International Journal of Advanced Technology in Engineering and Science
Vol. No. 12, Issue No. 05, May 2024
www.ijates.com
ijates
ISSN 2348 - 7550

VLSI Implementation of Turbo Coder for LTE using Verilog HDL

Mrs. L. SARITHA[1], B. Pujitha[2], G. Pavani[3], D. Kusuma[4], B. Sambasiva Rao[5]

[1]Assistant Professor, M. Tech Department of Electronics and communication Engineering, Tirumala Engineering College, Narasaraopet, Palnadu District, AndhraPradesh, India.

[2,3,4,5]UG Students Department of Electronics and communication Engineering, Tirumala Engineering College, Narasaraopet, Palnadu District, AndhraPradesh, India.

Abstract—

Turbo codes are error correction codes that are widely used in communication systems. Turbo codes exhibits high error correction capability as compared with other error correction codes. This paper proposes a Very Large Scale Integration (VLSI) architecture for the implementation of Turbo decoder. Soft-in-soft-out decoders, Interleavers and deinterleavers is used in the decoders side which employs Maximum-a-Posteriori (MAP) algorithm. The number of iterations required to decode the information bits being transmitted is reduced by the use of MAP algorithm. For the encoder part, this paper uses a system which contains two Recursive Convolutional encoders along with pseudorandom interleaver in encoder side.

Index Terms—Turbo codes, Channel coding, Interleaver, SISO,
Iterative decoder, MAP, Cadence, NClaunch, Xilinx Vivado

I. INTRODUCTION

Errors may occur in the received signal at the source end of a communication system during the transfer of data from the source system to the destination system. Therefore, in order to recover the original message, error correction is needed. The 1993 introduction of turbo codes marked a revolution in channel coding methods and a turning point for contemporary digital telecommunication. One of the most effective error-correcting codes now in use is Turbo. With its ability to attain close channel capacity by iterative decoding using only simple component code, Turbo Codes has influenced the coding community. A turbo encoder plus a turbo decoder make up the turbo coder architecture (Fig. 1). Two Recursive Convolutional Encoders (RSC) and an interleaver make up the encoder. In this work, Because a pseudo-random interleaver is utilized, the interleaved version of the code is typically large and jumbled, which improves random code speed. RSC encoders are used in turbo code implementation instead of traditional convolutional encoders since they provide low weight parity codes. The MAP algorithm is used to verify that the decoded data is error-free after decoding turbo encoded data, where faults are purposefully introduced.

II.LITERATURE REVIEW

Kavinilavu, Salivahanan, and Bhaaskaran in their work [1] develop and integrates the Convolutional encoder and Viterbi decoder. In Model Sim 10.0e they have

planned and modeled and synthesized using XILINX-ISE 12.4i. Max Log MAP algorithm-based turbo decoder output variations on implemented with fixed point, Vedic and Booth multipliers have been presented by authors in paper [2]. A basic turbo coding strategy is proposed in [3] to maximize the error quality of a standard rate-1/3 turbo code. In the paper [4] the authors introduced the high-speed turbo SISO Decoder. Standardization operation was applied to state metric branch values rather than branch metric values. A minimum- power, and area-efficient turbo soft-output (SISO) decoder based on the Viterbi algorithm are proposed in [5]. The paper presents the implementation of SOVA decoder for different constraint lengths. Compared to a conventional SOVA decoder application, simulation results show power savings and area savings. The designers in [6] developed a turbo decoder architecture in which utilizes both parallel SISO decoder tier and parallel trellis stage level.In the LTE-Advanced standard, the authors in paper [7] present the design and implementation of a memory reduced Turbo decoder on the field programmable gate array (FPGA). In this paper [8] the author presents a summary of the architecture issues relevant to turbo decoders. As a feature of the code's key parameter, an evaluation of different types of turbo decoders is carried out. The authors in paper [9] proposed a new low-complexity Min-Log-MAP algorithm variant in their study. The encoding of tail-biting codes using hierarchical input encoders is done in paper [10], which is an important design criterion. Authors in their work [11] explore the different methods used in turbo codes to end trellis in the recursive systematic convolutional encoders. With minimum generator polynomials, the author in [12] has achieved very low rate convolutional codes. For different code rates and for different constraint lengths, the authors in the paper [13], the Sum Product Algorithm (SPA) and One Step Majority Logic Decoding Algorithm (OSMLGD)is put together to decode the LDPC codes and their individual performances were compared.

III. IMPLEMENTATION

A. Architecture of Turbo Coder

The Basic Block Diagram of Turbo Coder consists of Turbo Encoder and Turbo Decoder. A channel for transmitting data from encoder to decoder. Turbo encoder produces an encoded output which is input to turbo decoder, that produces a decoded output by removing the errors and reducing the number of iterations by using MAP Algorithm and get back the original output.

The Encoder produces a codeword with randomlike properties. Two identical Recursive convolutional encoder(RSC) and a pseudo random inter-leaver constitutes the turbo encoder. The Decoder makes use of soft-output values and iterative decoding. It consists of two modules of SISO decoders together with two pseudo random inter-leaver and a pseudo random deinter-leaver.

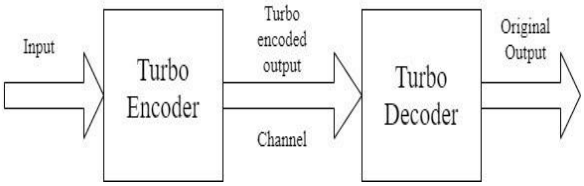


Fig. 1. Turbo Coder Block diagram

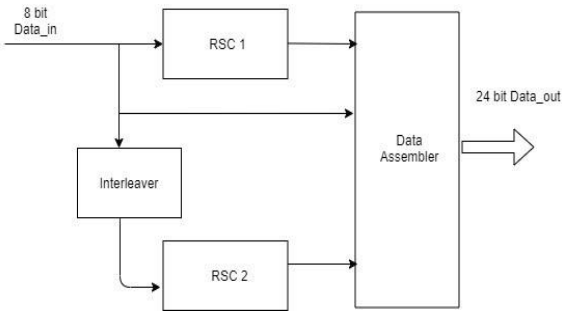


Fig. 2. Turbo Encoder Block Diagram

International Journal of Advanced Technology in Engineering and Science
Vol. No. 12, Issue No. 05, May 2024
www.ijates.com
ijates
ISSN 2348 - 7550

Turbo encoder produces an encoded output which is an input to turbo decoder. A turbo code is formed from the parallel concatenation of two codes separated by an inter- leaver. The two encoders normally used are identical. Encoders are recursive systematic convolutional codes (RSC). The inter-leaver reads the bits in a psuedo-random order. The fundamental turbo code encoder is built using two recursive systematic convolutional (RSC) codes with parallel concatenation.

LTE employs a 1/3 rate parallel concatenated turbo code. Each+ RSC works on two different data. Original data is provided to first encoder, while the second encoder receives the interleaved version of input data. A specified algorithm is used to scramble the data bits and the method is called interleaving.

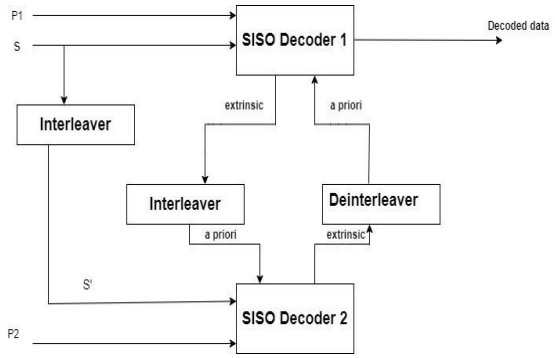


Fig. 3. Turbo Decoder Block diagram

B. SISO Decoder

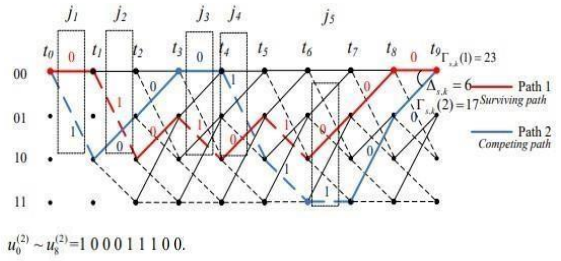
A Turbo decoder consists of two single soft-in soft-out (SISO) decoders that work iteratively. The output of the first (upper decoder) feeds into the second to form a Turbo decoding iteration. Turbo codes are decoded using a method called the Maximum Likelihood Detection or MLD. Filtered signal is fed to the decoders, and the decoders work on the signal amplitude to output a soft "decision" The a priori probabilities of the input symbols is used, and a soft output indicating the reliability of the decision is calculated which is then iterated between the two decoders. The form of MLD decoding used by turbo codes is called the Maximum a-posteriori Probability or MAP. However, ML decoder is often too complex to be implemented for turbo decoding because of the very complex trellis structure caused by the interleaver between the two constituent codes(CCs).In iterative decoding algorithm the two constituent decoders are used to perform SISO decoding over the coded sequences generated by the two CCs respectively, where the reliability information is exchanged between them during the decoding iterations.

The Soft output Viterbi Algorithm

The Soft-Output Viterbi Algorithm (SOVA) is a modified Viterbi Algorithm which can not only take in soft quantized samples but also provide soft decision outputs along with the hard decision. The SOVA can generate the hard decision in the same way as the Viterbi Algorithm. In addition, it can generate the reliability of the hard decision, which is named as the Soft-Output. The Soft-Output can be provided to its successor to improve the decoding performance of the concatenated decoder. At time k t, the ACSU in the Viterbi decoder selects a path with larger path metric between the two paths merging at the same states (It should be noticed that SOVA requires soft input. So the ACSU in the decoder should select the path with larger path metric). Without loss of generality, we assume path 1 is selected the survivor (that is, the path metric of path is larger). The probability of selecting the wrong path (path 2) as the

International Journal of Advanced Technology in Engineering and Science
Vol. No. 12, Issue No. 05, May 2024
www.ijates.com
ijates
ISSN 2348 - 7550

$$P_{s,k}=\frac{e^{-\Gamma_{s,k}(2)}}{e^{-\Gamma_{s,k}(1)}+e^{-\Gamma_{s,k}(2)}}=\frac{1}{1+e^{|\Delta_{s,k}|}}\leqslant\frac{1}{2},\ \Delta_{s,k}=|\Gamma_{s,k}(2)-\Gamma_{s,k}(1)|\geqslant 0$$



So, there are 5 positions where the information bit of path 1 differs from path 2, and the Viterbi decoder made errors in those 5 positions with probability

$$P_{0,9}=\frac{e^{-17}}{e^{-23}+e^{-17}}=\frac{1}{1+e^{6}}\approx 0.0025$$

SOVA is an algorithm derived from Viterbi algorithm family. Viterbi is an efficient hard decision algorithm for implementing trellis decoding. Since it only produces hard decisions, but the component decoders of turbo decoders should output soft values, Viterbi algorithm cannot be applied to SISO unit directly. SOVA uses a modified path metric which takes into account the a-priori probabilities of the input symbols, and produces a soft output indicating the reliability of the decision.

In Viterbi algorithm, only the survivor path metric is stored and the value of the loser is discarded. For this reason, information about the reliability of the selected path becomes unknown. Different from Viterbi, SOVA saves not only the survivor path metric, but also the path metric difference at each place where 2 paths merge. These path metric differences are later used to produce soft output. Larger the difference, stronger is the confidence value of the decision made for that transition. When reconstructing the most likely path, the hard decisions are used, as in Viterbi decoding, starting at the end of trellis (the solid line). For producing soft information, we use the differences between the most likely path and the most likely path when choosing the other path in first step (dashed line). By calculating the differences between the two paths until they converge, a measurement for the reliability of the decision is obtained.

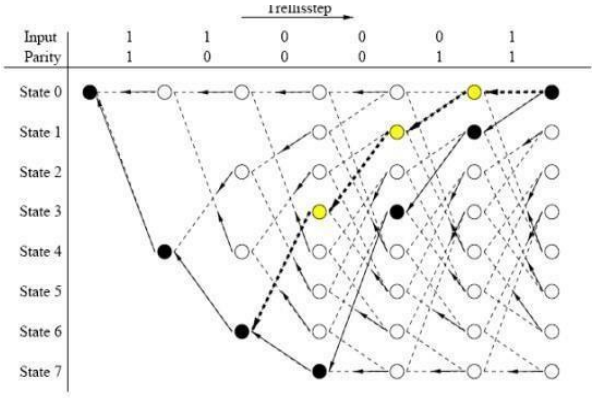


Fig 4: The decoding procedure of SOVA

C. Interleaver

For turbo codes, an interleaver is used between the two component encoders. The interleaver is used to provide randomness to the input sequences. Also, it is used to increase the weights of the code words systematic code

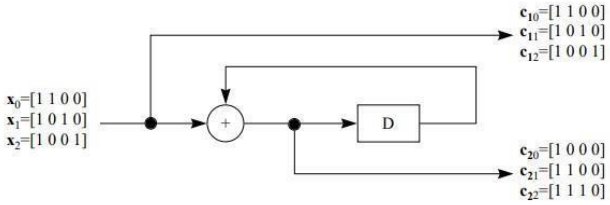


Fig 5: An illustrative example of an interleaver's capability

Types of Interleaver:

1.Matrix Interleaver: Matrix Interleaver is a type of block interleaver that performs interleaving by filling the input symbols row by row in a matrix of m rows and n columns and then output of the interleaver is read column

International Journal of Advanced Technology in Engineering and Science
Vol. No. 12, Issue No. 05, May 2024
www.ijates.com
ijates
ISSN 2348 - 7550

by column. The column size n is called the depth and the row size m is the span.

2. Random Interleaver: Random interleaver scrambles the data of different users with different pattern.Patterns of scrambling the data of users are generated arbitrarily. Because of the scrambling of data, burst error of the channel is randomized at the receiver side.

3.Convolutional Interleaver: A convolutional interleaver is an interleaver which consists of a number of shift registers. The shift registers have a fixed delay each, which are positive integer multiples of a fixed integer.

4.Pseudo-Random Interleaver: The random interleaver uses a fixed random permutation and maps the input sequence according to the permutation order. The length of the input sequence is assumed to be L.

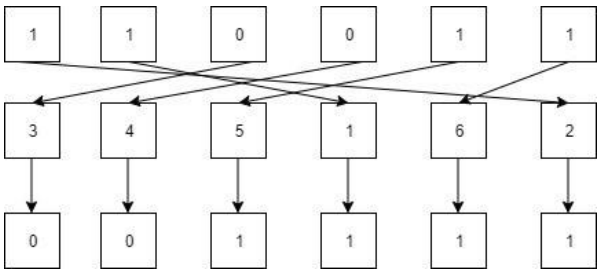


Fig. 4: Psuedo-random interleaver working principle

IV.RESULTS AND DISCUSSION

Turbo encoder and decoder simulations are performed with Verilog HDL in Xilinx Vivado 2018.3. Vivado Design Suite is a Xilinx-based software suite. It can be used for HDL design synthesis and analysis. Xilinx Vivado and Octave are used to simulate the outputs of recursive convolutional encoders and turbo encoder-decoders. During the synthesis process, Cadence NCLaunch generates the netlist from the RTL. NCLaunch is a GUI for managing large-scale design projects, as well as customizing and launching Cadence simulation software. The RTL compiler Ultra is used for logic synthesis and analysis in digital designs. The Encounter tool is used for physical design (floor layout, placement, and routing). The Cadence implementation tool takes a netlist as input and performs optimization, placement, and routing.

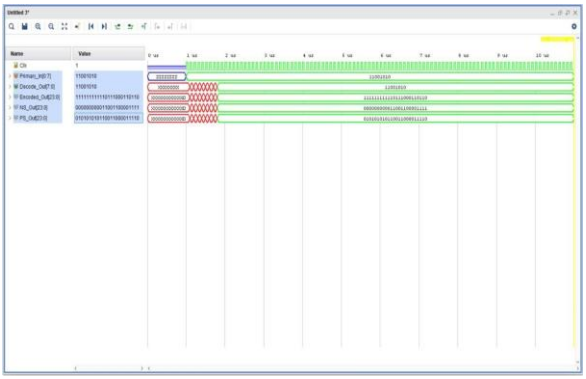


Fig 6: Simulation Result of Turbo coder

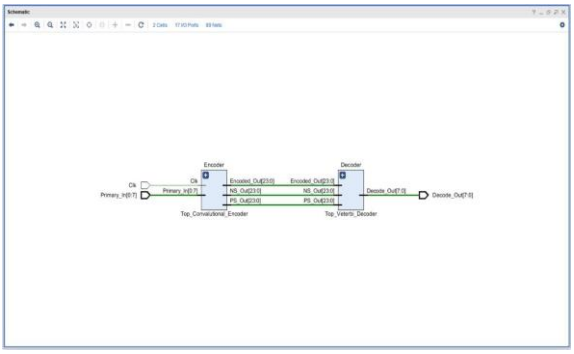


Fig 7: Schematic of Turbo Coder

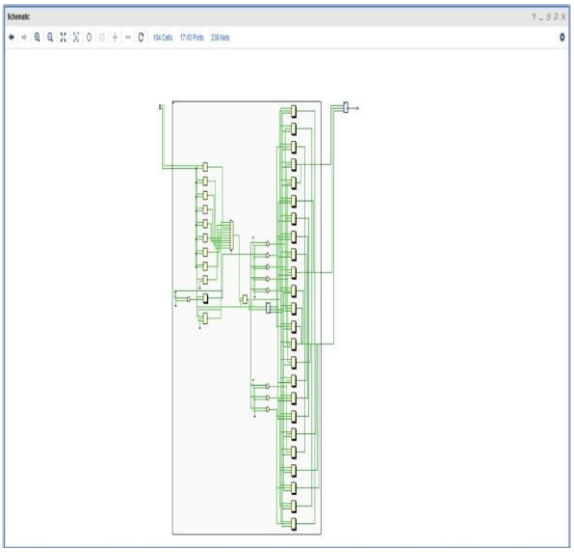


Fig 8: Schematic of Turbo Encoder

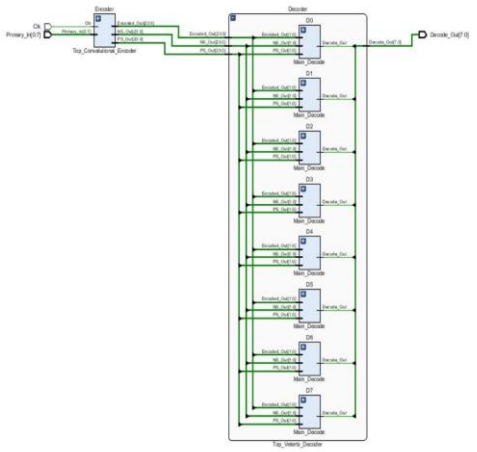


Fig 9: Schematic of Turbo Decoder

V. CONCLUSION

The Turbo encoder and Decoders are designed and simulated in Verilog HDL using Xilinx Vivado2020.2 version for 8-bit input. Input information bits are given to Turbo encoder that encodes data and then transmitted to Turbo Decoder through channel to obtain 24-bit output data. At decoder, the received data may contain errors, which are decoded using SOVA algorithm to obtain original information. Here, the decoder successfully corrects the error and retrieves the original message. Synthesis is done in Xilinx Vivado 2020.2 and results are tabulated.

REFERENCES

[1] V. Kavinilavu1, S. Salivahanan, V. S. Kanchana Bhaaskaran2 , Samiappa Sakthikumaran, B. Brindha and C. Vinoth "Implementation of Convolutional Encoder and Viterbi Decoder using Verilog HDL ", IEEE 3rd International Conference on Electronics Computer Technology,2011

[2] Aswathy Narayanan ; Senthil Murugan ; Ramesh Bhakthavatchalu, "Low Latency Max Log MAP based Turbo Decoder", International Conference on Communication and Signal Processing (ICCSP),2019

[3] C. Tanriover ; B. Honary ; Jun Xu ; Shu Lin", "Improving turbo code error performance by multifold coding",IEEE Communications Letters, 2002

[4] Shweta Ramteke ; Sandeep Kakde ; Yogesh Suryawanshi , "HDL Implementation of Turbo Decoder Using Soft Output Viterbi Algorithm",2015.

[5] J.H. Han ; A.T. Erdogan ; T. Arslan ,"Power and area efficient turbo decoder implementation for mobile wireless systems", IEEE Workshop on Signal Processing Systems Design and Implementation,2005

[6] Cheng-Chi Wong ; Ming-Wei Lai ; Chien-Ching Lin ; Hsie-Chia Chang ; Chen-Yi Lee,"Turbo Decoder Using Contention-Free Interleaver and Parallel Architecture",IEEE Journal of Solid-State Circuits",2010

[7] Yaqin Shi ; Ming Zhan ; Jie Zeng , "FPGA Implementation and Power Estimation of a Memory-Reduced LTE-Advanced Turbo Decoder", IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData), 2018.

[8] E. Boutillon, C. Douillard, and G. Montorsi , "Iterative decoding of concatenated convolutional codes: Implementation issues",Proc. IEEE, vol. 95, no. 6, pp. 1201–1227,June, 2007

[9] Rami Klaimi ; Charbel Abdel Nour ; Catherine Douillard ; Joumana Farah ,"Low-complexity decoders for non-binary turbo codes" ,IEEE 10th International Symposium on Turbo Codes Iterative Information Processing (ISTC), 2018

[10] Maria Kovaci ; Horia Balta,"Performance of trellis termination methods for RSC component encoders of turbo codes", 10th International Symposium on Electronics and Telecommunications,IEEE,2012

[11] L. Dinoi and S. Benedetto,"Variable-size interleaver design for parallel turbo decoder architectures",IEEE Global Telecommunications Conf., Nov. 2004, pp. 3108–3112 ,2004

[12] Neha.S, Pargunarajan.K "Hybrid sub-optimal decoding technique for LDPC codes",Proceedings of IEEE International Conference on Circuit, Power and Computing Technologies, ICCPCT 2016 ,2 August 2016, Article number 7530160.

[13] Priyadarshini, N. Pargunarajan, K., "A Constrained Search Algorithm for Selection of Optimal Generators in Convolutional Encoder," International Journal of Computer Science Engineering and Technology (IJCSET), July 2011,Vol 1, Issue 6, 324-326.