IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 1, September 2026
Impact Factor: 8.2

FPGA-Based VLSI Implementation of A Map Turbo Decoder for Reliable Space Communication Systems

Dr. Kumar Keshamoni[1], T. Swathi[2], Bathula Manoj Kumar[3]
Associate Professor, Dept. of ECE[1]
Assistant Professor, Dept. of ECE[2]
Dept. of ECE[3]
Brilliant Grammar School Educational Society's Group of Institutions - Integrated Campus, Hyderabad, India.
kumar.keshamoni@gmail.com, Swathi.tulam888@gmail.com, bathulamanojkumar13@gmail.com

Abstract: *Reliable error control is essential in space and long-distance digital communication because transmitted data may be corrupted by channel noise, fading, interference, and other impairments. This paper presents an FPGA-oriented Very Large Scale Integration (VLSI) implementation of a Turbo coding system using two recursive systematic convolutional (RSC) encoders, a pseudorandom interleaver, and an iterative soft-input soft-output (SISO) decoder based on the Maximum-a-Posteriori (MAP) algorithm. The proposed architecture targets efficient hardware realization in Verilog and is evaluated using the Xilinx Vivado design environment. The decoder iteratively exchanges reliability information between two SISO decoding stages through interleaving and deinterleaving, improving the estimate of the transmitted information sequence. The implementation results reported in the source design show a resource use of 12 LUTs and 48 I/O pins, an on-chip power estimate of 12.638 W, and a representative timing delay of approximately 2.30 ns. Compared with the Hamming-code baseline reported in the same project, the MAP-based Turbo design reduces LUT count from 50 to 12, power from 15.66 W to 12.638 W, and delay from 3.50 ns to 2.30 ns. These values correspond to approximately 76% lower LUT usage, 19.3% lower reported power, and 34.3% lower delay. The results support the use of iterative Turbo decoding as a hardware-efficient error-control approach for high-reliability communication links, including space communication applications.*

Keywords: Turbo code, MAP algorithm, SISO decoder, FPGA, VLSI, space communication, error correction, interleaver, Verilog, Vivado

I. INTRODUCTION

Digital communication systems increasingly carry large volumes of information through channels that may introduce random and burst errors. In terrestrial wireless links these errors arise from noise, interference, multipath effects, and fading; in long-distance and space communication, reliability is further challenged by weak received signal levels and stringent power and hardware constraints. Error-control coding is therefore a fundamental part of the physical layer because it allows a receiver to detect or correct corrupted information without requesting repeated transmission of every erroneous block.

Classical error-control techniques such as Hamming coding provide low-complexity protection and are attractive when the channel error rate is small. Their correction capability, however, is limited. Turbo codes were introduced as powerful iterative codes whose decoding performance can approach the theoretical limits of reliable communication. The main strength of Turbo decoding is the repeated exchange of soft reliability information between component decoders rather than making a hard binary decision after only one decoding pass.

DOI: 10.48175/IJARSCT-38269

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 1, September 2026 Impact Factor: 8.2

The Turbo architecture considered in this work uses two recursive systematic convolutional encoders separated by a pseudorandom interleaver. At the receiver, two soft-input soft-output decoders iteratively process the received systematic and parity information. The component decoder applies a Maximum-a-Posteriori (MAP) procedure, which estimates the probability of each transmitted data bit by considering the complete trellis observations. An interleaver and complementary deinterleaver rearrange the reliability information so that both component decoders operate on the same information sequence in different orders.

The computational advantage of a hardware implementation is important because iterative decoding contains repeated forward-state, backward-state, branch-metric, and reliability calculations. A software-only implementation can introduce significant latency, whereas an FPGA can execute many of these operations concurrently and can be reconfigured as algorithmic requirements evolve. The present design is described in Verilog and synthesized using the Xilinx Vivado tool chain.

This paper restructures the supplied project into a publication-style study focused on a Turbo decoder suitable for reliable space communication. The paper evaluates the proposed MAP-based Turbo architecture against the Hamming-code baseline reported in the project and analyzes its functional organization, hardware implementation, resource utilization, power estimate, and timing results.

The main contributions of the work are:

• A hardware-oriented Turbo coding architecture using two RSC encoders and a pseudorandom interleaver on the transmitter side.

• An iterative Turbo decoder containing two SISO component decoders that employ MAP processing and exchange extrinsic information through interleaving and deinterleaving.

• A Verilog/Vivado FPGA implementation flow with simulation, synthesis, power estimation, and timing analysis.

• A direct comparison with the Hamming-code baseline using the project-reported LUT, power, and delay values.

• An application perspective highlighting the suitability of strong iterative error control for space, satellite, broadband, and other high-reliability communication systems.

II. RELATED WORK

Akshaya et al. implemented a Turbo coder for LTE using Verilog HDL and described a decoder based on SISO processing, interleaving, deinterleaving, and MAP decoding [1]. Zhan et al. proposed a reverse-calculation low-memory Turbo decoder in which trellis partitioning and state-metric processing reduced state-metric cache capacity while preserving decoding performance [2]. These studies establish memory organization and MAP processing as important factors in practical Turbo decoder design.

Kumar and Bharath studied FPGA realization of a Turbo encoder for an in-vehicle system and reported that parallel computation improved both processing time and hardware utilization [3]. Weithoffer et al. examined fully pipelined iteration-unrolled Turbo decoders for extremely high-throughput operation and showed how spatial parallelization can trade silicon area against throughput [4]. Paidmalla and Prasanna likewise investigated an area-efficient pipelined Turbo encoder/decoder and emphasized delay and area optimization [5].

Energy is an equally important design objective. Rangachari and Chandrachoodan investigated dynamically varying bit widths in Turbo decoding to reduce energy per decoding operation [6]. Ali and Alneema used high-level synthesis to study a Turbo decoder based on a single MAP decoder with different parallelism and windowing options [7]. Shrimali and Sharma employed an efficient Log-MAP Turbo encoder/decoder within a MIMO-OFDM physical layer, demonstrating the importance of balancing error-correction performance with implementation complexity [8].

Recent work also addresses latency and throughput through algorithmic reformulation. Le Gal and Jego reported low-latency, high-throughput software Turbo decoding on multi-core processors [10]. Dheeb and Sabbar implemented LTE Turbo decoding with Log-MAP and serial-to-parallel processing on multiple FPGA families [11]. Salija et al. proposed a reliability-based decoding method for short Turbo codewords, while Le et al. developed a local SOVA formulation that reduces complexity for high-radix SISO decoding [14], [15].

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 1, September 2026 Impact Factor: 8.2

For high-performance hardware, Chennapalli and Nanjappa proposed a parallel Turbo decoder with a three-stage micro-pipeline and a simplified QPP interleaver, reporting throughput improvement and power reduction relative to recent architectures [19]. Beeharry studied early termination for fully parallel LTE Turbo decoding to reduce average iteration count [20]. These approaches illustrate that decoder efficiency depends not only on the MAP/SISO arithmetic but also on iteration scheduling, interleaver implementation, memory organization, and stopping criteria.

Space communication motivates strong error-control coding because retransmission may be costly and communication links can experience large path loss. Poongodi and Rani investigated multi-bit error detection and correction codes for space communication [23], while other hardware studies in the source literature considered FPGA-based encoders, decoders, and error-control architectures for wireless and storage applications [18], [24], [26]. The present work specifically evaluates a MAP-based Turbo architecture against a Hamming-code baseline using the synthesis and timing data available in the supplied design.

III. ERROR-CONTROL BACKGROUND AND BASELINE

3.1 Hamming Code Baseline

Hamming codes are linear block codes that add parity bits at positions that are powers of two. The parity pattern enables the decoder to identify the location of a single corrupted bit. For m information bits and r redundant bits, the number of parity bits must be sufficient to represent every possible single-error position and the no-error condition.

The source project uses Hamming coding as the existing method for comparison. The basic condition for selecting the number of redundant bits is:

$$2^r \geq m + r + 1$$

At the receiver, parity checks form a syndrome. A zero syndrome indicates that no single-bit error has been detected, while a nonzero syndrome identifies a bit position that can be inverted for correction. An extended Hamming code may add an overall parity bit to distinguish a single correctable error from a double-bit error. Although this mechanism is simple, it does not provide the iterative soft-decision gain of Turbo decoding.

3.2 Need for Turbo Coding in Reliable Links

High-reliability links benefit from decoders that exploit probabilistic information rather than discarding it after demodulation. A soft decoder uses the relative confidence of received symbols to improve the decision process. Turbo decoding applies this concept iteratively: one constituent decoder generates extrinsic reliability information, that information is rearranged by the interleaver, and the second constituent decoder uses it as prior knowledge in the next stage.

This process is particularly attractive for space and satellite links because coding gain can reduce the probability of erroneous information delivery without requiring the transmitter to repeat every corrupted block. The trade-off is increased arithmetic and memory complexity, making FPGA-based VLSI implementation an appropriate approach for achieving practical throughput.

IV. PROPOSED MAP-BASED TURBO CODING ARCHITECTURE

The proposed system combines a Turbo encoder, communication channel, and iterative Turbo decoder. The transmitter generates systematic and parity information, and the receiver applies MAP-based SISO processing to recover the original data. The overall organization from the source design is shown in Fig. 1.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 1, September 2026
Impact Factor: 8.2

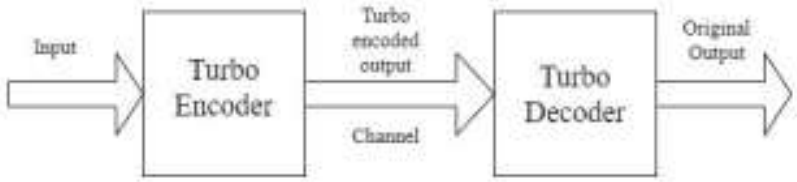

Fig. 1. Overall Turbo encoder-channel-decoder organization used in the proposed system.

4.1 Turbo Encoder

The encoder is a parallel concatenated structure formed by two identical recursive systematic convolutional encoders and a pseudorandom interleaver. The original information sequence is applied directly to the first RSC encoder. The same sequence is rearranged by the interleaver before being applied to the second RSC encoder. The source design describes an LTE-style rate-1/3 Turbo coding structure in which the systematic information and two parity sequences form the transmitted codeword.

The interleaver decorrelates neighboring information bits before the second convolutional encoding process. Consequently, an error event that is difficult for one constituent code may appear in a more favorable position for the other constituent decoder. This diversity in bit ordering is fundamental to the iterative gain of Turbo coding.

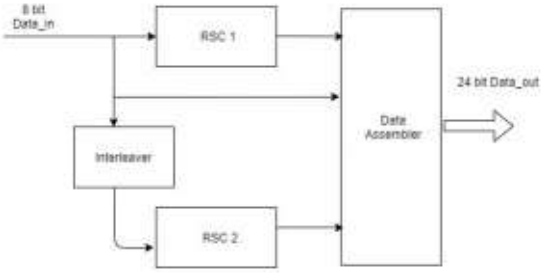

Fig. 2. Turbo encoder with two RSC encoders, interleaver, and data assembler.

4.2 Iterative Turbo Decoder

The Turbo decoder contains two SISO component decoders, two interleaving operations, and one deinterleaving operation. A Turbo iteration consists of one pass through the first component decoder followed by one pass through the second component decoder. During each pass, the decoder uses the received channel observations and reliability information generated by the companion decoder.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 1, September 2026
Impact Factor: 8.2

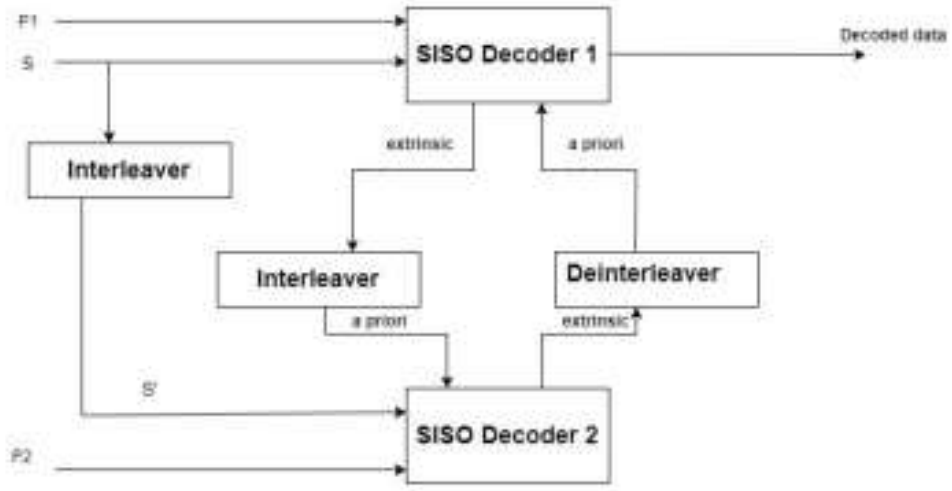

Fig. 3. Iterative Turbo decoder with two SISO decoders, interleavers, and deinterleaver.

The first SISO decoder processes the systematic sequence and the parity sequence associated with the first RSC encoder. It outputs a refined reliability estimate and an extrinsic component. The extrinsic information is interleaved before becoming a priori information for the second SISO decoder. The second decoder operates on the interleaved systematic information and the parity sequence of the second RSC encoder. Its extrinsic output is deinterleaved and returned to the first decoder for the next iteration.

This exchange continues for the selected number of iterations. As the extrinsic information improves, the reliability of the estimated information bits typically becomes stronger. The final hard decision is then obtained from the refined soft information.

4.3 MAP-Based SISO Processing

The source implementation uses the Maximum-a-Posteriori algorithm as the SISO decoding method. Unlike a decoder that selects only the single most likely path through the trellis, the MAP procedure estimates the posterior probability of each data symbol by considering the probability contributions of all relevant trellis transitions.

For a received sequence y and information symbol d_i, the decision is based on the posterior probability P(d_i = j | y). The source document expresses this probability using joint probabilities and decomposes the trellis calculation into forward, backward, and branch metrics.

$$P(d\_i = \frac{j}{y}) = \frac{P(d\_i = j, y)}{\Sigma\_k\, P(d\_i = k, y)}$$

$$P(d\_i = j \mid y) \propto \sum_{(S\_i, S\_(i+1))} \alpha\_i(S\_i) \cdot \gamma\_i(S\_i, S\_(i+1)) \cdot \beta\_(i+1)(S\_(i+1))$$

$$\gamma\_i(S\_i, S\_(i+1)) = p(y\_i \mid x\_i) \cdot P\_a(d\_i)$$

Here, α represents the forward state metric, β represents the backward state metric, and γ represents the branch metric. The branch metric combines the channel transition probability with the a priori information supplied to the SISO decoder. This decomposition maps naturally to hardware modules for state-metric recursion, branch-metric computation, and reliability generation.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 1, September 2026
Impact Factor: 8.2

4.4 Interleaver and Deinterleaver

The interleaver rearranges the information sequence according to a predefined pseudorandom pattern. Its purpose is not to change the information values but to change their positions so that the second constituent code observes a different neighborhood structure. On the decoder side, the same mapping is used to align the extrinsic information with the second SISO decoder, while the deinterleaver applies the complementary mapping before information is returned to the first decoder.

The interleaving concept used in the source design is illustrated in Fig. 4. A deterministic hardware implementation can realize the mapping using address-generation logic or a stored permutation table, depending on the selected frame size and resource constraints.

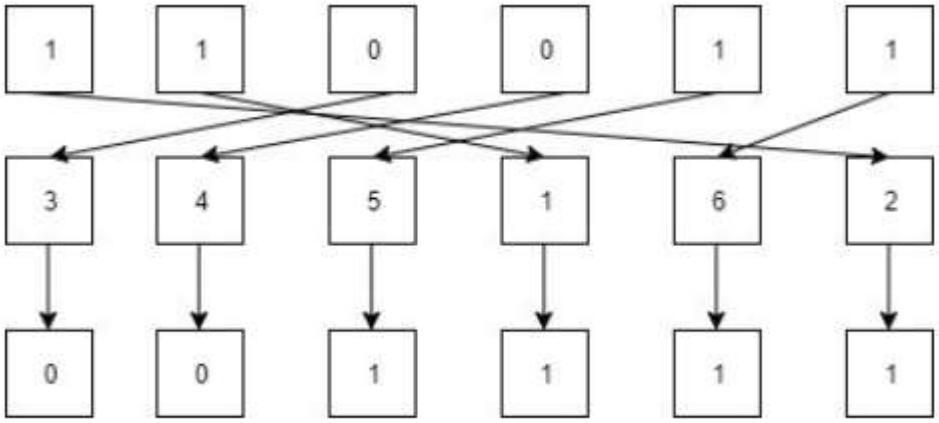

Fig. 4. Pseudorandom interleaving operation used to permute the information-bit order.

V. FPGA IMPLEMENTATION METHODOLOGY

The hardware design is implemented using Verilog HDL and evaluated in Xilinx Vivado. The RTL description defines the Turbo encoder, interleaving logic, decoder datapath, and associated control. Vivado is then used for behavioral simulation, synthesis, implementation, resource analysis, timing analysis, and power estimation.

The design flow begins with functional verification of the RTL. Known input patterns are applied to the simulation model and the encoded output signals are observed in the waveform viewer. After functional correctness is established, synthesis converts the RTL into technology-specific logic. Implementation maps, places, and routes the resulting netlist on the target FPGA. The post-implementation reports provide LUT utilization, I/O utilization, timing path delay, and estimated on-chip power.

This flow is suitable for communication-oriented VLSI design because it allows algorithmic changes to be evaluated without fabricating a new ASIC. In addition, the FPGA can serve as a prototype for later migration to an application-specific implementation if higher volume or lower power is required.

VI. RESULTS AND DISCUSSION

6.1 Functional Simulation

The simulated waveform reported in the project is reproduced in Fig. 5. The waveform demonstrates clocked operation and the generation of encoded output signals for the applied input data. The source material presents the simulation as functional evidence that the Verilog implementation produces the expected Turbo-coded output sequence.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 1, September 2026 Impact Factor: 8.2

Fig. 5. Vivado simulation waveform of the implemented Turbo coding design.

6.2 Resource Utilization

The Vivado utilization report shows that the proposed design uses 12 LUTs out of 101,400 available LUTs and 48 I/O resources out of 285 available I/O resources. The corresponding utilization values shown in the report are 0.01% for LUTs and 16.84% for I/O. The very small LUT count reflects the compact logic reported for the implemented configuration; the comparatively larger I/O percentage is associated with the external input and output ports of the design.

Graph | **Table**

Resource	Utilization	Available	Utilization %
LUT	12	101400	0.01
IO	48	285	16.84

Fig. 6. FPGA resource utilization reported by Vivado: 12 LUTs and 48 I/O resources.

6.3 Power Analysis

The Vivado power report in Fig. 7 gives a total on-chip power estimate of 12.638 W for the implemented design. The report shows approximately 12.405 W dynamic power and 0.233 W device-static power. Within the displayed dynamic breakdown, I/O power is the dominant component at approximately 12.144 W, while signals and logic account for smaller contributions. These values are reported exactly as available in the supplied project implementation.

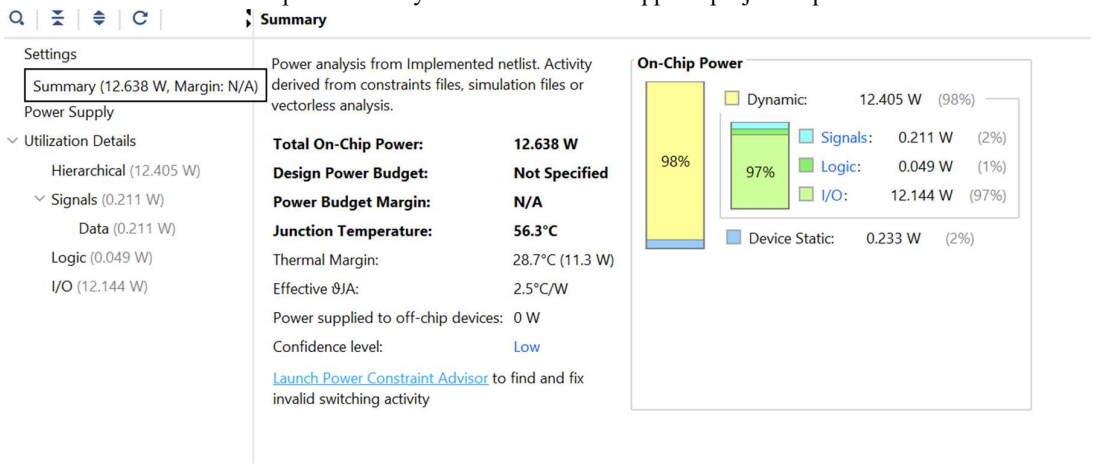

Fig. 7. Vivado on-chip power summary for the implemented Turbo coding architecture.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 1, September 2026
Impact Factor: 8.2

6.4 Timing Analysis

The timing report lists representative path delays between approximately 2.173 ns and 2.327 ns. The project performance table summarizes the proposed MAP-based Turbo design with a delay of 2.30 ns. The detailed timing view is shown in Fig. 8; the small difference between individual path values and the table value is consistent with rounding of the reported result.

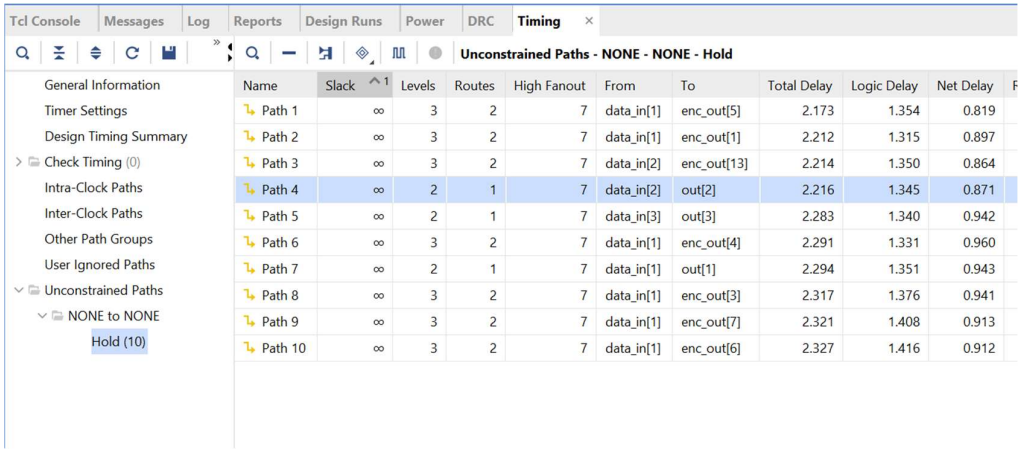

Fig. 8. Vivado timing summary showing representative combinational path delays.

6.5 Comparison with Hamming-Code Baseline

The source project directly compares the Hamming-code existing method with the MAP-based proposed design using area, power, and delay. Table 1 reproduces these values and adds the relative reduction calculated from the reported measurements.

Table 1. Performance comparison reported in the source project.

Parameter	Hamming Code	MAP-Based Turbo Design	Relative Reduction
Area (LUTs)	50	12	76.0%
Power (W)	15.66	12.638	19.3%
Delay (ns)	3.50	2.30	34.3%

The largest improvement is observed in LUT utilization. Reducing the reported count from 50 LUTs to 12 LUTs corresponds to a 76% reduction relative to the Hamming-code baseline. In an FPGA implementation, lower LUT demand can leave more resources available for other communication functions, buffering, interfaces, or parallel decoder instances.

The reported power decreases from 15.66 W to 12.638 W, corresponding to an improvement of approximately 19.3%. Power reduction is important in space and embedded communication platforms because electrical energy, thermal management, and available cooling capacity are limited design resources.

The delay decreases from 3.50 ns to 2.30 ns, a reduction of approximately 34.3%. Lower combinational delay can support a higher operating frequency or provide additional timing margin for a larger communication subsystem. The timing screenshot additionally shows paths up to about 2.327 ns, which is consistent with the rounded 2.30 ns value used in the source comparison table.

The comparison should be interpreted within the scope of the supplied implementation. Hamming and Turbo codes provide different levels of coding capability and decoder complexity, so the table is a hardware comparison of the specific project implementations rather than a universal ranking of all Hamming and Turbo decoder architectures. No BER-versus-SNR curve is provided in the source document, and therefore no unsupported error-rate values are introduced here.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 1, September 2026
Impact Factor: 8.2

VII. RELEVANCE TO SPACE COMMUNICATION

Space communication systems require high reliability because received signals may be weak and retransmission may incur substantial delay. The iterative soft-decision nature of Turbo decoding is therefore attractive for telemetry, command links, scientific payload data, and satellite communication where protection against channel errors is essential. The FPGA implementation also provides a useful prototyping platform for space-oriented digital communication subsystems. Logic can be reconfigured to change frame size, interleaver mapping, iteration count, or interface behavior. This flexibility is valuable during system development and testing, although a flight-qualified implementation would additionally require device-level radiation tolerance, fault mitigation, and mission-specific verification that are outside the scope of the supplied project.

The same architecture can also be applied to terrestrial wireless communication, broadband links, Wi-Fi-like systems, mobile communication, and other high-speed digital channels described in the source study. The underlying benefit is the same: stronger iterative error correction with a hardware implementation capable of low-latency operation.

VIII. LIMITATIONS AND FUTURE SCOPE

The present implementation demonstrates functional simulation and hardware synthesis, but the source project does not provide a complete BER-versus-Eb/N0 characterization. A stronger publication study should evaluate error-rate performance across multiple channel conditions and compare the number of Turbo iterations required for convergence.

Future work may incorporate adaptive iteration control or early termination so that decoding stops when sufficient reliability has been reached. This can reduce average energy consumption without forcing every frame to execute the maximum number of iterations. Dynamically varying internal bit widths, as discussed in related work, is another potential power-saving technique.

The interleaver can be optimized for larger frame sizes and for deterministic hardware address generation. Parallel or windowed MAP architectures can also be studied to increase throughput while controlling memory use. For space deployment, error-control hardware should additionally be combined with radiation-aware FPGA techniques, redundancy, scrubbing, and fault-tolerant memory protection.

The source project also suggests future investigation of adaptive Turbo decoding, hybrid processor/FPGA implementations, and non-binary codes. These directions can extend the decoder to different link conditions and application requirements while maintaining the reconfigurability of the FPGA platform.

IX. CONCLUSION

This paper presented an FPGA-based VLSI implementation of a MAP Turbo decoder for reliable digital and space communication systems. The encoder uses two recursive systematic convolutional encoders and a pseudorandom interleaver, while the receiver uses two iterative SISO decoders with MAP processing, interleaving, and deinterleaving. The architecture exploits soft reliability information and repeated decoding passes to provide stronger error-control capability than a simple single-error-correcting baseline.

The supplied Vivado results report 12 LUTs, 48 I/O resources, 12.638 W total on-chip power, and a representative delay of approximately 2.30 ns. Relative to the Hamming-code implementation reported in the same project, the proposed MAP-based design reduces LUT count by 76%, reported power by approximately 19.3%, and delay by approximately 34.3%. Functional simulation further confirms the expected encoded data behavior of the Verilog design.

The study demonstrates the practical value of mapping iterative channel decoding to reconfigurable hardware. With further work on BER characterization, iteration control, memory organization, radiation tolerance, and parallel MAP processing, the architecture can be developed toward more demanding high-reliability communication applications.

REFERENCES

[1] V. Akshaya, K. N. Sreehari, and A. Chalil, “VLSI Implementation of Turbo Coder for LTE using Verilog HDL,” in Proc. 4th Int. Conf. Computing Methodologies and Communication (ICCMC), IEEE, 2020.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 1, September 2026 Impact Factor: 8.2

[2] M. Zhan et al., "Reverse calculation-based low memory turbo decoder for power constrained applications," IEEE Trans. Circuits Syst. I: Regular Papers, vol. 68, no. 6, pp. 2688–2701, 2021.
[3] N. S. V. Kumar and G. L. Bharath, "Designing of IVS chip encoder module in VLSI Architecture," 2020.
[4] S. Weithoffer et al., "Fully pipelined iteration unrolled decoders: The road to TB/s turbo decoding," in Proc. ICASSP, IEEE, 2020.
[5] N. Paidmalla and T. L. Prasanna, "Design and evaluation of area-efficient pipelined turbo encoder and decoder," Turkish Journal of Computer and Mathematics Education, vol. 12, no. 11, pp. 7252–7258, 2021.
[6] Mahesh, M. Javeed, and D. Tiwary, "Internet of Things for Health Care Mechanism," *International Journal of Research in Engineering, Science and Management*, vol. 1, no. 9, pp. 1–6, Sep. 2018. Google Scholar
[7] A. T. Ali and D. A. F. Alneema, "Design Analysis of Turbo Decoder Based on One MAP Decoder Using High Level Synthesis Tool," Al-Rafidain Engineering Journal, vol. 25, no. 1, pp. 70–77, 2020.
[8] Y. Shrimali and J. B. Sharma, "Efficient HDL Implementation of Turbo Coded MIMO-OFDM Physical Layer," in Nanoelectronics, Circuits and Communication Systems, Springer, 2021.
[9] A. Verma and R. Shrestha, "A new partially-parallel VLSI-architecture of quasi-cyclic LDPC decoder for 5G new-radio," in Proc. VLSID, IEEE, 2020.
[10] B. Le Gal and C. Jego, "Low-latency and high-throughput software turbo decoders on multi-core architectures," Annals of Telecommunications, vol. 75, pp. 27–42, 2020.
[11] K. O. Dheeb and B. Sabbar, "Different FPGA products based implementation of LTE turbo code," Iraqi Journal of Information and Communication Technology, vol. 3, no. 1, pp. 40–51, 2020.
[12] A. Boudaoud, M. El Haroussi, and E. Abdelmounim, "Turbo decoder based on DSC codes for multiple-antenna systems," in Advanced Communication Systems and Information Security, Springer, 2020.
[13] S. Joseph, R. Kirubakkar, and M. Mariammal, "Design and Implementation of Turbo Coder for 5G Technology," in Proc. ICACCS, IEEE, 2021.
[14] P. Salija et al., "A novel reliability-based high performance decoding algorithm for short block length turbo codes," International Journal of Ad Hoc and Ubiquitous Computing, vol. 33, no. 3, pp. 155–167, 2020.
[15] V. H. S. Le et al., "Revisiting the max-log-map algorithm with SOVA update rules: New simplifications for high-radix SISO decoders," IEEE Trans. Communications, vol. 68, no. 4, pp. 1991–2004, 2020.
[16] M. O. Matar et al., "A turbo maximum-a-posteriori equalizer for faster-than-Nyquist applications," in Proc. IEEE FCCM, 2020.
[17] A. Bourenane et al., "Shuffled Decoding of Serial Concatenated Convolutional Codes," in Proc. 11th Int. Symp. Topics in Coding, IEEE, 2021.
[18] Md. Javeed, "Design of Shift Register Based on Multi Bit Flip Flop for Universal Asynchronous Receiver and Transmitter," International Journal of Engineering Research in Electronics and Communication Engineering (IJERECE), vol. 1, no. 7, pp. 42–45, Aug. 2015. Google Scholar
[19] S. Chennapalli and G. P. M. Nanjappa, "A New VLSI Architecture for High-Performance Parallel Turbo Decoder," IIUM Engineering Journal, vol. 23, no. 2, pp. 125–137, 2022.
[20] Y. Beeharry, "A Novel Hybrid SDR-Regression based Early Termination Processes for the Fully Parallel Turbo Decoder for LTE," 2022.
[21] F. Z. Zenkouar, M. El Alaou, and S. Najah, "GF(q) LDPC encoder and decoder FPGA implementation using group shuffled belief propagation algorithm," International Journal of Electrical & Computer Engineering, vol. 12, no. 3, 2022.
[22] R. K. Dhanavath, "An Efficient Technique to Implement Encoder and Decoder in Communication System for Error Detection and Correction Using Golay Code," 2022.
[23] S. Poongodi and A. S. Rani, "VLSI Implementation of Multi-Bit Error Detection and Correction Codes for Space Communications," Ratio Mathematica, vol. 44, p. 107, 2022.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 1, September 2026
Impact Factor: 8.2

[24] Ch. Sundeep Kumar, S. Vinay Kumar, and M. Javeed, "An Efficient Scheme for Inter Carrier Interference Cancellation in High Speed OFDM System," International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering, vol. 6, no. 11, pp. 8638–8645, Nov. 2017. Google Scholar
[25] Z. A. Humadi and Q. F. Al-Doori, "New FPGA architecture for 8, 16 and 24 bit chaotic interleaver," in Proc. IICETA, IEEE, 2022.
[26] S. Radhakrishnan, S. I. Ahmed, and S. R. Ramesh, "Implementation of Classical Error Control Codes for Memory Storage Systems Using VERILOG," in Intelligent Sustainable Systems, Springer, 2022, pp. 29–41.
[27] N. A. Kumar et al., "Routing Strategy: Network-on-Chip Architectures," in VLSI Architecture for Signal, Speech, and Image Processing, Apple Academic Press, 2022.
[28] Md. Javeed and G. Kande, "A Survey on Linking of Cell Tracks Using Segmentation," International Journal of Applied Engineering Research, vol. 13, no. 7, pp. 5183–5189, 2018. Google Scholar
[29] C.-H. Lin et al., "Reconfigurable Low-Density Parity-Check (LDPC) Decoder for Multi-Standard 60 GHz Wireless Local Area Networks," Electronics, vol. 11, no. 5, p. 733, 2022.
[30] S. A. Hussain et al., "A hybrid soft bit flipping decoder algorithm for effective signal transmission and reception," TELKOMNIKA, vol. 20, no. 3, pp. 510–518, 2022.
[31] B. R. Kadhim and F. Korkmaz, "Reducing LDPC decoder complexity by using FPGA based on min sum algorithm," Kufa Journal of Engineering, vol. 13, no. 1, pp. 82–101, 2022.