{\mu}}{c+\|x(n)\|^2} e(n)x(n)$$

…….3.3

where $c + \|x(n)\|^2$ is the normalization factor. With this, we obtain a more robust and reliable implementation of the NLMS algorithm.
In summary, we can write the LMS algorithm for every search iteration, in the form of three operations:
Initial Condition: $0 < \tilde{\mu} \leq 2$
x(0) = w (0) = [0, ……, 0]^T ,
c = a small constant

1. Filter Output: y(n) = w(n) x' (n)
2. Error Estimation: e(n) = d(n) - y(n)
3. Tap-weight adaptation:

$$w(n+1) = w(n) + \frac{\tilde{\mu}}{c+\|x(n)\|^2} e(n)x(n)$$

4. DETAILED DESIGN AND IMPLEMENTATION

The below block diagram (figure 4.1)shows the inputs and outputs of the NLMS algorithm. It has four inputs and two outputs. Inputs are x_in (input data to adaptive filter),d_in (desired input), clk(clock) and adpt_enable (input bit used for ……). Outputs are error_out (difference between output of filter(y-out) and desired input (d_in)) and final out (…).
The inputs x_in and d_in and outputs error_out and final_out are 8bit data.clock and adpt_enable are single bit data.
The NLMS block consists of two shift registers, calculator, adder and a multiplexer. The inner structure of the NLMS is as shown in figure 4.2 below

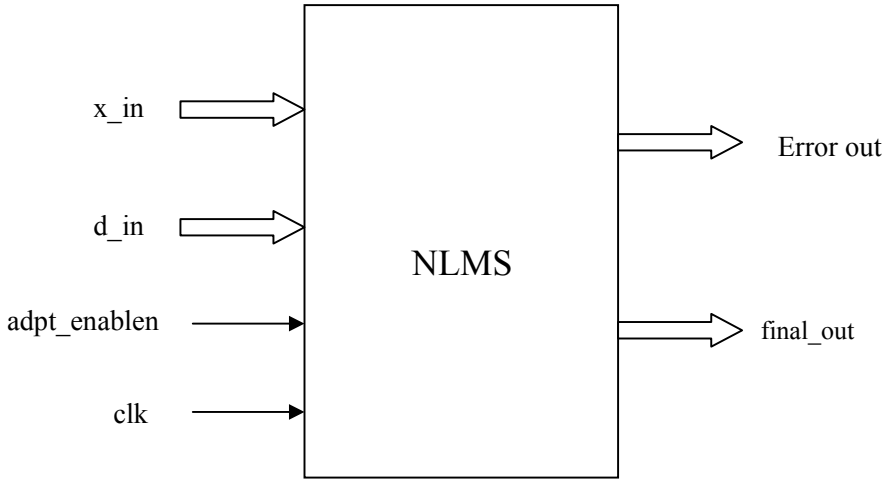


Figure.4.1: Input and output of NLMS Adaptive Algorithm

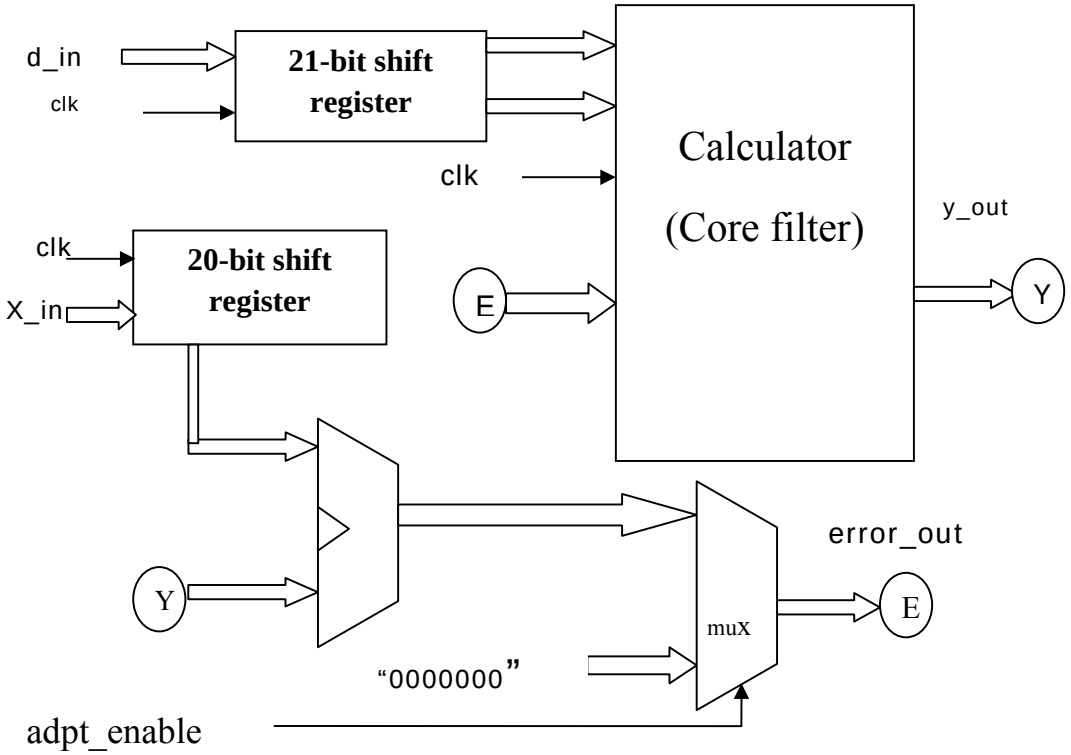


Figure.4.2: Inner Structure of NLMS

The above block shows the inner structure of arrangements of the internal blocks of NLMS. Here the input to the adaptive filter x_in is given to the 20-bit shift register and desired input d_in is given to the 21-bit shift register. The two outputs of the 21-bit shift register are given to the calculator block which generates the output y_out. The output of the 20-bit shift register is subtracted with the output of calculator y_out. The subtracted output is given to the multiplexer where the enable bit is used as a select line. The other input of the multiplexer is "00000000". The calculator block is used for calculation of y_out at several iterations. Here we used for five iterations. We can call the calculator block as the core filter block because it generates the filter output y_out.

When we considering the calculator block as main block which is used for several iterations, it consist of

five inputs and four outputs. In which only clock is the single bit input, all others are 8-bit data. The calculator block is as shown in figure 4.3.

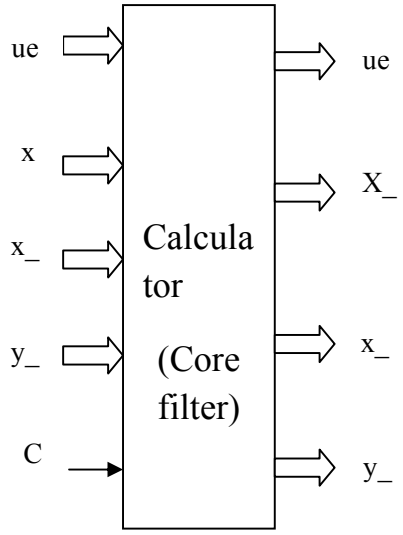


Figure.4.3: Block Diagram of Calculator

The inputs of core_filter are assigned to different sub-blocks of it. Outputs of core_filter are used as the inputs for next iterations. At last iteration only y_out is considered as the output of the core_filter.

The inner structure of the core filter block is as shown in fig 4.4. As mentioned above the unit calculator is used 5 times in single core filter for the calculation of y_out. the other outputs are left open which are of no use.

The calculator block has several internal blocks which performs the different arithmetic operations inside it.

The different blocks inside the core_filter are adder, multiplier, shift register, saturation, scaling, and truncation. The single calculator block uses some signals which are used for the calculation. The signals are shiftx(32), shiftxn(32), shiftue(24), shifty(16), coeff16(16), coeff8(8), xnin_ue(16), xnin_ue_scaled(16), new_coeff (16), delayed_new_coeff (16), y_out16(16) and y_out8(8). The numbers in bracket refers to the bit size of that signal.These signals are used during the different arithmetic operation of the input data at the calculator block.

The first block of the calculator is multiplier which multiplies the 8bit datas x_N_in and ue_in. the output signal xnin_ue is 16bit data. The second stage is scaling. In scaling stage the out may be upgrade or degrade of the input signal. Here we always degrade the output of the multiplier stage. The output is xnin_ue_scaled is also 16-bit, which is added to the 16-bit co-efficient signal coeff16 at next stage. The signal new_coeff is the output of the adder stage which is delayed by using shift register in next step. The delayed new coefficient which is the output of shift block is 16-bit data which is used in calculation of the next coefficient for core filter. The output checked for its limit in saturation stage. In saturation stage care is taken about the data not to exceed the limit of the coefficient. The saturated data is truncated to 8-bit in next stage.

The next stage is 8bit multiplier which multiplies the 8bit truncated coefficient with 8MSB bits of shiftx signal which gives 16-bit product. These 16 bit output i.e y_out 16 is truncated to 8-bit in next stage. The 8-bit output is the final output of the core_filter after 5th iterations.

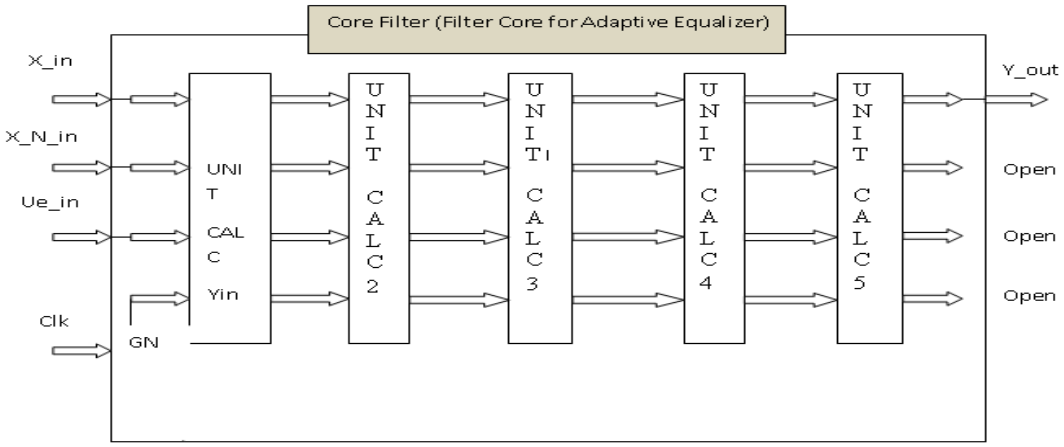


Figure 4.4: Inner Structure of the NLMS Core Filter

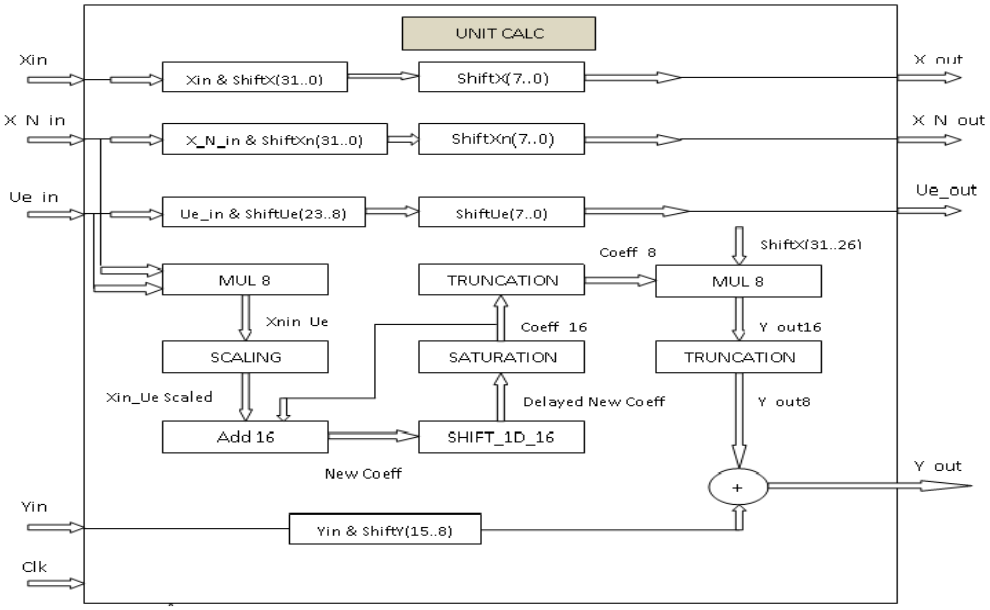


Figure 4.5 Inner Structure of the Calculator

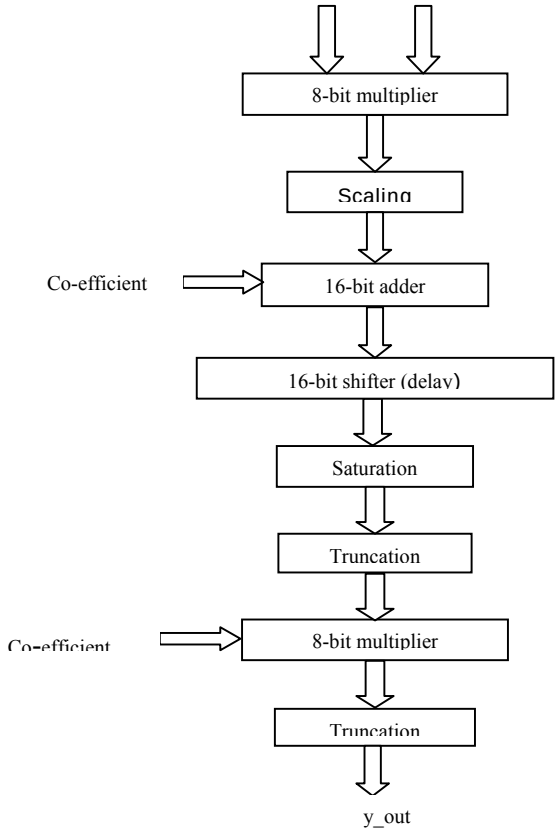


Fig.4.6: Internal Blocks of the Core Filter and its data flow

5. SYNTHESIS AND SIMULATION RESULTS

The implemented VHDL program for NLMS adaptive algorithm is now synthesised using Xilinx 10.1. Modelsim is used to study the waveforms of each stage. The VHDL programs are synthesised separately for each block. The synthesis result observed for core filter of the NLMS block and the total NLMS block separately are shown below

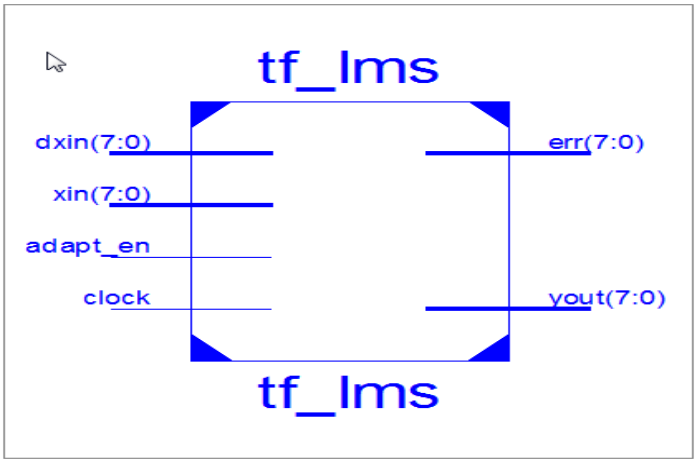


Figure 5.1Top Level Structure for NLMS Algorithm.

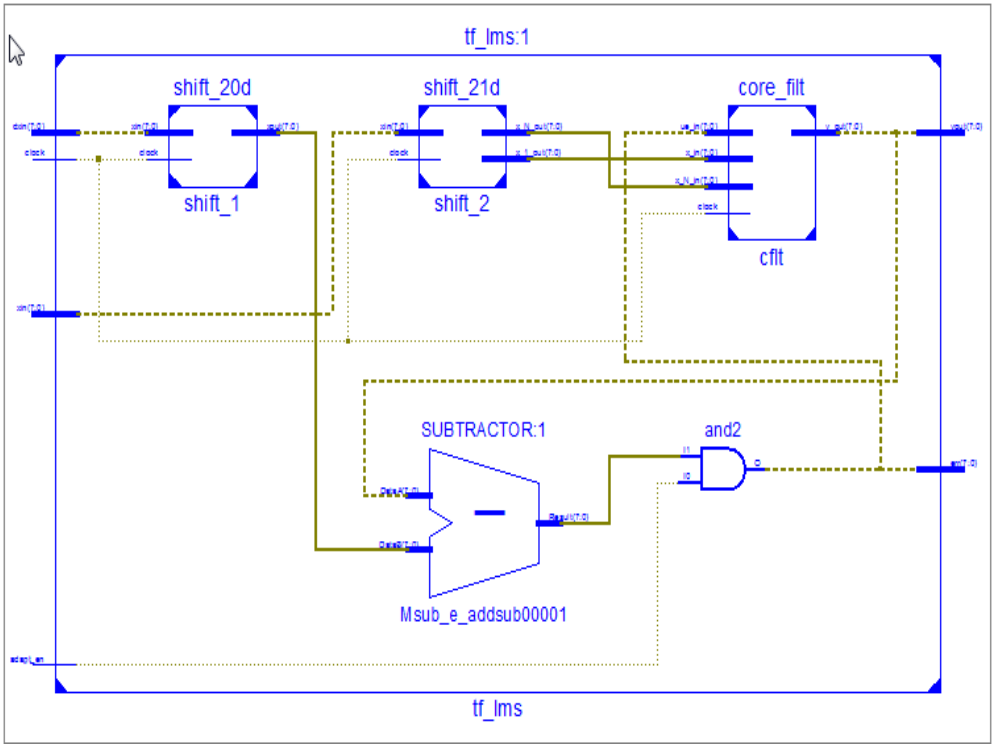


Figure 5.2 1st levels Internal Structure

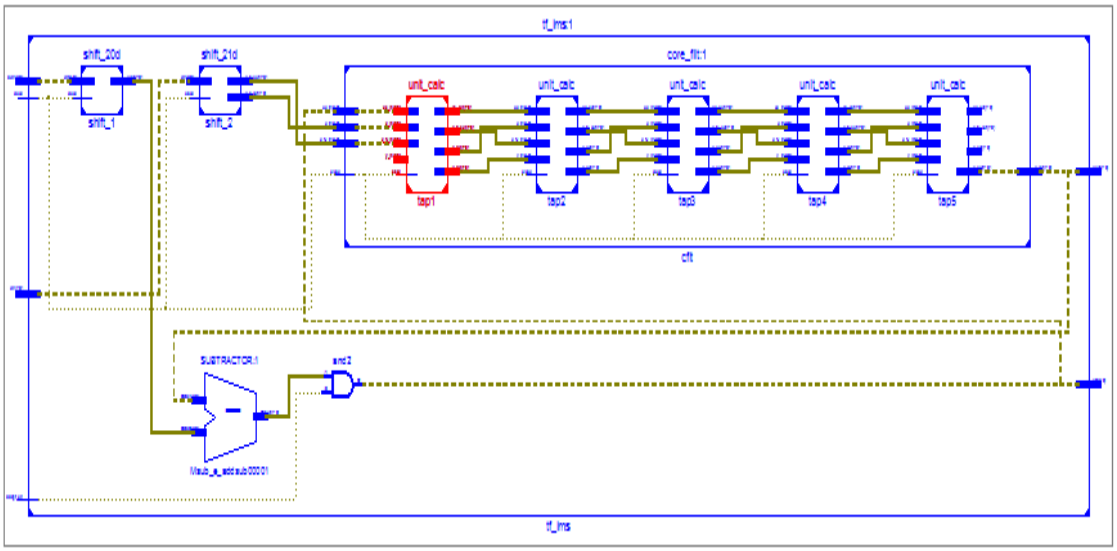


Figure 5.3 Internal Structure of Core filter

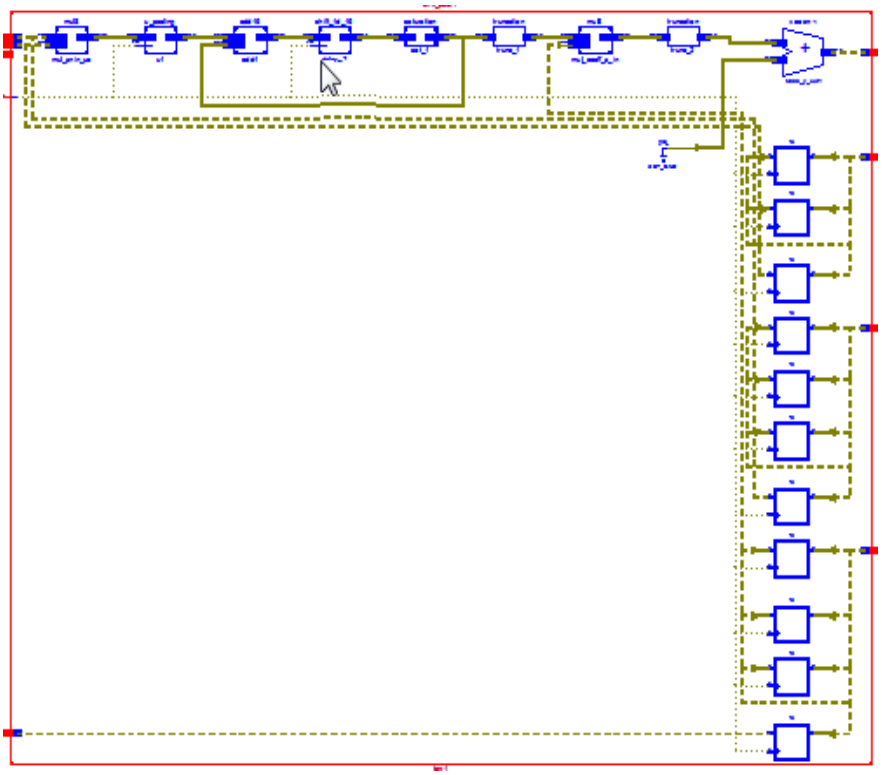


Figure 5.4 Internal Structure for Unit Calculators

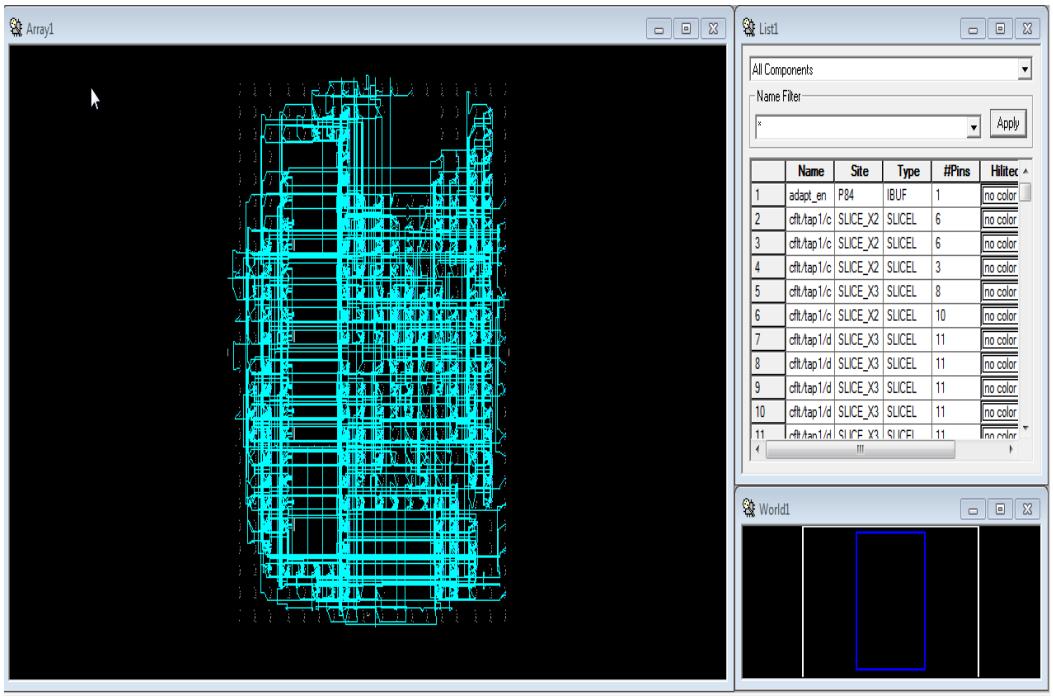


Figure 5.5 outed DesignFPGA R for NLMS

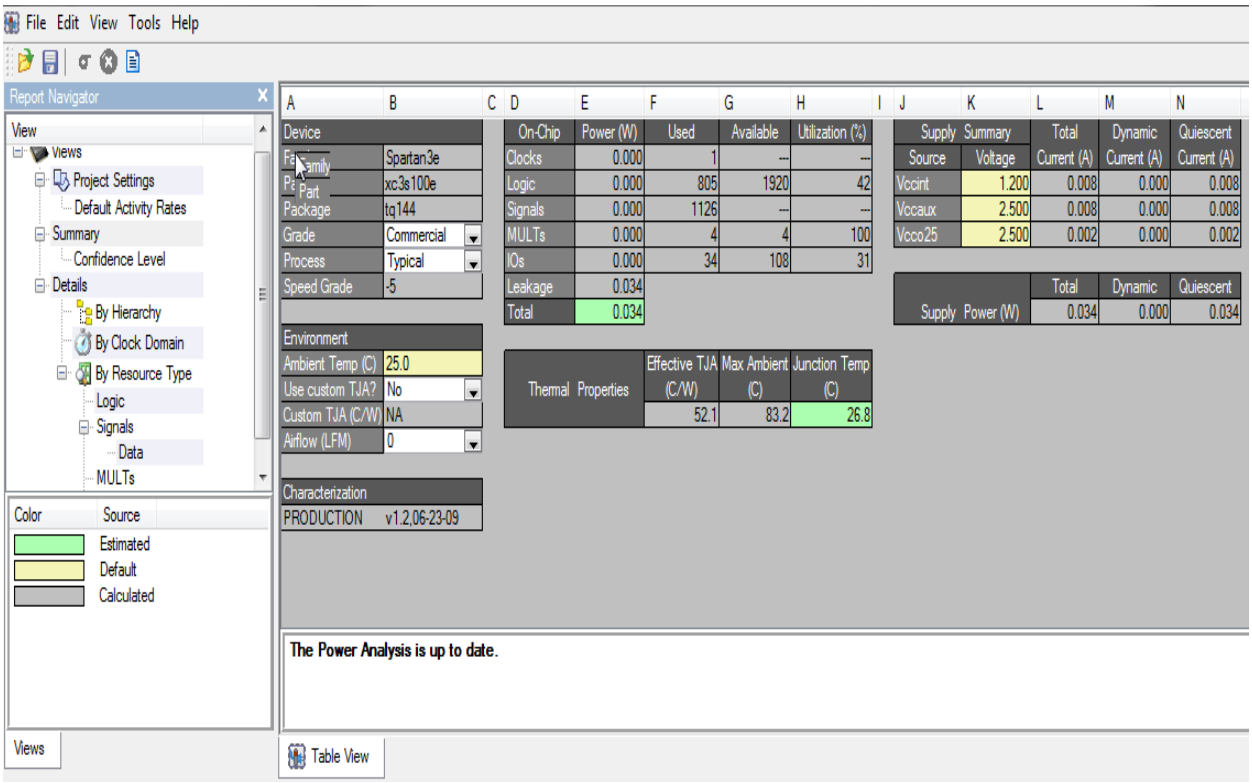


Figure 5.6 Power Analysis of NLMS using Xilinx X Power Analyzer

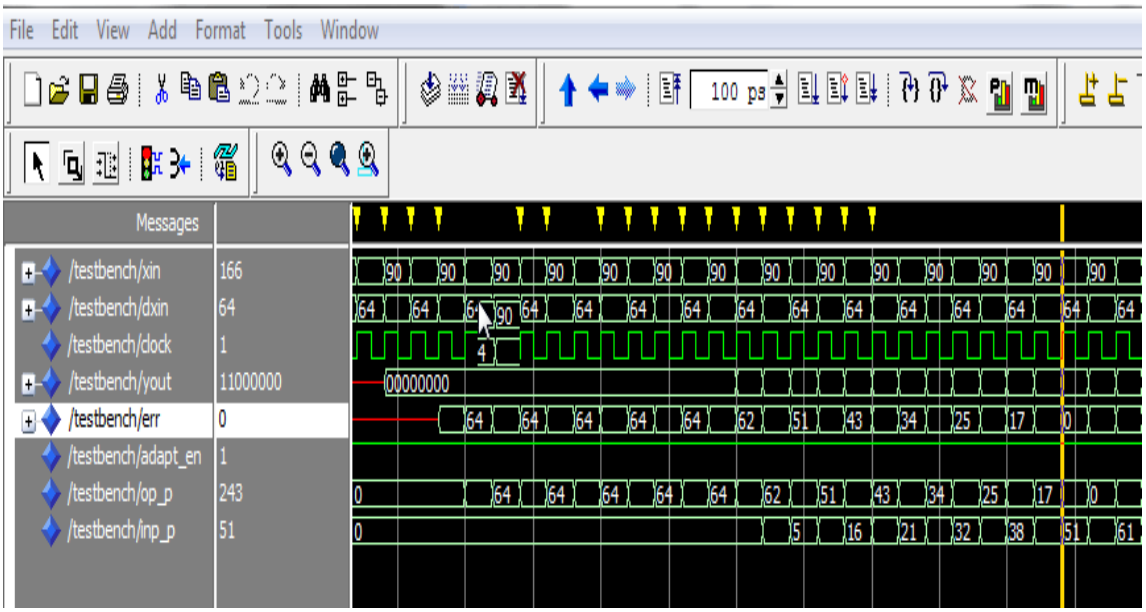


Figure 5.7 Top level Simulation Results for NLMS

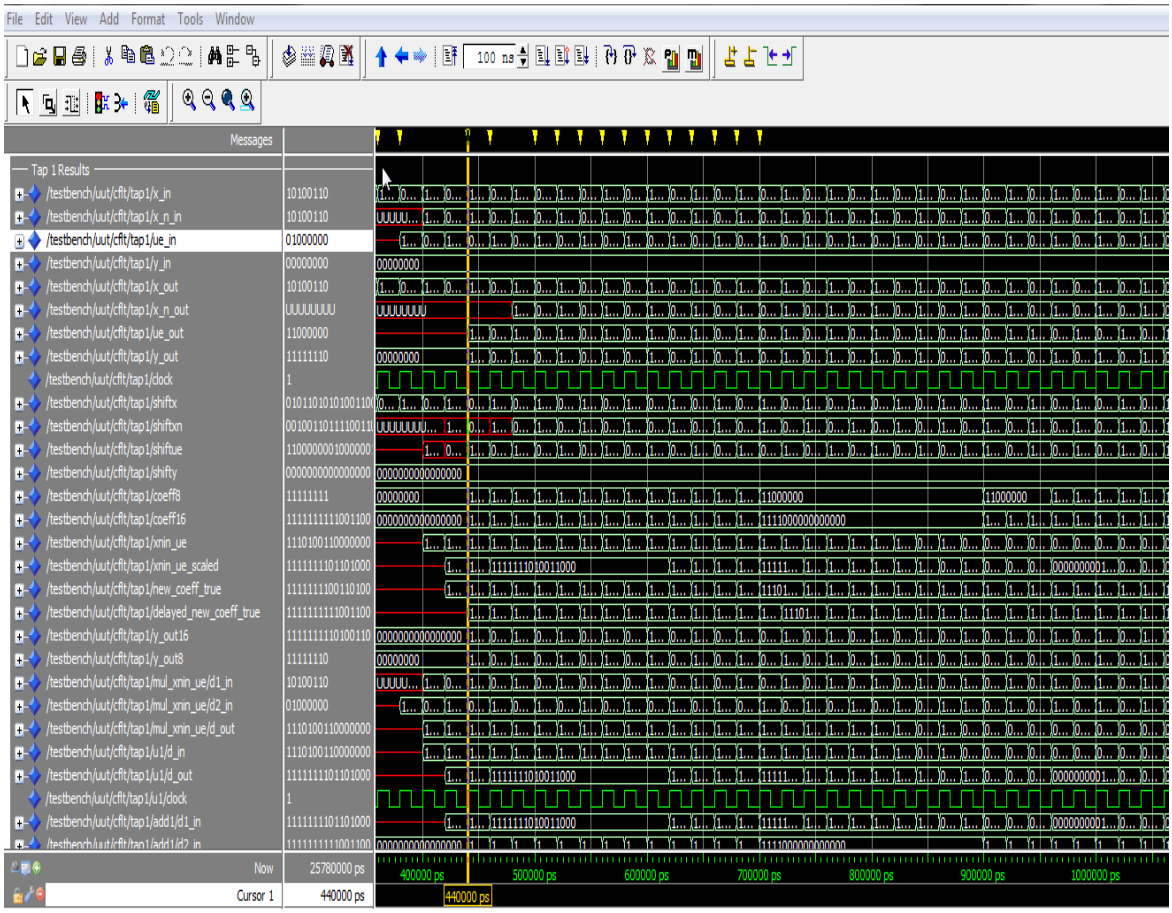


Figure 5.8 Tap 1 Simulation Result

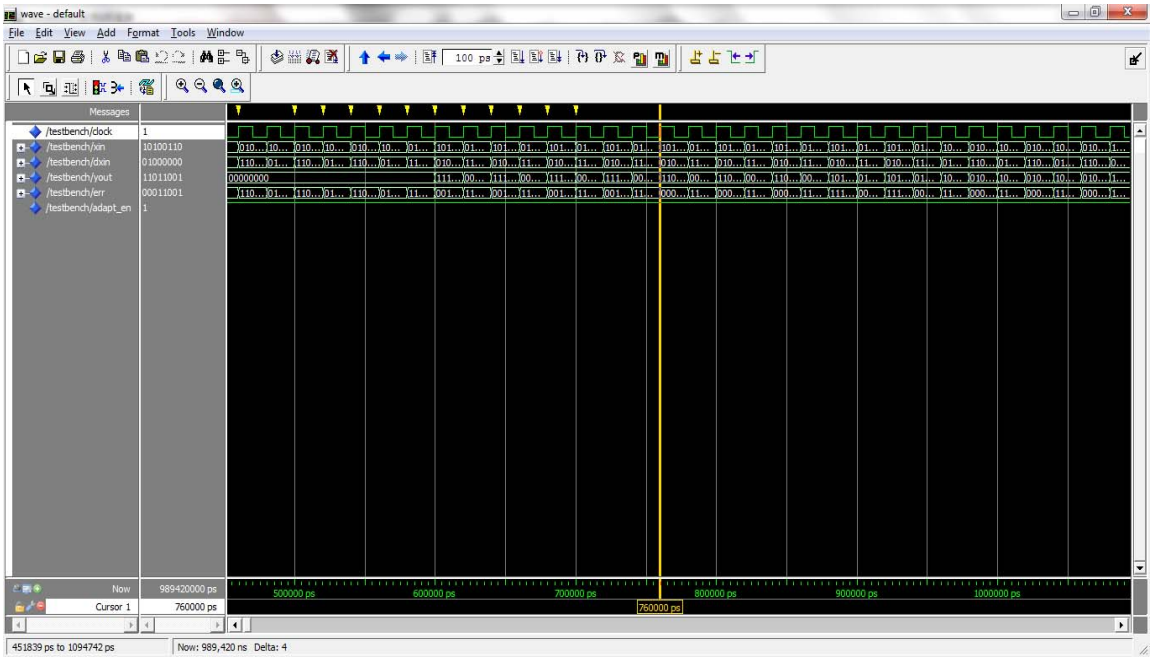


Figure 5.9 Simulation output for 2K Hz

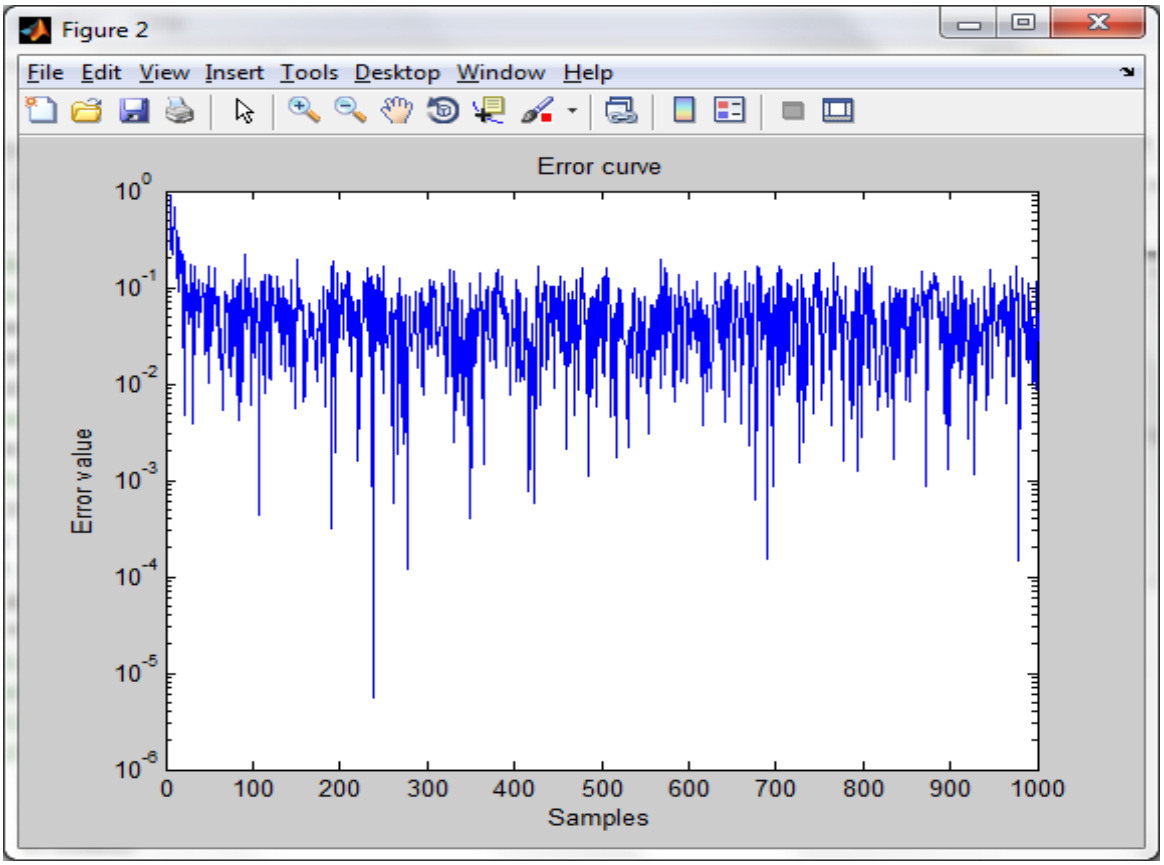


Figure 5.10 Error Curve

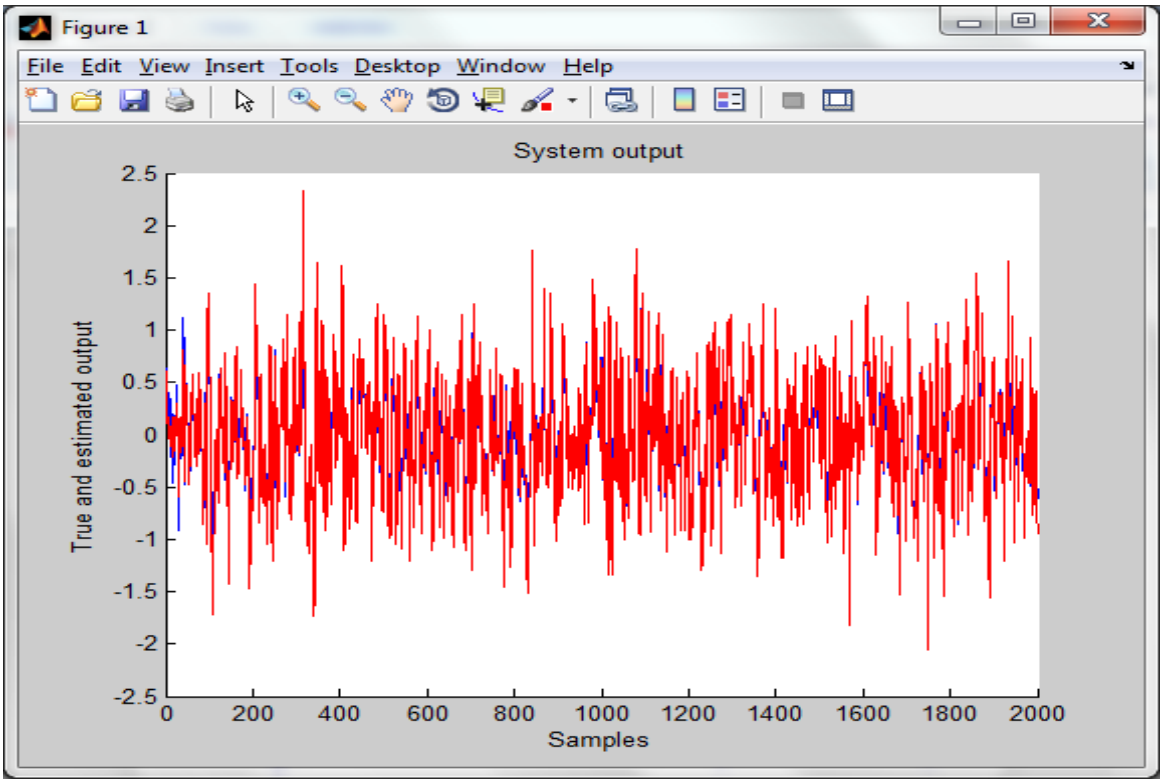


Figure 5.11 System Output

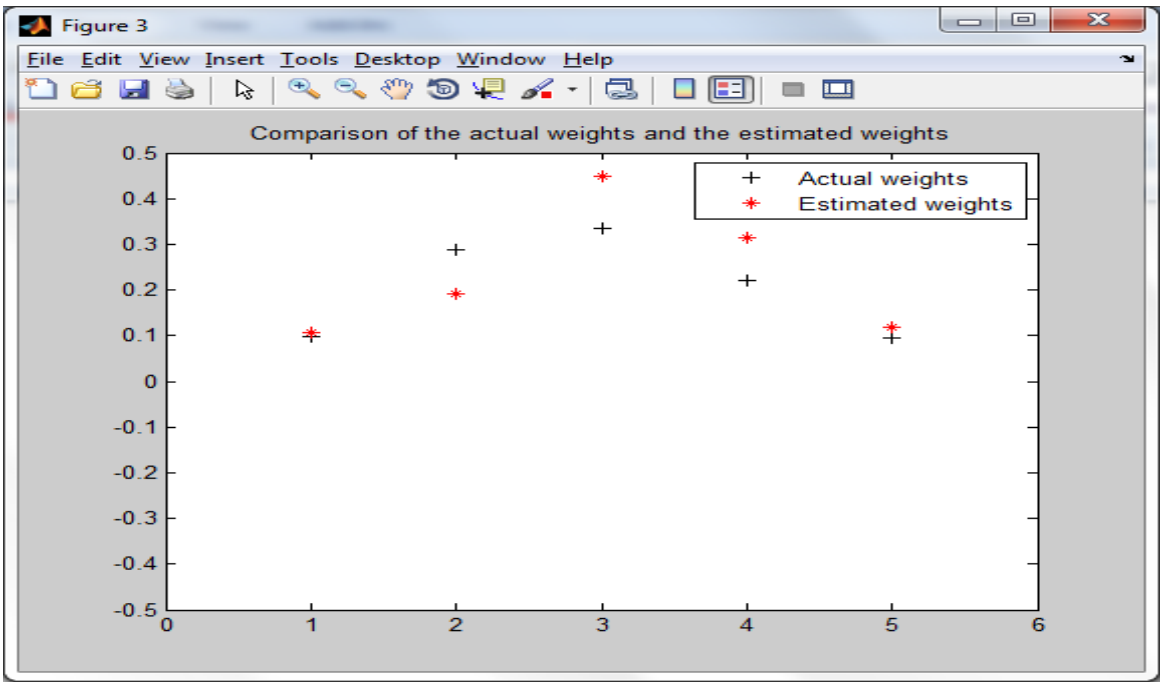


Figure 5.12 Comparison of the Actual Weights and the Estimated Weights

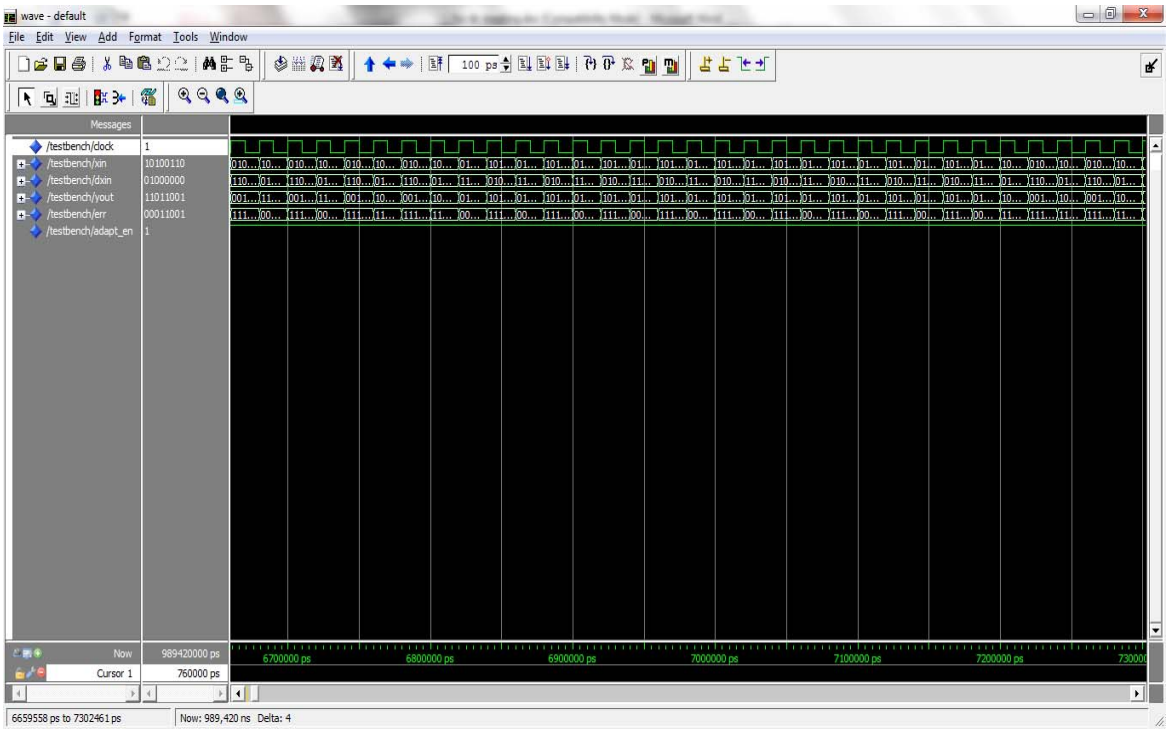


Figure 5.13 Simulation results for 20K

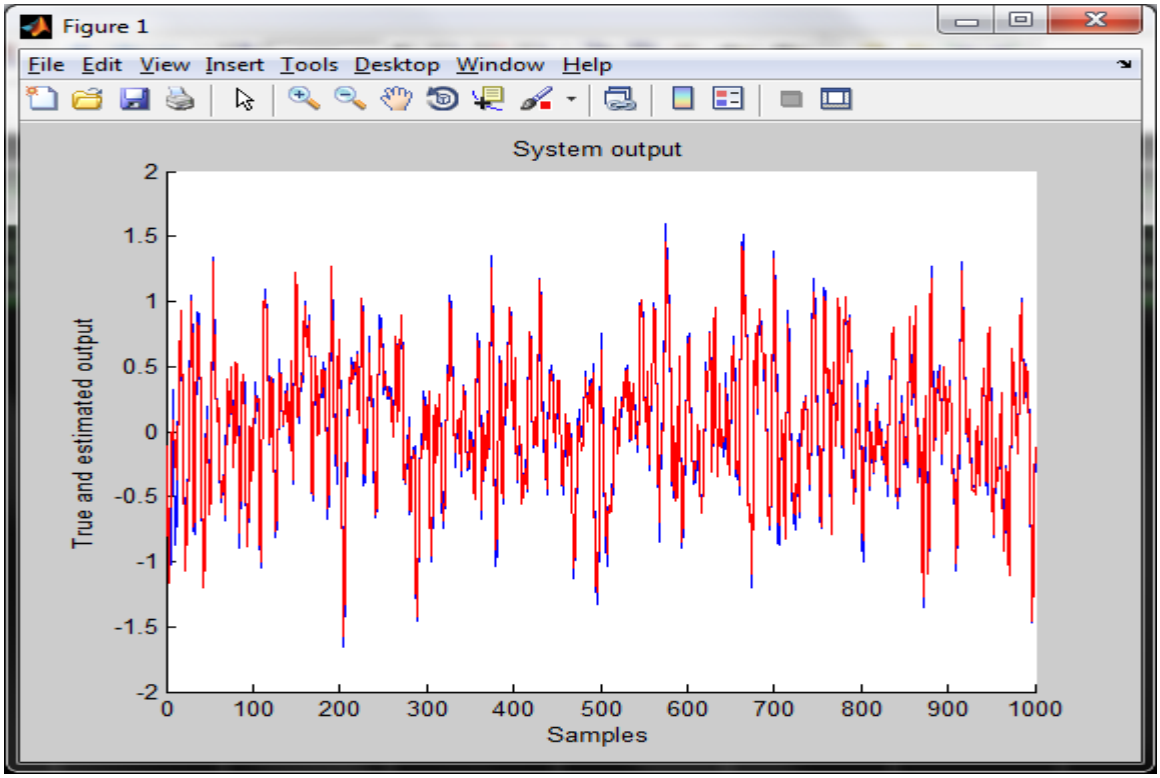


Figure 5.14 System Output

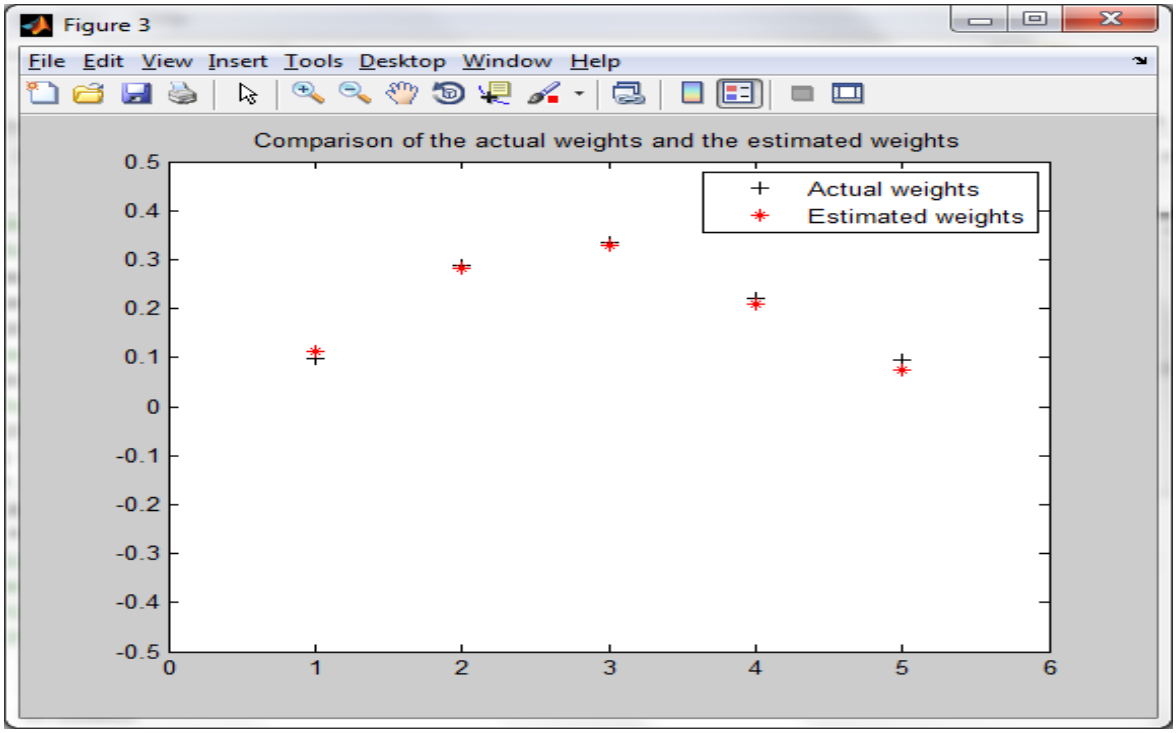
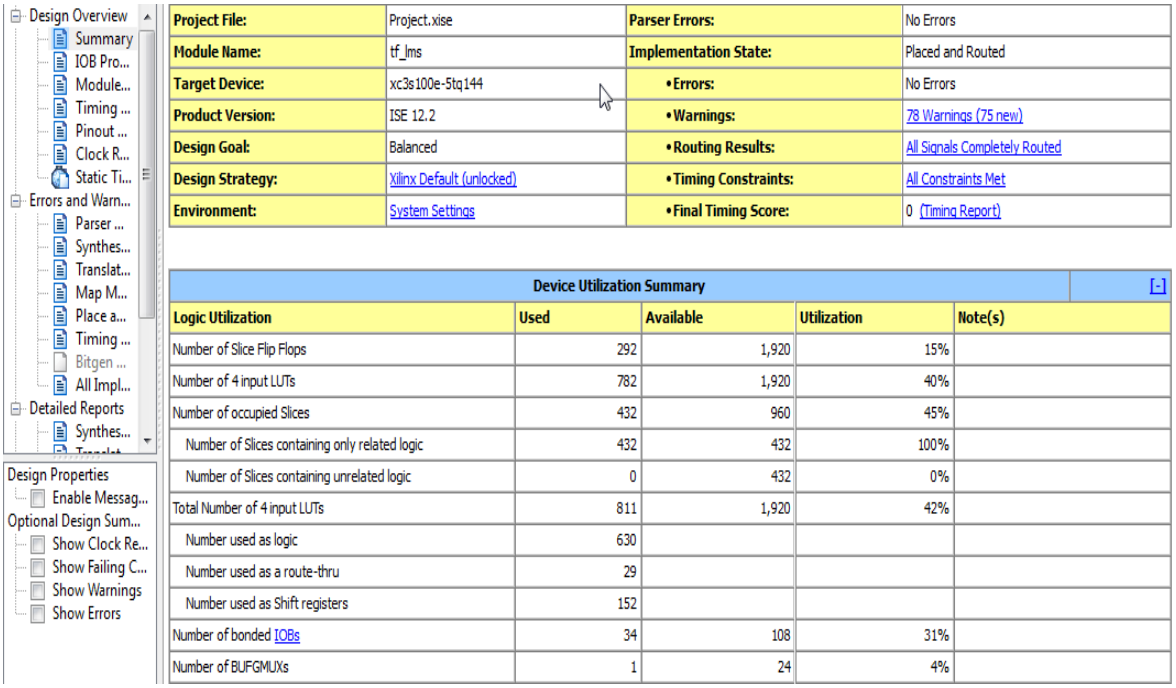


Figure 5.15 Comparison of the Actual Weights and the Estimated Weights



Design Overview
Summary
IOB Pro...
Module...
Timing ...
Pinout ...
Clock R...
Static Ti...
Errors and Warn...
Parser ...
Synthes...
Translat...
Map M...
Place a...
Timing ...
Bitgen ...
All Impl...
Detailed Reports
Synthes...
Design Properties
Enable Messag...
Optional Design Sum...
Show Clock Re...
Show Failing C...
Show Warnings
Show Errors

Project File:	Project.xise	**Parser Errors:**	No Errors
Module Name:	tf_lms	**Implementation State:**	Placed and Routed
Target Device:	xc3s100e-5tq144	**•Errors:**	No Errors
Product Version:	ISE 12.2	**•Warnings:**	78 Warnings (75 new)
Design Goal:	Balanced	**•Routing Results:**	All Signals Completely Routed
Design Strategy:	Xilinx Default (unlocked)	**•Timing Constraints:**	All Constraints Met
Environment:	System Settings	**•Final Timing Score:**	0 (Timing Report)

Device Utilization Summary [-]

Logic Utilization	Used	Available	Utilization	Note(s)
Number of Slice Flip Flops	292	1,920	15%	
Number of 4 input LUTs	782	1,920	40%	
Number of occupied Slices	432	960	45%	
Number of Slices containing only related logic	432	432	100%	
Number of Slices containing unrelated logic	0	432	0%	
Total Number of 4 input LUTs	811	1,920	42%	
Number used as logic	630			
Number used as a route-thru	29			
Number used as Shift registers	152			
Number of bonded IOBs	34	108	31%	
Number of BUFGMUXs	1	24	4%	

Figure5.16 Design Summary for Normalised Least Mean Square Algorithm

7. Conclusion

A review of adaptive filters shows that the NLMS algorithm is still a popular choice for its stable performance and high-speed capability. The other advantage of the NLMS over other adaptive algorithm is its high convergence rate.

The high-speed capability and register rich architecture of the FPGA is ideal for implementing NLMS. A hybrid adaptive filter is designed with a direct-form FIR filter coded in verilog and with the NLMS algorithm written in verilog code executing on the Xilinx output is simulated on MATLAB.
The fig 5.9 and fig 5.13 shows the ouput waveform for NLMS with 2K Hz and 20K Hz frequency input. We can see in the snapshot it contains of 3 different inputs namely expected input(the output of demodulator), desired input(), and clock with the oupts namely error signal showing the error generated due to the difference between the disered input and the expected input. We can also see that the error signal is slowly turned to 0 after some long iteration. Also we can see the final wave in

the graph called as y output of the NLSM. The fig 5.10 shows the graph of error curve generated as the output of Error signal at NLMS. We can see from the snapshot the error is approximately equal to 0 or less than zero. This is because we have directly considered the output of RRC demodulator which contains very less amount error.
The fig 5.11 and fig 5.14 shows the output of the system. We can see from the graph it contains two wave red as the true output and blue as the estimated output. The analysis done form the graph is that there is very silight difference between the two signal because of less error. If the error increases then the difference can go on increasing. Above fig 5.12 and fig 5.15 show the snpashot for the value of the weights for the NLMS. We can see from the snpashot that the difference between the two is very less as the error is less and the data is approximately the same as the desired input so the weights are not varied much. As the error in the wave goes on inreasing the difference aslo goes on increasing so the the outpt can be approximated to the desired signal.

REFERENCES

[1] B. Widrow and S.D.Stearns," Adaptive Signal Processing", Prentice-Hall, Englewood Cliffs, N.J., 1985.

[2] S. Haykin, "Adaptive Filter Theory", Fourth Edition, Prentice Hall, Upper Saddle River, N.J., 2002.

[3] Choo, P. Padmanabhan, S. Mutsuddy, „An Embedded Adaptive Filtering System on FPGA", Department of Electrical Engineering, San Jose State University, CA 95198-0084 USA.

[4] Scott C. Douglas, "Analysis of the Multiple-Error and Block Least-Mean-Square Adaptive algorithms", IEEE Transactions on Circuits and Systems – II: Analog and Digital Signal Processing, Vol 42, No. 2, p. 92, February 1995.

[5] Lattice Semiconductor Corporation, "LMS Adaptive Filter", Reference Design RD1031, December 2006.

[6] Sinead Mullins, Conor Heneghan, "Alternative Least Mean Square Adaptive Filter Architectures for Implementation on Field Programmable Gate Arrays", Digital Signal Processing Group, Department of Electronic and Electrical Engineering, University College Dublin.

[7] Ahmed Elhossini, Shawki Areibi, Robert Dony, "An FPGA Implementation of the LMS Adaptive Filter for Audio Processing", IEEE International Conference on Reconfigurable Computing and FPGA's, ReConFig 2006 ,ISBN: 1-4244-0690-0.

[8] S. Haykin, Adaptive Filter Theory, Prentice Hall, New Jersey, 1996.

[9] B. Farhang-Boroujeny, Adaptive Filters: Theory and Applications, John Wiley & Sons, West Sussex, England, 1998.

[10] J. Homer, —Adaptive Echo Cancellation in Telecommunications", Ph.D. dissertation, The Australian National University, Canberra, April 1994.

[11] S.C. Douglas and T.H.-Y. Meng, —Normalized Data Non-linearities for LMS Adaptation", IEEE Transactions on Acoustics Speech Signal Process, vol. 42, pp 1352-1365, 1994.

[12] T.C. Hsia, —Convergence Analysis of LMS and NLMS Adaptive Algorithms", Proceedings for ICASSP, Boston, Mass., pp 667-670, 1983.

[13] HDL CHIP DESIGN by Douglas J Smith, 1997 Edition.

[14] D.Gajaski and R. Khun, "Introduction: New VLSI Tools", IEEE Computer, volume 16, Dec 1983.

[15] Digital System Design Using VHDL by Charles H Roth, Jr PWS Publication Company 1998 Edition.

[16] Circuit Design with VHDL by Volnki A.Pedroni, MIT Press 2004, PP 3-4.

[17] A Modified Least Mean Square Algorithm Using Fractional Derivative and its Application to System Identification, Raja Muhammad Asif Zahoor, International Islamic University, Ijaz Mansoor Qureshi, Air University, Pakistan, Vol.35 No.1 (2009), pp 14-21 Euro Journals Publishing, Inc. 2009.

[18] "HDL Implementation of the Variable Step Size N-LMS Adaptive Algorithm" I. Dornean1, M. Topa1, B.S. Kirei1, G. Oltean1, Technical University of Cluj-Napoca.

[19] Chew, W. C., Farhang-Boroujeny, B., FPGA Implementation of Acoustic Echo Cancelling. Proceedings of the IEEE Region 10 Conference TENCON 1999, vol. 1, pp. 263-266, 1999.

[20] Diniz, P. S. R., Adaptive Filtering – Algorithms and Practical Implementation. 2nd Edition, Kluwer Academic Publishers, Norwell, Massachusetts, 2002.

[21] Diniz, P. S. R., da Silva, E.A.B. and Netto, S.L., Digital Signal Processing – System Analysis and Design. Cambridge University Press, Cambridge U.K., 2002.

[22] Douglas, S. C., Zhu, Q. and Smith, K. F., A Pipelined LMS Adaptive FIR Filter Architecture without

Adaptation Delay. IEEE Transactions on Signal Processing, vol. 46, no. 3, pp. 775-779, Mar. 1998.
[23] Eweda, E., Reducing the Effect of Finite Wordlength on the Performance of an LMS Adaptive Filter. IEEE International Conference on Communications, vol. 2, pp. 7-11, Jun. 1998.
[24] Eweda, E., Convergence analysis and Design of an Adaptive Filter with Finite-bit Power-of-Two Quantized Error. IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing, vol. 39, issue 2, pp. 113-115, Feb. 1992.
[25] Guillou, A., Quinton, P., Risset, T. and Massicotte, D., Automatic Design of VLSI Pipelined LMS Architecture, Proceedings in International Conference on Parallel Computing in Electrical Engineering, pp. 144-149, 2000.
[26] Goslin, G. R., A Guide to Using Field Programmable Gate Arrays (FPGAs) for Application-Specific Digital Signal Processing Performance, Digital Signal Processing program report, Xilinx Inc., 1995.
[27] Gupta, R. and Hero, A.O., Transient Behavior of Fixed Point LMS Adaptation. Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing, vol. 1, pp. 376-379, April, 2000.
[28] Haykin, S. Adaptive Filter Theory. 4th edition, Prentice Hall, Upper Saddle River, New Jersey, 2002.
[29] Nichols, K., Moussa, M. and Areibi, S., Feasibility of Floating Point Arithmetic in FPGA based ANNs. In Proceedings of the 15th International Conference on Computer Applications in Industry and Engineering, pp. 8-13, November 2002.
[30] Papoulis, A. and Pillai, S.U., Probability, Random Variables and Stochastic Proceses. 4th edition, McGraw-Hill, New York, New York, 2001.
[31] Schertler, T., Cancellation of Acoustic Echoes with Exponentially Weighted Step-Size and Fixed Point Arithmetic. Conference records of the 32nd Asilomar Conference on Signals, Systems and Computers, vol. 1, pp. 399-403, November 1998.
[32] Song, M.S., Yang, P.P.N. and Shenoi, K., Nonlinear Compensation for Finite Word Length Effects of an LMS Echo Canceller Algorithm Suitable for VLSI Implementation. Proceedings of International Conference on Acoustics
[33] G. Long, F. Ling and J. G. Proakis, "The LMS algorithm with delayed coefficient adaptation," IEEE Trans. on ASSP, vol. 37, Sept. 1989, pp. 1397-1405.
[34] C.-L.Wang, "Bit-serial VLSI implementation of delayed LMS adaptive FIR filters," IEEE Trans. Signal Process., vol. 42, Aug. 1994, pp. 2169–2175.
[35] L. K. Ting, R. F. Woods and C. F. N. Cowan, "Virtex FPGA Implementation of a Pipelined Adaptive LMS Predictor for Electronic Support Measures Receivers," IEEE Trans. VLSI Syst., vol. 13, Jan. 2005, pp. 86-95.
[36] M. D. Meyer and D. P. Agrawal, "A high sampling rate delayed LMS filter architecture," IEEE Trans., Circuits Syst. II, Analog Digit. Signal Process., vol. 40, Nov. 1993, pp. 727–729.
[37] L.-K. Ting, "Algorithms and FPGA implementations of adaptive LMS-based predictors for radar pulse identification," Ph.D. dissertation, Queen's Univ. Belfast, N. Ireland, Jul. 2001.
[38] P. Waldeck and N. Bergmann, "Evaluating software and hardware implementations of signal-processing tasks in an FPGA," in Proc. IEEE International Conference on Field-Programmable Technology, Brisbane, Australia, Dec. 2004, pp. 299-302.
[39] A. Elhossini, S. Areibi and R. Dony, "An FPGA Implementation of the LMS Adaptive Filter for Audio Processing," in Proc. IEEE International Conference on Reconfigurable Computing and FPGAs, Sept. 2006.
[40] S. Haykin, Adaptive Filter Theory, Prentice-Hall, third edition,

K. R. Rekha obtained her ME degree from Bangalore University, India in 2000. She is working as a Asst Professor in the Department of Electronics and Communication in SJB Institute of Technology, Bangalore. Presently, she is pursuing her Ph. D. degree in Dr MGR University, Chennai. Her research interests include Wireless communication, FPGA implementation, and Microcontroller and Embedded systems design. She is a member of MIE, MISTE and IETE

Dr B. S. Nagabushana obtained his M. Tech. and Ph. D. degrees from My sore University and Indian Institute of Science, Bangalore respectively. He has wide academic as well as software industrial experience for over 25 years. He has worked as Professor in various engineering.
colleges as well a consultant to software industries like OMED (Software), Japan, BFL, CG-Smith Software Pvt. Ltd, KPIT Cum Info systems (Bangalore), Pvt. Limited, San Lab Technologies etc. His research interests include Wireless Communication, Neural Network, and Fuzzy Logic and Embedded systems. He is the recipient of NRDC Independence Day award for the year 1992, Best Project Execution award for the year 2000 from M/s BMC

Dr K. R. Nataraj obtained his ME degree from Bangalore University, India in 2000. He worked as Professor and Head of the Department during 2000-2008 and currently he is the Post Graduate Coordinator in the Department of Electronics and Communication in SJB Institute of Technology, Bangalore. Presently, His research interests include Wireless communication, FPGA implementation, and Microcontroller and Embedded systems design. He is a member of MIE, MISTE , IETE and IEEE.