ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 2, February 2014

Efficient VLSI architecture of SISO based Turbo decoder for wireless applications

S.Badrinarayanan[1], J.M.Mathana[2], R.Rani Hemamalini[3]

Research Scholar, Vinayaka Missions University, Salem, India [1]

Professor, Dept. of ECE, S.A Engineering College, Chennai, Tamilnadu, India[2]

Professor& Head, Dept. of ECE, St.Peter's College of Engg.& Tech, Chennai,Tamilnadu, India[3]

ABSTRACT: Error correction codes are the essential components of the digital communication and data storage system to ensure robust operation of digital applications wherein turbo code is one of the most attractive near optimal error correction codes. Speed is one of the key factors beside the power consumption and area usage for efficient implementation of turbo decoders. In high speed digital communication such as broadband wireless access based on IEEE 802.16e standard and the fourth generation cellular systems, the design of turbo decoder with high throughput is a critical issue. Since turbo decoders inherently have a long latency and low throughput due to the iterative decoding process. This paper presents VLSI architecture for an efficient soft input soft output based turbo decoder using sliding window method. The speed of operation of the implemented architecture is improved by modifying the value of the branch metrics. The intended SISO based decoder architecture has been implemented using Verilog HDL at RTL level and synthesized to investigate its performance in terms of area usage and timing delay.

KEYWORDS: SISO, ML-MAP, BMU, SMU, LLR, APP.

I.INTRODUCTION

Turbo codes were introduced in 1993 by Berrou et.al. [1]. The advantage of turbo codes in the communication system is that they enable reliable communication with performance close to theoretical limit given by Claude Shannon [2].With the wide deployment of wireless networks, there has been tremendous interest in designing turbo decoders. There have been considerable research efforts in both the areas of iterative decoder design and FPGA based computing platforms. Turbo codes have reasonable complexity and provide powerful error correcting capability for various block lengths and code rates. In order to address the circuit complexity issue, a reduced complexity turbo decoder specifically optimized for contemporary FPGA devices [3] has been developed using decoder run time dynamic reconfiguration in response to variations in the channel conditions.

A channel decoder chip compliant with the 3GPP mobile wirelessstandard has been described by Bickerstaff et al [4].The implementation of a low power Log-MAP decoder with reduced storage requirement based on the optimized MAP algorithm that calculates the reverse state metrics in the forward recursive manner has been reformulated by IndrajitAtluri et al [5].The original MAP based turbo decoder has been modified by Saeed Ghazanfari Rad et al [6] to obtain a reduced complexity scheme of turbo decoder. In order to provide interleaved data at the speed of the decoder, a 16 bit single instruction multiple data processors has been equipped with processing elements. In order to reduce the internal bit width of the state metrics and thereby to decrease the entire energy dissipation of a turbo decoder, a technique has been described by Haisheng LIU et al [7]. Hardware architecture for modified Max-Log-MAP (MLMAP) algorithm using MacLaurin series has been provided by Rahul Shrestha and Roy Paily[8] to reduce the complexity.The performance of the architecture has been improved by replacing all the multipliers with shifters and adders. Algorithmic approximation and architectural optimization has been incorporated in the design of radix-8 Log-MAP turbo decoder [9] to reduce the critical path and achieve high throughput of 693 Mbps. But the hardware complexity is linearly increased. A combinational logic cell with four subtractors has been introduced by Martin I.del Barco et al [10] to improve the speed of the turbo decoder at the cost of increased hardware. Double flow and shuffled turbo decoding scheme has been presented by Jaesung Choi and Jeong Woo Lee [11].

The proposed reconfigurable turbo decoder based on ML-MAP algorithm with Sliding Window (SW) technique provides flexibility to choose constraint lengths of 3,4&5 and also reduces the critical path of a turbo decoder and there by provides high throughput rate. This is achieved by a new design of branch metric unit and state metric allocation technique.

II.PROPOSED ITERATIVE SISO DECODER

The original turbo code employed two Recursive Systematic Convolutional (RSC) encoders concatenated in parallel and separated by a pseudo-random interleaver. Each rate 1/2 RSC encoder produces a set of systematic and parity bits. The overall code is broken down into its constituent parts at each decoder and each constituent code can be decoded easily because of its inherent structure [12]. The encoder for constraint length K = 5 is shown in figure 1.The encoder consists of four shift registers and four modulo-2 adders.

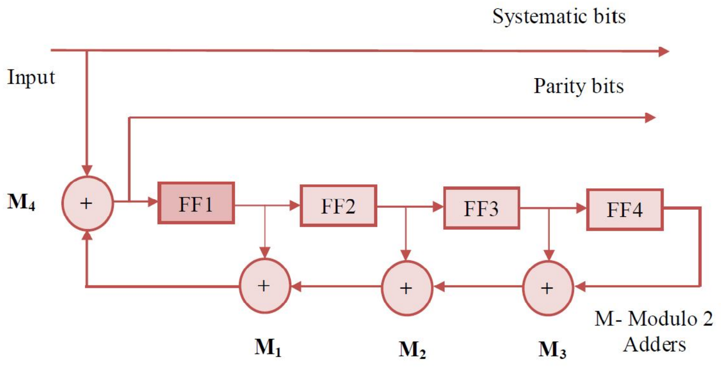


Fig. 1 Turbo Encoder

The main components in the SISO decoder are forward state metric unit, backward state metric unit, dummy backward state metric unit and likelihood ratio computation unit. The architecture of proposed reconfigurable SISO decoder is shown in figure.2

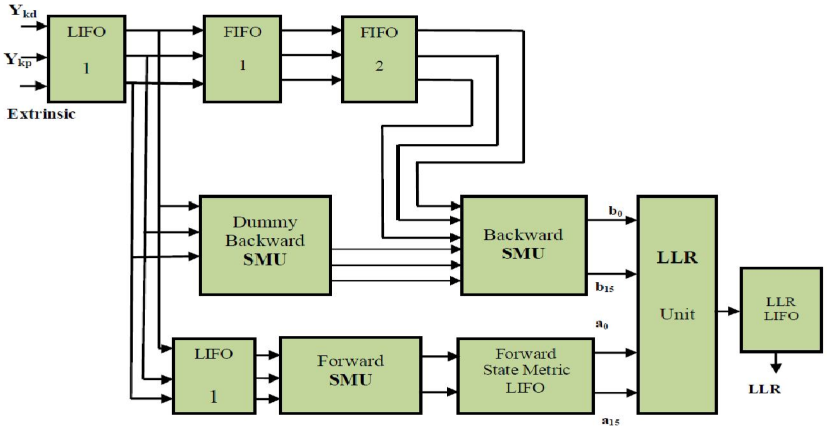


Figure.2 Proposed reconfigurable SISO decoder

In figure 2, last in first out (LIFO) and first in first out (FIFO) are memory units used to store the input data symbols and they accompany the SW method. Similarly State Metric LIFO and LLR LIFO are used to store the computed state metrics and log likelihood ratio values respectively. The conventional MAP decoding process has very high latency due to the processing of forward and backward state metric calculations in all trellis states. Computing the LLR values require the state metric values generated by the forward and backward processes. Therefore, a large memory size.is required to store the state metric values which in turn depend on the input data block size. The SW method is used proposed work to reduce the memory size by dividing the input data into sub-blocks.

III. BRANCH METRIC UNIT

The first computational block in the turbo decoding algorithm is the Branch Metric Unit (BMU). The conventional branch and state metric unit [13]consists of branch metric calculation, add, compare, select and normalization processes. In the proposed work, turbo decoder design can be simplified by considering the insensitivity of Max-Log-MAP algorithm to the AWGN channel variance (No) of the noise. If the AWGN channel variance of the noise is ‘2’ in branch metric equation, then there will be no multiplication or division. Then the four branch metric values are

$$\ln\gamma_{k,00}(S',S) = (-Y_{kd} - Y_{kp}) + \ln(Ap_k(-1)$$
$$\ln\gamma_{k,01}(S',S) = (-Y_{kd} + Y_{kp}) + \ln(Ap_k(-1)$$
$$\ln\gamma_{k,10}(S',S) = (+Y_{kd} - Y_{kp}) + \ln(Ap_k(+1)$$
$$\ln\gamma_{k,11}(S',S) = (+Y_{kd} + Y_{kp}) + \ln(Ap_k(+1)$$

The new design of BMU based on the above equation is shown in figure 3.

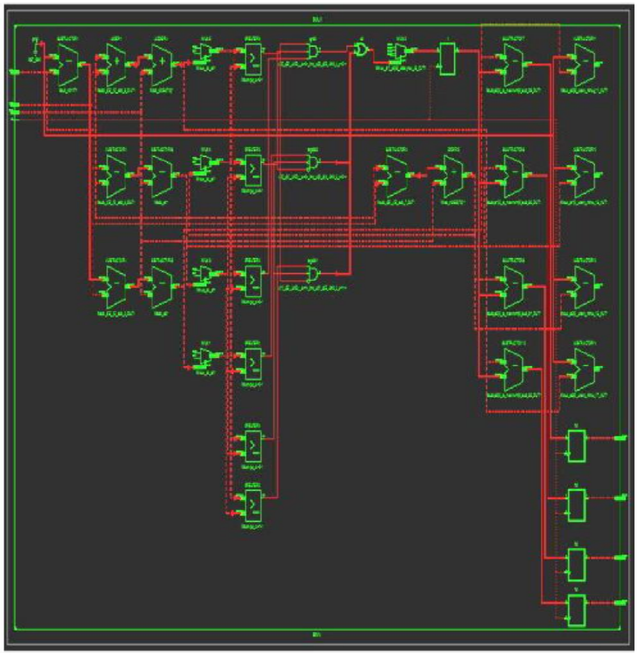


Figure 3 RTL Diagram of Proposed BMU

The input symbol data and the soft output feedback are computed to generate the branch metric values. The branch metric values are added to the state metric values to generate the new state metric value for the next cycle. In general, State Metric Unit (SMU) in SISO decoder should include the normalization process to avoid overflow of the state metric value. It should be noted that, the state metrics keep on increasing as the recursion goes on [14].This normalization in state metric value leads to a complex SMU and also reduces the speed of the device. This problem is nullified in the proposed architecture by means of normalizing the branch metric values.The generated branch metric values are converted into absolute values (A) which are then compared to select the maximum or minimum branch metric value. These processes are performed in one clock cycle recursively. The branch metric normalization is done by the following equation

$$\gamma^{'}(S^{'}_{k-1},S_k) = \gamma\,(S^{'}_{k-1},S_k) - \max[\gamma\,(S^{'}_{k-1},S_k)]$$
$$\gamma^{''}(S^{'}_{k-1},S_k) = \gamma\,(S^{'}_{k-1},S_k) - \min[\gamma\,(S^{'}_{k-1},S_k)]$$

In the above equation, γ is the branch metric value, γ’ is the normalized value by maximum branch metric value, γ’’ is the normalized value by minimum branch metric value. Therefore γ’ is always equal to zero or less than zero and γ’’ is always equal to zero or larger than zero.

The normalized branch metric values are used to compute state metric values. Since the branch metric values are normalized, the finite length of the state metric values is reduced. Hence state metric value does not require normalization. This branch metric normalization method leads to a simple SMU (add, compare and select) as shown in figure 4.This structure reduces the critical path delay by eliminating the state metric normalization process used in the conventional SMU. This Add Compare Select unit (ACS) is the basic unit to compute the state metric values.

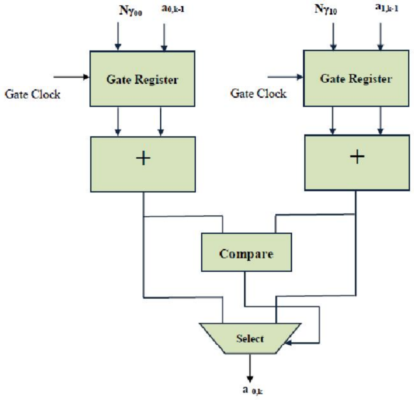


Figure 4 ACS Unit

IV. RECONFIGURABLE STATE METRIC UNIT

The forward state metric ‘a’ is the next step of computation in the algorithm which represents the probability of a state at time ‘k’ given the probabilities of states at previous time instance. It is calculated using equation

$$\ln \alpha_k(S) = \max_{s^{'}}[\alpha_{k-1}(s^{'}) + \gamma_k(S^{'},S)]$$

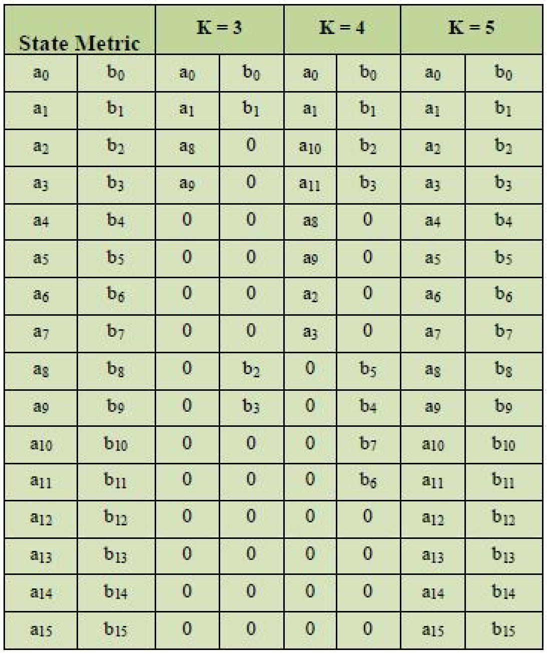
The backward state probability being in each state of the trellis at each time ‘k’, given the knowledge of all the future received symbols is recursivelycalculated and stored. The backward state metric ‘b’ is computed using the following equation

$$\ln \beta_k(S^{'}) = \max_k[\beta_{k+1}(S^{'}) + \gamma_{k+1}(S^{'},S)]$$

in the backward direction going from the end to the beginning of the trellis at time instance ‘k-1’, given the probabilities at time instance ‘k’.The backward state metric computation can start only after the completion of the computation by the branch metric unit. State Metric value for a particular node is computed based on the trellis diagram of the encoder. In the reconfigurable SMU, the add compare select (ACS) units are recursively processed to compute the state metrics, through the connection network that allocates the state metrics for the next ACS based on the current constraint length ‘K’ value. For a particular constraint length ‘K’, this state metric allocation must be done before they are fed as input to the ACS in the next clock cycle.

Table I describes how allocation of state metric values is done for constraint lengths K=3, K= 4 and K = 5. The state metrics of the forward and backward processes are reordered by constraint length ‘K’ after the allocation. The proposed reconfigurable SMU is shown in figure 5. Thereconfigurable SMU consists of totally 16 ACS starting from ACS0 to ACS15.

TABLE I ALLOCATION OF STATE METRIC VALUES



State Metric		K = 3		K = 4		K = 5	
a_0	b_0	a_0	b_0	a_0	b_0	a_0	b_0
a_1	b_1	a_1	b_1	a_1	b_1	a_1	b_1
a_2	b_2	a_8	0	a_{10}	b_2	a_2	b_2
a_3	b_3	a_9	0	a_{11}	b_3	a_3	b_3
a_4	b_4	0	0	a_8	0	a_4	b_4
a_5	b_5	0	0	a_9	0	a_5	b_5
a_6	b_6	0	0	a_2	0	a_6	b_6
a_7	b_7	0	0	a_3	0	a_7	b_7
a_8	b_8	0	b_2	0	b_5	a_8	b_8
a_9	b_9	0	b_3	0	b_4	a_9	b_9
a_{10}	b_{10}	0	0	0	b_7	a_{10}	b_{10}
a_{11}	b_{11}	0	0	0	b_6	a_{11}	b_{11}
a_{12}	b_{12}	0	0	0	0	a_{12}	b_{12}
a_{13}	b_{13}	0	0	0	0	a_{13}	b_{13}
a_{14}	b_{14}	0	0	0	0	a_{14}	b_{14}
a_{15}	b_{15}	0	0	0	0	a_{15}	b_{15}

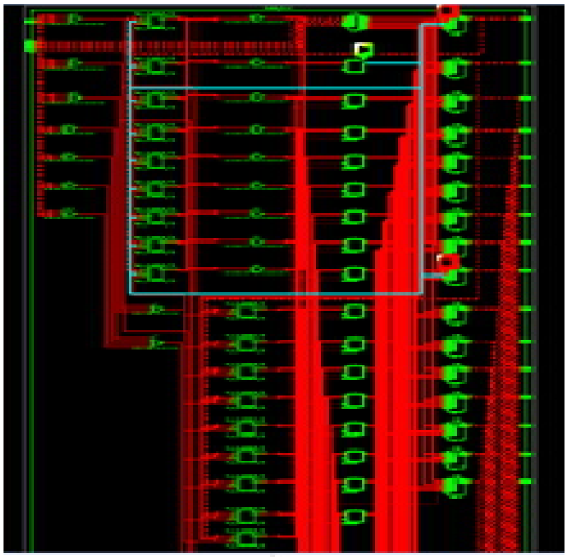


Figure 5 RTL diagram of Proposed Reconfigurable Architecture of SMU for Forward Processes

V. PROPOSED RECONFIGURABLE LLR COMPUTATION UNIT

Log likelihood ratio is the output of the turbo decoder. The LLR for each symbol at time 'k' is calculated using the equation

$$LLR(X_{k+1}) = \ln \frac{\sum_{(S',S),X=1} \gamma_{k+1}(S',S).\beta_{k+1}(S)\alpha_k(S')}{\sum_{(S'S),X=-1} \gamma_{k+1}(S',S).\beta_{k+1}(S)\alpha_k(S')}$$

The main operations involved in LLR computation are comparison, addition and subtraction. Finally these values are de-interleaved at the second decoder output after the required number of iterations to make the hard decision in order to retrieve the information that is transmitted. The sign of the number corresponds to the hard decision while the magnitude gives a reliability estimate. In order to compute LLR value, forward; backward state metric values and branch metric values of all states are required.The proposed reconfigurable LCU consists of two identical blocks which calculates the LLR of bit 0 and bit 1 respectively. The maximum calculated value of LLR1 and LLR0 is subtracted to get the final LLR output value. The sign of a posteriori value gives the value of decoded bit 1 or 0. The LLR block is pipelined to reduce the critical path delay.

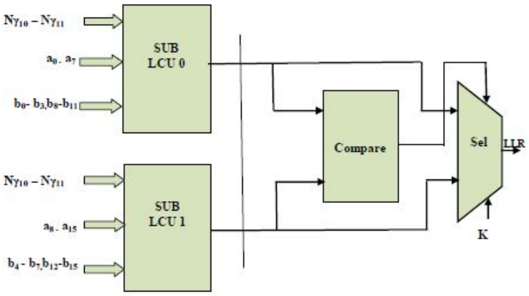

$N\gamma_{10} - N\gamma_{11}$
$a_0 - a_7$
$b_0 - b_3, b_8 - b_{11}$
SUB LCU 0
$N\gamma_{10} - N\gamma_{11}$
$a_8 - a_{15}$
$b_4 - b_7, b_{12} - b_{15}$
SUB LCU 1
Compare
Sel
LLR
K

Figure 6 Proposed Reconfigurable LCU

Figure 6 shows the structure of the proposed reconfigurable LCU consisting of two sub-LCUs, one compare and one select unit. The architecture of Sub-LCU makes the difference between the conventional and proposed LCU.

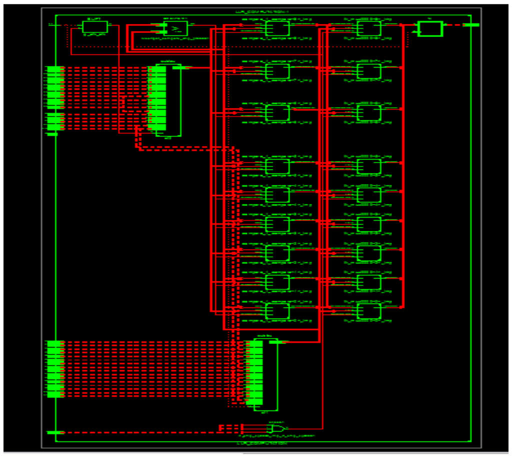


Figure 7 RTL Diagram of proposed SISO Decoder

The output of the LCU is determined by compare and select units with constraint length 'K' and associated with LLR0 or LLR1. Each sub-LCU can be mapped to forward, backward state metric values and the branch metrics. The conventional LLR Computation Unit (LCU) is implemented by tree structure consisting of compare and select units.

VI.PERFORMANCE ANALYSIS OF THE PROPOSED RECONFIGURABLE TURBO DECODER

The proposed Max-Log-MAP turbo SISO decoder is initially simulated at high level to verify its functionality with Model simulator 6.4 Edition. The design has been synthesized using Xilinx ISE 12.2 FPGA to investigate its area usage and time delay. The intended reconfigurable decoder architecture has been implemented using Verilog HDL at RTL level and synthesized to investigate its performance.

The complete panorama of the area requirement results is explicitly shown in the figure 8. From the figure, it is evident that the reconfigurable LCU module utilizes more slices in comparison with the other basic modules of SISO decoder. It is due to its configurability with different constraint lengths K = 3, 4, 5.

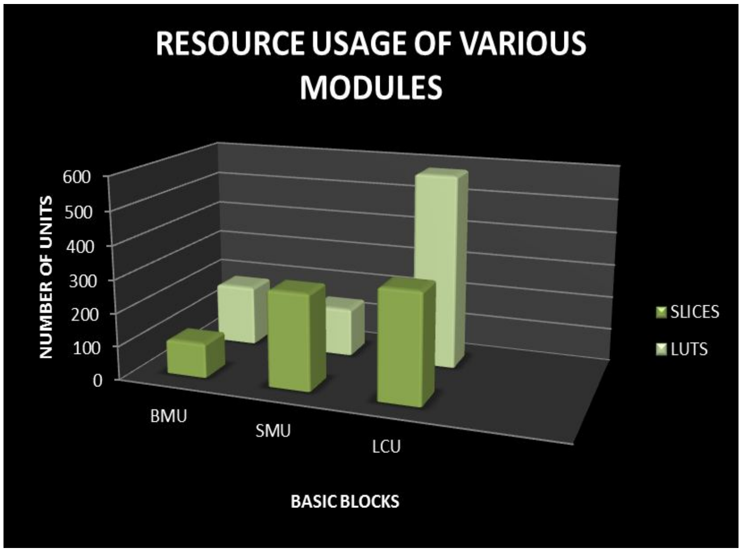


Figure 8.Comparision of area utilization of various modules

The combinational path delay and area utilization of reconfigurable and non-reconfigurable SMU, LCU architectures are obtained from the synthesis report which is furnished in Table.II and III respectively.

Table IIPATH DELAY AND AREA COMPARISON OF RECONFIGURABLE AND NON RECONFIGURABLE SMU

MODULE	CONVENTIONAL	PROPOSED
BMU	**23.789 ns**	**9.036 ns**
SMU	**15.884 ns**	**9.284 ns**
LCU	**23.207 ns**	**6.210 ns**

Table III COMPARISON OF NUMBER OF SLICES

MODULE	CONVENTIONAL	PROPOSED
BMU	72	104
SMU	321	288
LCU	363	329

From the combinational path delay comparison table II, the path delay for various modules are known and the maximum combinational path delay in the SMU block of the decoder is found as 9.284 ns and also it is observed that, in the proposed reconfigurable SMU, the critical path delay is only 9.284 ns resulting 86.08 % speed-up compared to the conventional SMU. Similarly, the speed ofthe proposed LCU is improved by 73.24 % over conventional one.

Table III compare the number of slices utilized by the proposed reconfigurable and conventional BMU, SMU and LCU architectures [15].Since the proposed architecture of BMU includes the normalization process, more number of slices are utilized. It reduces the finite length of the state metric values. The method of branch metric normalization reduces the critical path delay by eliminating the state metric normalization process used in the conventional SMU.

VII. CONCLUSION

The paper has presented a turbo soft-in soft-out decoder based on ML-MAP algorithm. The proposed reconfigurable turbo decoder has a critical path delay of only 9.284 ns resulting 86.08% speed-up in the proposed SMU compared with conventional design. In non-reconfigurable architecture maximum delay is caused by LCU while in the proposed architecture the maximum delay is due to SMU. Comparing the non-reconfigurable and the proposed reconfigurable LCUs, the delay time for the reconfigurable LCU is reduced by 26.76%. Future works consist of implementing low latency reconfigurable turbo decoder architecture and analyze the impact of the reconfigurability on the speed of the turbo decoder.

REFERENCES

[1] A. Glavieux, C. Loat, nad J. Labat, " Turbo Equalization over Frequency Selective Channel," in Proc. Int. Symp. Turbo Codes and Related Topics, Best France, pp. 96-102, Sep. 1997.

[2] C. Berrou, A. Glavieux, P. Thitimajshima, "Near Shannon Limit Error Correcting Coding and Decoding: Turbo Codes," in proc. IEEE Intl. Conf. Commun., (Geneva), pp. 1064-1070, May 1993.

[3] Jian Liang, Tessier, R. and Goeckel,D. "A dynamically-reconfigurable, power efficient turbo decoder", Proceedings of 12th Annual IEEE Symposium on Field Programmable Custom Computing Machines, Napa, California, pp.91-100 , 2004.

[4] Bickerstaff, M. A., Garrett, D., Prokop, T., Thomas, C., Widdup, B., Gongyu, Z., Davis, L.M., Woodward Nicol, G. and Yan, R.H. "A unified Turbo/Viterbi channel decoder for 3GPP mobile wireless in 0.18 µm CMOS ", IEEE J. Solid-State Circuits, Vol.37, No. 11,pp. 1555-1564, 2002.

[5] IndrajitAtluri, Ashwin K., Kumaraswamy and Chouliaras, V.A."Energy efficient architectures for the log-map decoder through intelligent memory usage", Proceedings of IEEE Computer Society Annual Symposium on VLSI: New Frontiers in VLSI Design, Tampa, Florida, pp. 263-265, 2005.

[6] Saeed Ghazanfari Rad, V.T. Vakili and Falahati, A. "A new MAP-based algorithm to reduce the complexity of Turbo Decoder", Proceedings of IEEE 9th International Conference on Advanced Communication Technology, Gangwon, Korea, pp.1058- 1061, 2007.

[7] Haisheng LIU, Jean-Philippe DIGUET, Christophe JEGO, Michel JEZEQUEL and Emmanuel BOUTILLON," Energy Efficient Turbo Decoder With Reduced State Metric Quantization", Proceedings of IEEE workshop on Signal Processing Systems, Shanghai, China, pp. 237-242, 2007.

[8] Shrestha, R. and Paily, R. "Hardware implementation of Max-Log-MAP algorithm based on MacLaurin series for turbo decoder", Proceedings on International Conference in Communications and Signal Processing (ICCSP), pp. 509-511, 2011.

[9] Faxun Jin, Jian Tang, Zhongfeng Wang, and Li Guo, "A High Speed Radix-8 Log- MAP Recursion VLSI Architecture", Proceedings of 11th IEEE international Conference on Communication Technology, Hangzhou, pp.347-350, 2008.

[10] RMart´ın I. del Barco, Gabriel N. Maggio, Dami´an A. Morero, Javier, Fern´andez, Facundo Ramos, Hugo S. Carrer, and Mario R. Hueda ," FPGA Implementation of High Speed Parallel Maximum Posteriori (MAP) Decoders" , Proceedings of the Argentine School of Micro-Nano Electronics, Technology and Applications ISBN 978-9- 8725-1029-9, 2009.

[11] Jaesung Choi, Jeong Woo Lee, " Study on High Throughput Turbo Decoder", Proceedings of IEEE 73rd Vehicular Technology Conference, Budapest, Hungary, pp.1 –5, 2011.

[12] Montorsi, G. and Benedetto, S. "Design of fixed point Iterative Decoders for Concatenated Codes with Interleavers ", Proceedings of IEEE Journal on selected Areas in Communication,Vol.19, No.5, pp.871-882,2001.

[13] Wang, Z., Suzuki, H. andPahri, K. K. "VLSI implementation issues of turbo decoder design for wireless applications ", Proceedings of IEEE International workshop on Signal Processing System, pp. 503-512,1999.

[14] Wang, Z. "High-speed recursion architecture for MAP-based Turbo decoders", IEEE Trans. VLSI Trans. VLSI System. Vol.14,No. 4, pp.470-474, 2007.

[15] Imran Ahmed and TughrulArslan, "VLSI Design of Multi Standard Turbo Decoder for 3G and beyond", Proceedings on Design automation Conference, Asia and Pacifi Design Automation Conference ,ASP-AC'07,Yokohama ,Japan,pp.589-594, 2007.