HDL Implementation of an Efficient Partial Parallel LDPC Decoder Using Soft Bit Flip Algorithm

Sandeep Kakde* and Atish Khobragade**

ABSTRACT

Nowadays Low Density Parity Check Codes (LDPC) is more in demand in Wireless Communication Systems due to its excellent performance. Use of LDPC code for encoding and decoding purposes found to be more reliable and highly efficient data transfer over wide bandwidth in presence of corrupting noise. Various Algorithms were used to interpret LDPC codes. Out of which Sum Product Algorithm and Min Sum Algorithm are very much complicated because of their high interconnection complexity, high memory requirement, high variable and check node column degree but their error correcting performance is best as compared to other decoding algorithms. Bit flipping algorithms have lower interconnection complexity because it require only small amount of simple computations. We have used soft bit flip algorithm for decoding which uses advantages of both sum product algorithm and bit flip algorithm. Therefore, soft bit flip algorithm reduces the interconnection complexity and its error correction performance is near to min sum and sum product decoding algorithms. VLSI architecture of Soft Bit Flip Algorithm based Decoder is proposed which uses the value reuse property of Bit Flip Algorithm results in high throughput. The 64-bits half code rate LDPC decoder is designed using Xilinx xc5vlx110t-2ff1136 FPGA device. The calculated decoding throughput is 1.13 Gbps. The pipelining system used in LDPC decoder architecture results in enhanced decoding throughput.The proposed work has been implemented and synthesized using Xilinx ISE 14.7 and Virtex-5 Field Programming Gate Array.

Keywords: LDPC code, Soft Bit Flip algorithm, Min Sum algorithm, Iterative decoding, VHDL, Throughput.

1. INTRODUCTION

The Low-density Parity Check (LDPC) codes are forward error correcting codes used in noisy communication channels. It is used to reduce probability of loss of information. Gallager while doing his PhD work in 1962s at MIT first invented it. They were not popular so not used for a few decades due to its high computational and high implementation complexity. Almost after thirty-five years, Mackay and Neal reintroduced it in 1997, after that, they gain popularity. Nowadays LDPC codes are use as one of the most popular forward error correcting technique that can achieve a bit error rate performance near to the Shannon limit [1]. LDPC codes is one type of block codes, they can be represent either in matrix form or in graphical form. Its matrix representation called as parity check matrix or sparse matrix because it consists of less number of non- zero elements. Its graphical representation is shown by tanner graph or Bipartite Graph. LDPC matrixes are mainly of two types i.e. Regular LDPC codes and Irregular LDPC codes. Regular LDPC codes is defined as a Sparse Parity Check Matrix having fixed weight of rows and fixed weight of columns. While In Irregular LDPC codes neither weight of rows is fixed nor weight of columns. Weight of the row and weight of the column indicates number of 1's present in respective row and column.

* Department of Electronics Engineering Yeshwantrao Chavan College of Engineering, Nagpur, Maharashtra, India, *Email: sandip.kakde@gmail.com*

** Department of Electronics Engineering Rajiv Gandhi College of Engineering, Nagpur, Maharashtra, India, *Email: atish_khobragade@rediffmail.com*

The rest of the paper is organized as follows. Section 2 focuses on graphical representation of LDPC code. Section 3 explains the basic steps followed in Soft Bit Flip algorithm. The design of reduced complexity based decoder is presented in section 4. Section 5 focus on verification and synthesized results. Finally, last section concludes the paper.

2. GRAPHICAL REPRESENTATION OF LDPC CODES

Graphical representation of LDPC code is called as bipartite graph. Bipartite graph accomplishes two sets of nodes namely check nodes and variable nodes. Number of check node present in bipartite graph is equal to number of rows in sparse parity check matrix and number of variable node is equal to number of column in parity check matrix. These variable and check node connected to each other by line called as edge. An edge between variable node and check node is drawn only if there is one in the parity check matrix. In bipartite graph, number of lines connected to variable nodes is called degree of variable node and number of lines connected to check node called as degree of that check node. Now let's see graphical representation of Parity Check Matrix H of size (4, 8) shown in Fig. 1.

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

Figure 1. Parity check matrix of size (4, 8)

There are four rows so there are four check node denoted as c1, c2, c3 and c4. Moreover, there are eight columns in matrix so there are eight variable node denoted as v1, v2, v3,........ v8.
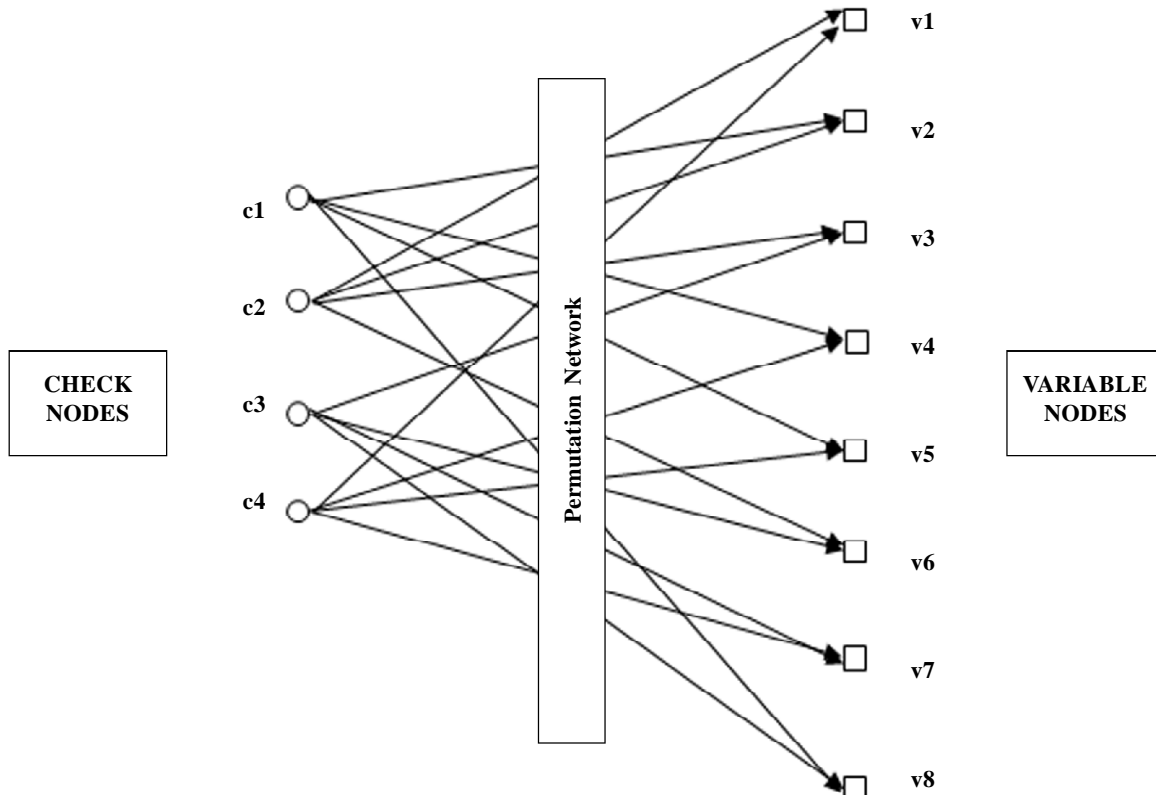

Figure 2: Bipartite Graph Representation of Sparse Parity Check Matrix of size (4, 8)

Now consider first row of parity check matrix means first codeword, one is present only in 2nd, 4th, 5th and 8th position so check node c1 connected to variable node v2, v4, v5 and v8. In similar manner, all other nodes are connected as shown in Fig 2. In order to reduce the area, common processing unit is proposed and soft bit flip algorithm based LDPC decoder was designed [6]. The SBF algorithm was same as the conventional Bit Flip algorithm but was used to reduce the interconnection complexity and the amount of computations, while it applies consistency estimation which imitates the SPA to viaduct the performance slit. VLSI Architecture of high throughput soft bit flip algorithm based decoder was proposed [7] and a hybrid scheme which is combination of the BF and SBF algorithms is also proposed to shorten the decoding time. FPGA Implementation of a LDPC Decoder using a Reduced Complexity Message Passing Algorithm is proposed [8] which is a simplified message passing algorithm for reducing the implementation complexity of decoder. This algorithm uses simple hard-decision decoding techniques which also utilizing the soft information to be processed for improvement in decoder performance. Stream-Based Processor which is reliability ratio-based, weighted bit-flipping algorithm was proposed [9]. Bit flip algorithm based encoder and decoder was designed and implemented on Virtex-5 FPGA family [11].

3. SOFT BIT FLIP ALGORITHM

The soft bit flip algorithm is simply the basic architecture of bit flip algorithm. Interconnection complexity simply reduced by using variant of bit flip algorithm. Its bit error performance is also improved as this algorithm is mixing of sum product and bit flip algorithm. Following are the steps [6] followed in soft bit flip algorithm. Consider a regular (N,K) LDPC codes defined as M by N sparse parity-check matrix $H = [h_m, n]$. Let $B(m) = \{n \mid h_m, n = 1\}$ be the set of variable nodes which take part in the mth check node. The bit vectors are transmitted using BPSK modulation with AWGN channel. Let us assumed that the received block word as $y = (y_1, y_2, y_3,........., y_N)$ and its hard decision vector $z = (z_1, z_2, z_3......, z_N)$. We have used this algorithm and define architecture for LDCP Codes.

Steps of Algorithm:

Step 1: Initialization: All message passing algorithms are Iterative one because at each round of the algorithm, message are passed from variable nodes to check nodes and from check nodes to variable nodes. All Log Likelyhood values are stored initially.

Step 2: Variable Node to Check Node: The information bits from variable nodes to check nodes are computed on the basis of sending the Log Likelyhood values.

Step 3: Check Node to Variable Node : Received values of the variable node and some of the information bits passed from adjacent check nodes to that variable node. A vital point is that the message that is sent from a variable node v to check node c must not take into account the message sent in the last round from c to v. The same is true for messages that are passed from check nodes to variable nodes.

Step 4: Decision

Figure 3 shows the Internal Block Design of Soft Bit Flip Decoder Module and Figure 4 shows HDL implemented top Module. We have designed 64-bit codeword decoder. It mainly consist of Variable node processing element (VNPE), Check node processing element (CNPE), add/sub unit, saturator and multiplexer. And two shift register one for storing variable node and other for check node. As decoding process start firstly multiplexer initialize with 1 so that received codeword directly stored at variable shift register bit by bit. After finishing check node updates, Check Node Processing Element (CNPE) will process on all connected check nodes. Check nodes updated by Variable Node Processing Element (VNPE) units are stored in shift register one bit by bit by shifting it once at every cycle and which given as input to CNPE unit. Check nodes further process in it and calculate flipping value. Flipping value is very important parameter. Comparing flipping value with threshold values and then CNPE unit detect flipping power and accordingly take decision of flipping for output variable node.
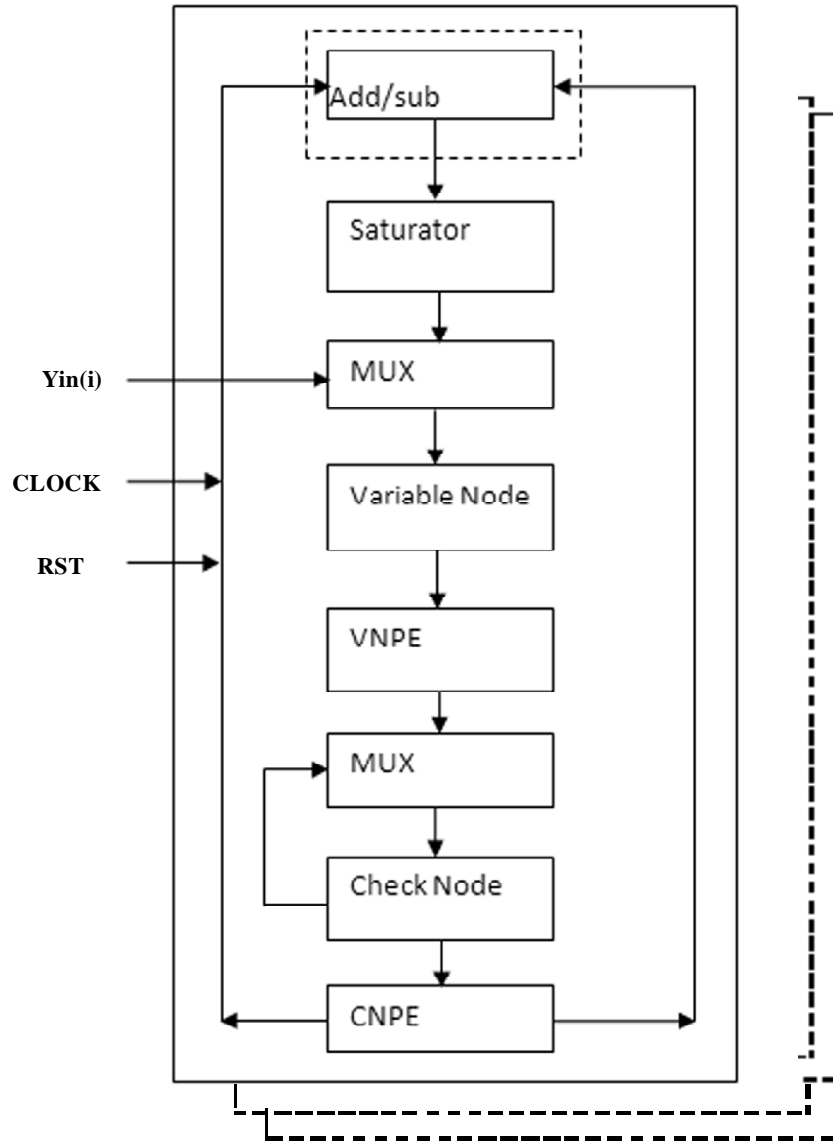
4.   HDL IMPLEMENTATION

Figure 3: Internal Block Design of Soft Bit Flip Decoder Module
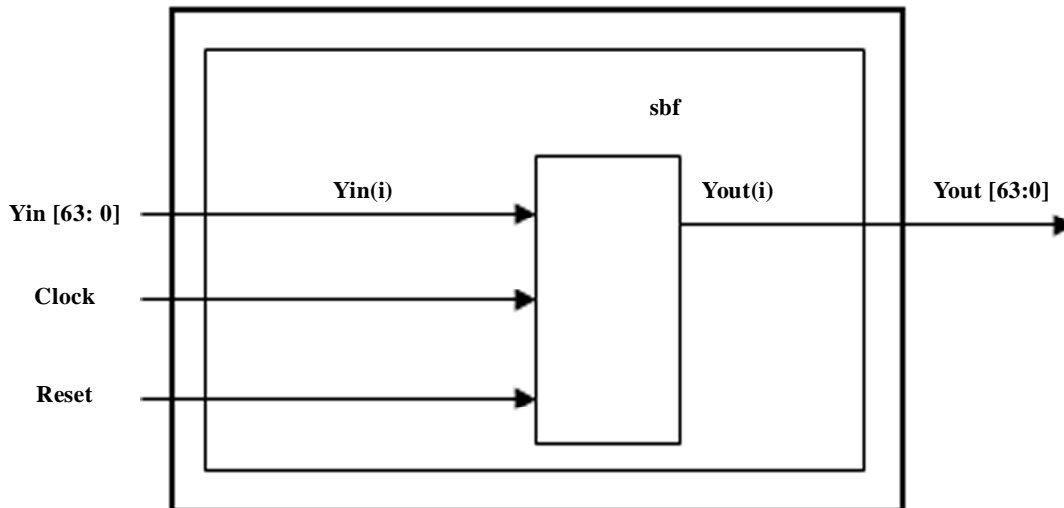

Figure 4: Top Level Module of Soft Bit Flip LDPC decoder

5. SYNTHESIS RESULTS

Table 1 shows the throughput comparison with existing works and Table 2 shows FPGA Device Utilization Summary of LDPC Decoder. A Register Transfer Level (RTL) description of the various block and soft bit flip based decoder is written in VHDL. The maximum clock frequency and number of slices required for designing decoder is obtained from synthesis report. The VHDL code is simulated and all designs are synthesize using Xilinx ISE 14.7 simulation tools. The maximum clock frequency and number of slices required for designing decoder is obtained from synthesis report. VHDL code is written for various block of decoder in behavioral style of modeling. The maximum throughput of the developed LDPC decoder increases to 1.13 Gbits/s when the iteration limit is reduced to 4. We have used Virtex-5 family and xc5vlx110t-2ff1136 target device. The maximum frequency from the synthesis report is 282.925 MHz.

Throughput Calculations: Throughput of the LDPC Decoder is given by the formula

$$Throughput = \frac{Coderate \times codelength \times F_{max}}{N_{it} \times \theta}$$

Where, Fmax is the maximum operating frequency, Nit is the number of decoding iterations and è is the number of clock cycle required to complete one iteration.

Table 1
Throughput Analysis

| Feature | Throughput Comparison | |
	[11]	This Work
Throughput	57.14 Mbps	1.13 Gbps

Table 2
FPGA Device Utilization Summary

| | Proposed LDPC Decoder | | |
| | FPGA Family: Virtex-5 Target Device: xc5vlx110t-2ff1136 | | |
	Used	Available	Utilization
No. of Slice Registers	3265	69120	0%
No. of Slice LUTs	5120	69120	0%
No. of fully used LUT-FF pairs	2448	5937	41%
No. of Bonded IOBs	130	640	20%

6. CONCLUSION

In this paper, we have presented an efficient partially- parallel decoder hardware implementations using bit flip algorithm. There are different ways for decoding LDPC codes. Out of which Soft Bit Flip algorithm is very easy to find out error with low complexity. The trade-off between throughput and hardware implementation of LDPC codes is essential term for the design of any LDPC decoder. The objective was to implement hardware of 64-bit LDPC decoder having high decoding throughput. We achieved our objectives by developing LDPC decoder using reduced complexity algorithm. We implemented the 64-bits, half code rate LDPC decoder using Xilinx xc5vlx110t-2ff1136 FPGA device. The calculated decoding throughput is 1.13 Gbps. The pipelining system used in LDPC decoder architecture results in enhanced decoding throughput.

REFERENCES

[1]   R. G. Gallager, "Low-density parity-check codes," IRE Trans. Inf. Theory, vol. IT-8, no. 1, pp. 21–28, Jan. 1962.

[2]   R. M. Tanner, "A recursive approach to low complexity codes," IEEE Trans. Inform. Theory, vol. IT-27, pp. 533–547, Sep. 1981.

[3]   D. J. C. MacKay and R. M. Neal, "Near Shannon limit performance of low density parity check codes," Electron. Lett., vol. 32 , no. 18, pp.1645–1646, 1996. Lett., vol. 32 , no. 18, pp. 1645–1646, 1996.

[4]   Luca Fanucci , Francesco Ross, "A Throughput / Complexity Analysis for the VLSI Implementation of LDPC Decoder" Signal Processing and Information Technology, 2004. Proceedings of the Fourth IEEE International Symposium, Pages: 409–412, 18-21 Dec., 2004.

[5]   Xiaofu Wu, Chunming Zhao, and Xiaohu You, "Parallel Weighted Bit-Flipping Decoding", IEEE Communications letters, vol. 11, no. 8, Aug 2007.

[6]   Junho Cho, Jonghong Kim, Hyunwoo Ji, and Wonyong Sung, "VLSI Implementation of a Soft Bit-Flipping Decoder for PG-LDPC Codes", 978-1-4244-3828-0/09 ©2009 IEEE.

[7]   Junho Cho, Member, IEEE, Jonghong Kim, and Wonyong Sung, Senior Member, IEEE "VLSI Implementation of a High-Throughput Soft-Bit-Flipping Decoder for Geometric LDPC Codes" IEEE Transactions on Circuits and Systems—I: regular papers, Vol. 57, No. 5, May 2010.

[8]   Vikram Arkalgud Chandrasetty and Syed Mahfuzul Aziz, "FPGA Implementation of a LDPC Decoder using a Reduced Complexity Message Passing Algorithm" Journal of Networks, vol. 6, No. 1, January 2011.

[9]   Honey Durga Tiwari, Huynh Ngoc Bao, and Yong Beom Cho, "A Parallel IRRWBF LDPC Decoder Based on Stream-Based Processor", IEEE transactions on Parallel and Distributed Systems, vol. 23, no. 12, 1045-9219/12 @ 2012 IEEE.

[10] Chinna Babu.J, S.Prathyusha, V. Usha Sree, "Hard Decision and Soft Decision Decoding Algorithms of LDPC And Comparision of LDPC With Turbo Codes, RS Codes and BCH Codes", 09th IRF International Conference, Bengaluru, India, ISBN: 978-93-84209-40-7©2014 IEEE.

[11] Straus, Pavel, and Zdenek Kolka. "Simulation and implementation of LDPC code in FPGA." In *Radioelektronika* (RADIOELEKTRONIKA), *2013 23rd International Conference*, pp. 346-349. IEEE, 2013.

[12] Gade Sribala, D.V.N. Sukanya K. Gouthami1andTai-hoon Kim, "LDPC Encoder and Decoder Architecture for Coding 3-bit Message Vector", International Journal of Security and Its Applications Vol. 9, No. 10 (2015), pp. 21-30