

LDPC Architecture for Improved BER in Wireless Networks

Maria Rubiston. M

Department of Electronics and Communication Engineering,
Sathyabama University, Chennai

ABSTRACT

To achieve high throughput in wireless networks a partial parallel LDPC decoder is proposed in this paper. For fully-parallel decoders, it suffers from large hardware complexity caused by a large set of processing units and complex interconnections. In wireless networks coding complexity and routing congestion can be reduced by designing the decoder with partially-parallel architecture. The partially-parallel architecture with Split Row algorithm reduces the total global wire length by about 26% without any hardware overhead and increasing the throughput by 60% and 71% in wireless networks.

Keywords

LDPC-Low Density Parity Check Decoder, VN-Variable Node, PU-Processing Unit, WNs-Wireless Networks, CHNU-Check Node Unit, CNU-Control Node Unit, MU-Memory Unit, AGU-Automatic Gain Unit, SNR-Signal to Noise Ratio, SP-Split, Col-Column, Mem-Memory, BER-Bit Error Rate.

1. INTRODUCTION

The partially parallel architecture is good trade-off between throughput and hardware cost. Since a PU is shared for a number of rows or columns, the number of PUs becomes much smaller than that of the fully-parallel architecture. As decoding operations are parallel in nature, it is important to determine which rows or columns are processed in PU. In the grouping, the dependencies between rows and columns should be considered to minimize the overall cycles by overlapping the decoding operations^[2].

A key concern in the design of high throughput LDPC code decoders comes from the communication structure that must be allocated to support message passing among VNs and CNs. Three approaches can be followed in the high level organization of the decoder:

The recently extended version^[13] of Split-Row decoding method^[14] for irregular LDPC codes decreases interconnect complexity by splitting the rows of the parity check matrix into nearly independent halves and provides a reduced complexity and smaller interconnect complexity.

This paper introduces a Split-Row threshold method to further enhance the throughput and energy efficiency for irregular LDPC codes as presented in reference [15] for regular LDPC codes. This method reduces the wire interconnect complexity

between row and column processors and increases parallelism in the row processing stage. The Split-Row method also simplifies row processors which results in an overall smaller decoder.

1. Fully Parallel Architectures (FPA): separate processing units are allocated for each VN and CN and all messages are passed in parallel along dedicated routes.
2. Partially Parallel Architectures (PPA): more processing units work in parallel, serving all VNs and CNs within a number of cycles; suitable organization and hardware support are required to exchange messages.
3. Serial architectures (SA): a single processing instance is allocated for both VN and CN computations and nodes is served sequentially; messages are exchanged by means of a unique memory.

The first approach leads to very high throughput, large implementation cost and severe congestion problems in the routing of interconnects^[3]. For these reasons it is not adopted in practical implementations. The partially parallel architecture requires a large bandwidth between processing units and memories where messages are stored. Moreover, special attention is necessary to avoid collisions in the memory access^[4]. However, the partially parallel organization allows to precisely tune the wanted degree of parallelism with respect to the addressed throughput and it was proved to be the best solution for the implementation of efficient decoders^[4-8].

The serial approach leads to low cost and low power implementations and it also offers a high level of flexibility with respect to the supported code. However serial architectures did not receive much attention, due to the fact that the sequential processing does not achieve large throughput. This solution is particularly suitable for software implementations on Digital Signal Processors^[9]. As throughput requirements in WNs applications are usually much lower than in wireless communications, the serial approach appears as the best solution to implement low cost and low energy decoding in a sensor node.

This Paper proposes high performance LDPC for wireless networks using partially parallel decoder architecture with split row algorithm and comparison of simulation results are discussed in Section (6), which shows the bit error rate performance of different architectures. Output comparison of both existing and proposed decoder architecture is tabulated in Section (7).

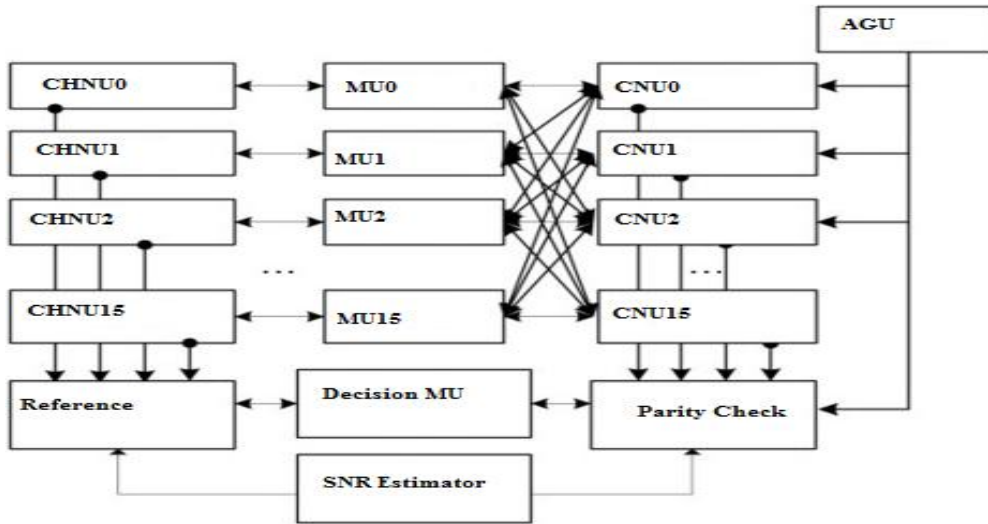


Fig 1. Block Diagram of Partial Decoder

2. PARTIAL PARALLEL DECODER

The main idea behind partial-parallel implementations is to build a balance between parallelism in decoding and the interconnection wiring complexity. In this architecture, a subset of variable nodes and check nodes is implemented in the hardware, and by changing the routing network between implemented nodes, different partitions of parity check matrix are processed. Since intermediate messages need to be stored, memory resources are essential for this architecture. One iteration of decoding takes multiple cycles, which means the throughput is lower compared to full-parallel decoders. However, the decoding circuit is much smaller (Fig. 1).

Although the routing network is much simpler in partial parallel decoders comparing to full parallel ones, it is still a challenge for efficient hardware implementation, specially when the high throughput requirements force large number of implemented processing nodes in hardware. Adjusting the interconnection network to different partitions is achievable by utilizing Permuter networks^[10], or in general a network of muxes^[11]. Since the routing network becomes adjustable, reconfigurability can be entered into the design, therefore this architecture is ideal for standards supporting multiple code rates.

3. SPLIT-ROW

A threshold decoding method is proposed for Split-Row to compensate for the difference between minimums among the partitions and therefore improve the error performance with negligible additional hardware (Fig. 3). The basic idea is that each partition sends a signal to the next partition if its own local minimum is smaller than a threshold (T). Thus, the other partition is notified if there exists a minimum smaller than the threshold. The algorithm is explained below. Similar to the MinSum decoder, the first and second minimums (Min1, Min2) in each partition are computed locally. The proposed algorithm checks if Min1 is less than Threshold T then both Min1 and Min2 are used to update α values. Additionally, a threshold signal (Threshold_en) which goes to the next partition is asserted high, indicating that the minimum in this partition is smaller than threshold T .

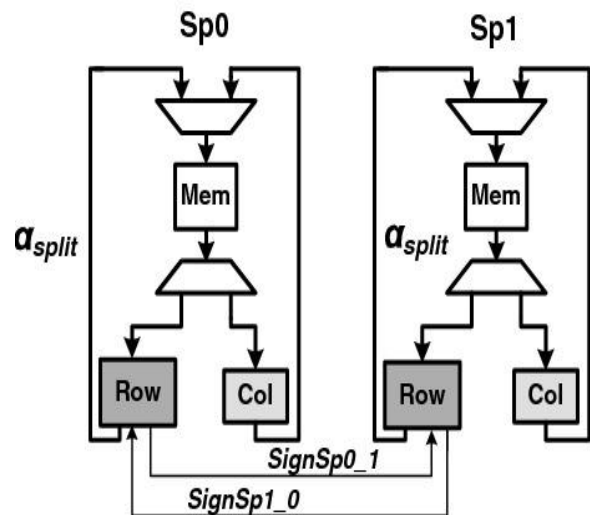


Fig 2. Split Row Threshold Method

The Split-Row decoder partitions the check node processing into two or multiple nearly independent partitions, where each block is simultaneously processed using minimal information from an adjacent partition (Fig. 2). The key idea of Split-Row is to reduce communication between check node and variable node processors which is shown to have a major role in the interconnect complexity of existing LDPC decoding implementations.

4. PROPOSED ARCHITECTURE

A Partial parallel architecture^[1] for the proposed system is shown in Fig. 3. Architecture includes check nodes and central node unit, check node unit performs the code generation based on the data elements present in the matrix. The proposed architecture is responsible for 16 bit code generation, all check nodes are controlled by central node unit. Output from all control node units are compared with the parity check matrix value, absolute detected output from parity check matrix is matched with decision reference and produces the decoded data. (Fig. 3)

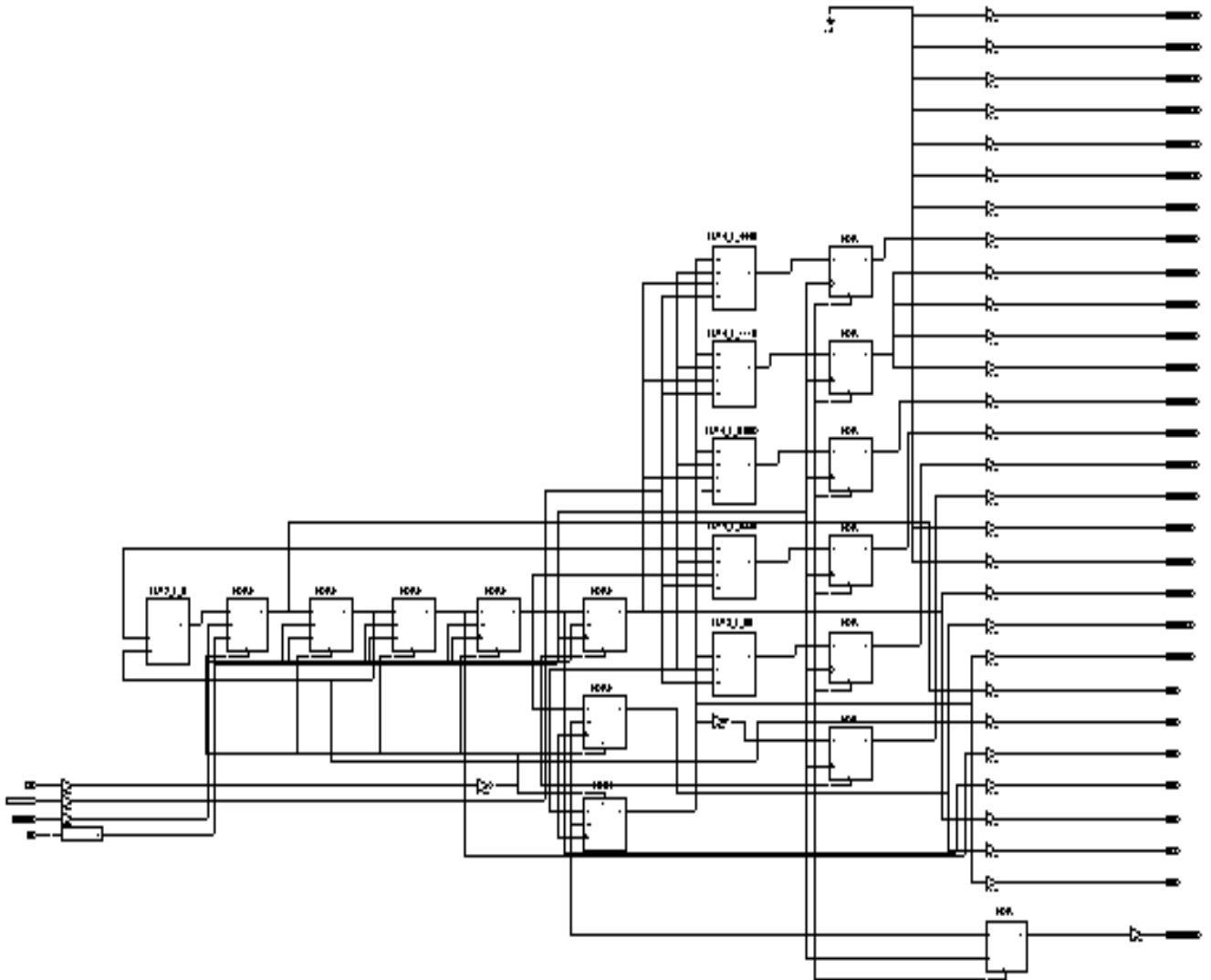


Fig 3. Partially Parallel Architecture

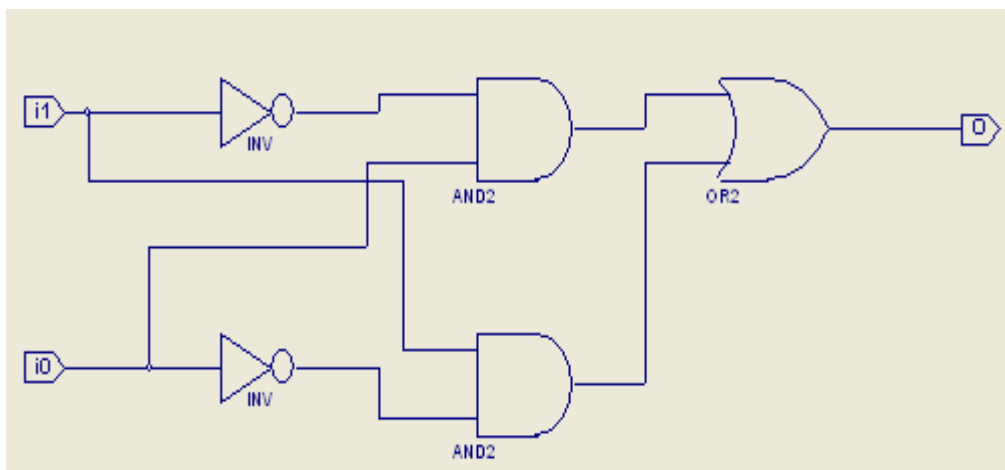


Fig 4. Internal circuit of check node unit

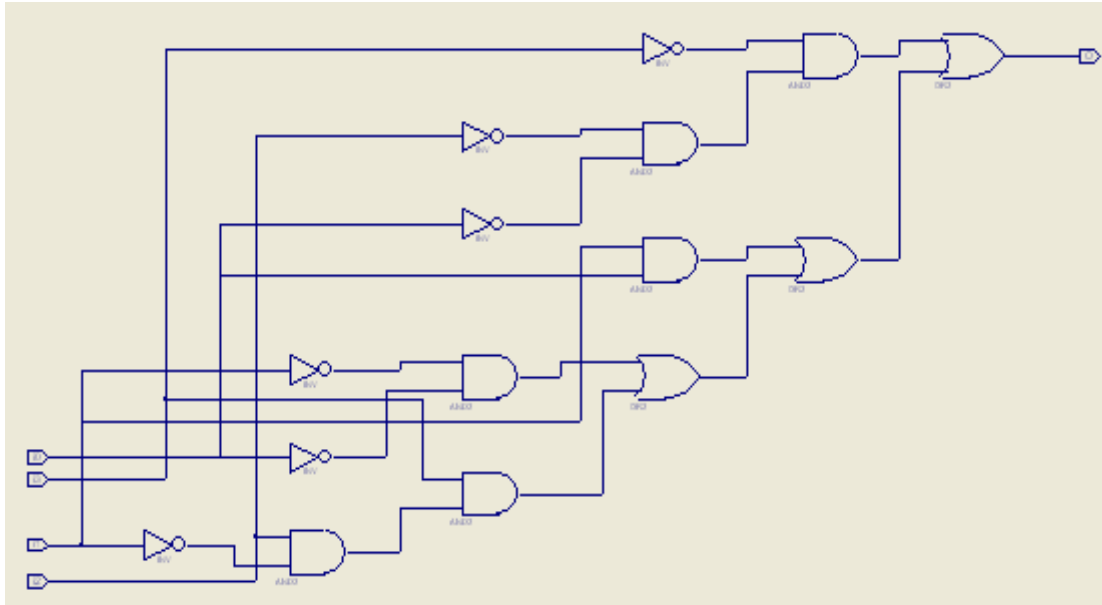


Fig 5. Internal circuit of control node unit

5. LDPC MATRIX

LDPC codes are defined by an $M \times N$ binary matrix called the parity check matrix H . The number of columns, represented by N , defines the code length. The number of rows in H , represented by M , defines the number of parity check equations for the code. Column weight W_c is the number of one's per column and row weight W_r is the number of one's per row. LDPC codes can also be described by a bipartite graph or Tanner graph^[11].

$$H = \begin{pmatrix} 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 1 & 0 \end{pmatrix}$$

$$Z = Hxr$$

where Z is the code output and r is the data from control nodes. Data elements are compared with the row of the parity check matrix. LDPC codes are defined by a sparse parity-check matrix. This sparse matrix is often randomly generated,

subject to the sparsity constraints. The LDPC codes are generated by comparing central node see Fig.5 output values with the check matrix data. Decoded output is obtained when the data is compared with the check node (Fig. 4) output codes, if the code is absolutely matched with the code generated by the control node.

6. SIMULATION RESULTS

Simulation results for BER comparison of various decoder architectures are shown in Fig. 8 and the performance of different iterations is shown in Fig.9. The following labeling is used for the figures: "MS" for normalized Min-Sum, "MS Split-Row" for the extended method Min-Sum Split-Row algorithm and "S" for the scaling factor. The performances of irregular LDPC codes are illustrated in the figures given below. Simulations were run to determine the performance of these LDPC codes in AWGN channel with BPSK modulation.

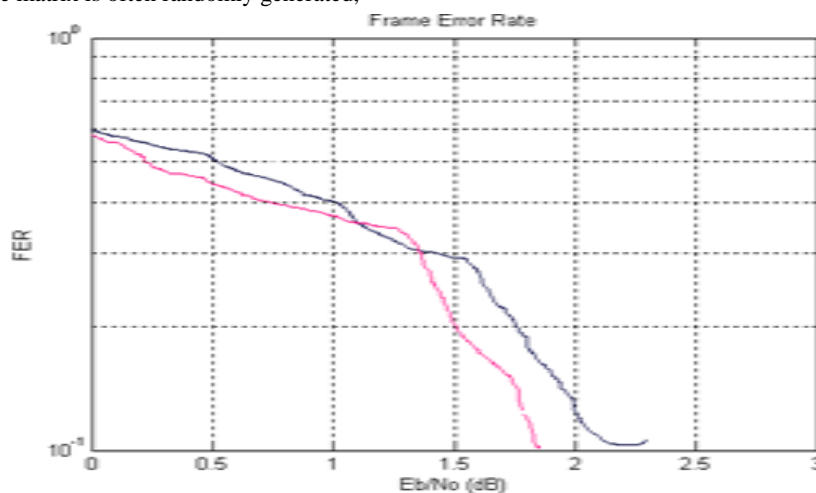


Fig 6. FER Result for different iteration

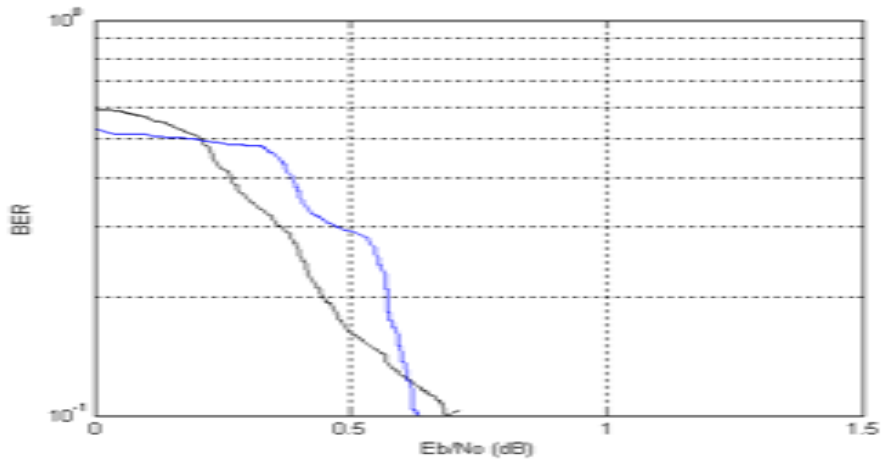


Fig 7. BER Performance Result for different iteration

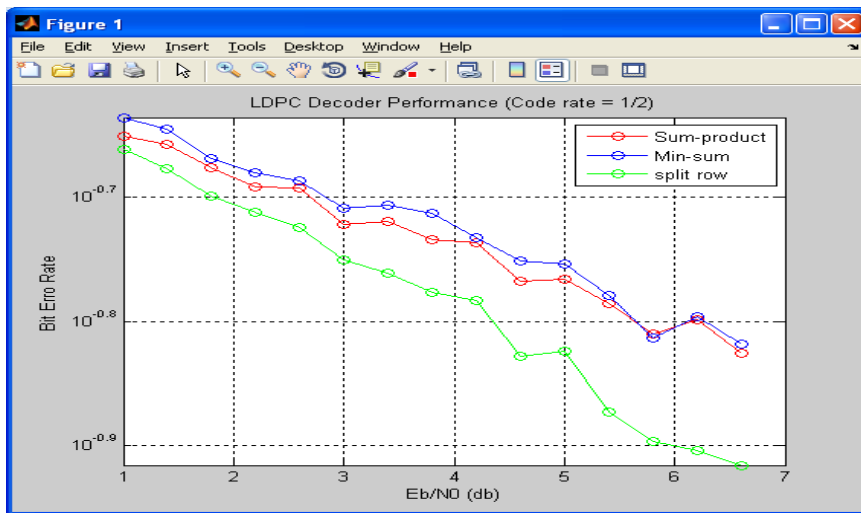


Fig 8. BER Comparison of Split Row with other Algorithms

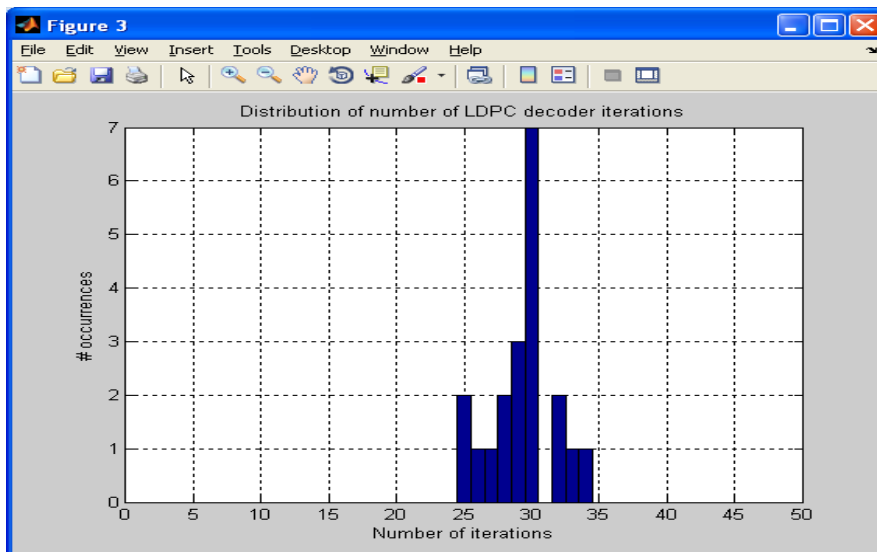


Fig 9. Iterative Comparison LDPC decoder

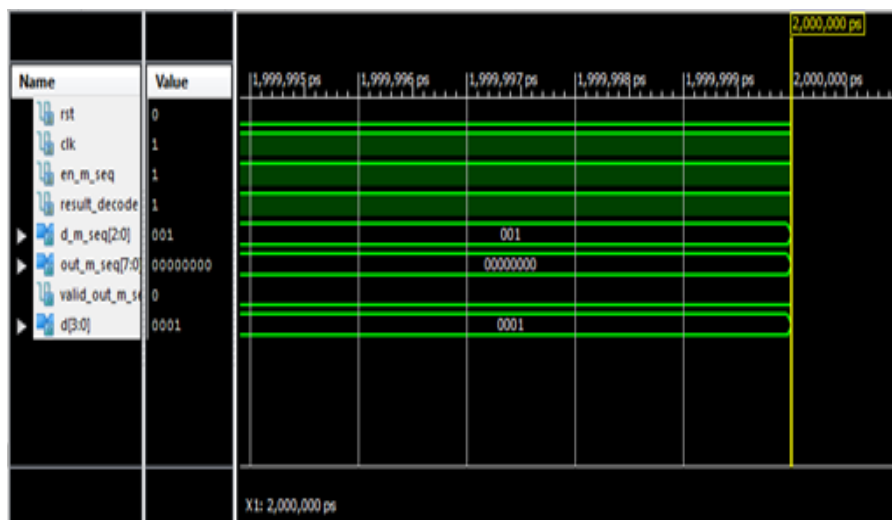


Fig 10.Xilinx Verilog output for check node processing

7. CHECK NODE PROCESSING

A LDPC code is defined as binary matrix called parity check matrix H . Rows define parity check constraints between encoded symbol in a code word and columns defines the length of the code.

V is the valid code word if $H \times V^T=0$

Decoder in the receiver checks the condition $H \times V^T$ is valid or not. If this condition is satisfied then the received output contains no error which is shown in Fig. 10.

8. RESULT COMPARISON

The comparison output for both fully parallel and partially parallel architecture is obtained using Xilinx Verilog code is shown in Table1. Table 1 deals with the delay analysis of the proposed work with the existing work. Delay values are the most important factor in the improvement of throughput. The role of split row algorithm reduces the complexity in check node processing.

Table 1. Performance output comparison.

APPROACHES	EXISTING WORK	PROPOSED WORK
Architecture	Fully parallel architecture	Partially parallel architecture
Algorithm	Min sum algorithm	Split row threshold algorithm
Operating Frequency	382.15MHz	271.66MHz
Input delay for 8 bit data	13.5ns	5.825ns
Input delay for 16 bit data	20.31ns	10.23ns
Output delay for 8 bit data	15.623ns	6.480ns
Output delay for 16 bit data	24.723ns	13.364ns

9. CONCLUSION

In this paper high performance LDPC architecture for wireless networks was proposed, which requires the modification in LDPC architecture using Partially-Parallel structure with Min sum Split Row threshold decoding algorithm. Simulation and synthesis results show that better performance in throughput of 60 to 73% and BER of 0.23 dB for a LDPC code in comparison to the Split-Row decoding algorithm.

10. REFERENCES

- [1]. M. MariaRubiston, Rajasekar. B, Logashanmugam. E “High Performance LDPC Architecture for Wireless Networks” International conference on Innovations in Intelligent Instrumentation Optimization & Signal Processing, Karunya University, 1st & 2nd March 2013, pp. 48-52.
- [2]. Y. Chen, K.K. Parhi, “Overlapped message passing for quasi-cyclic low-density parity check codes,” IEEE Trans. Circuits and Syst. I, vol. 51, pp. 1106-1113, Jun. 2004
- [3]. Banksby, A.J., Howland, C.J. A 690-mW 1-Gb/s 1024-b, rate-1/2 low-density parity-check code decoder. IEEE J. Solid-State Circuits **2002**, 37, 404–412.
- [4]. Quaglio, F.; Vacca, F.; Castellano, C.; Tarable, A.; Masera, G. Interconnection framework for high-throughput, flexible LDPC decoders. In Proceedings of the Design, Automation and Test in Europe (DATE '06), Munich, Germany, 6–10 March 2006; pp. 1–6.
- [5]. Moussa, H.; Baghdadi, A.; Jezequel, M. Binary de bruijn on-chip network for a flexible multiprocessor LDPC decoder. In Proceedings of the 45th Annual Design Automation Conference, Anaheim, CA, USA, 9–13 June 2008; pp. 429–434.
- [6]. Shih, X.Y.; Zhan, C.Z.; Wu, A.Y. A 7.39 mm, 2 76 mW (1944, 972) LDPC Decoder Chip for IEEE 802.11n Applications. In Proceedings of the IEEE Asian Solid-State Circuits Conference (A-SSCC '08), Fukuoka, Japan, 3–5 November 2008; pp. 301–304.

- [7]. Muller, S.; Schreger, M.; Kabutz, M.; Alles, M.; Kienle, F.; Wehn, N. A novel LDPC decoder for DVB-S2 IP. In Proceedings of the Design, Automation and Test in Europe Conference and Exhibition (DATE '09), Nice, France, 20–24 April 2009; pp. 1308–1313.
- [8]. Xiang, B.; Bao, D.; Huang, S.; Zeng, X. An 847-955Mb/s 342-397mW dual-path fully-overlapped QC-LDPC decoder for WiMAX system in 0.13 μ m CMOS. *IEEE J. Solid-State Circuits* **2011**, 46, 1416–1432.
- [9]. Lechner, G.; Sayir, J.; Rupp, M. Efficient DSP implementation of an LDPC decoder. In Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP '04), Montreal, QC, Canada, 17–21 May 2004; pp. 665–668.
- [10] M. Karkooti, P. Radosavljevic, and J.R. Cavallaro. Configurable, high throughput, irregular LDPC decoder architecture: Tradeoff analysis and implementation. In *ASAP*, pages 360–367, Sep. 2006.
- [11] Zhengya Zhang, V. Anantharam, M.J. Wainwright, and B. Nikolic. An efficient 10GBASE-T ethernet LDPC decoder design with low error floors. *Solid-State Circuits, IEEE Journal of*, 45(4):843–855, Apr. 2010.
- [12] R. Tanner, “A recursive approach to low complexity codes,” *IEEE Transaction of Information Theory*, vol. 27, pp. 533–547, Sept. 1981.
- [13] R. El Alami, C. B. Gueye, M. Boussetta, M. Mrabti and M. Zouak, “Reduced complexity of decoding algorithm for Irregular LDPC Codes using Split Row Method” accepted in Proc. Int. Conf. on Multimedia Computing and Systems, Ouarzazate, Morocco, Apr 2010
- [14] T. Mohsenin and B. Baas, “Split-row: a reduced complexity, high throughput LDPC decoder architecture,” in Proc. ICCD, Oct. 2006.
- [15] T. Mohsenin and B. Baas, “High-throughput LDPC decoders using a multiple Split-Row method,” in *ICASSP*, 2007, vol. 2, pp. 13–16.