**Abstract**

**Objective:** VLSI implementation of Decoder Architecture for high throughput using LDPC codes. **Methods/Analysis:** In this paper, the VLSI architecture of layered partial parallel soft decoding algorithm based decoder for different code size is presented. **Findings:** LDPC codes provide remarkable error detection performance. Proposed decoder is well matched for VLSI implementation and it is implemented on Xilinx FPGA family. LDPC codes are well-known linear block codes. The computational complexity of LDPC codes is very high as compared to other existing codes like Convolutional codes and Turbo codes. The major benefit is that they offer an enhanced performance, which is very close to the Shannon’s capacity for many different channels and complex algorithms for decoding. **Novelty/Improvement:** By using layered partial parallel soft decoding algorithm, we proposed a pipelined structure that is helpful to achieve higher throughput. The proposed architecture is implemented and tested on FPGA Virtex-5 family with device as 5XC5VLX85. The decoder can attain a throughput of 1.5 Gbps with reduced hardware resources.

**Keywords:** Low-Density Parity-Check Codes, Layered Partial Parallel Soft Decoding Algorithm (LPPSD), Parity Check Matrix (PCM), Soft Decoding Algorithm Processing Element (SDPAE)

**1. Introduction**

VLSI-based execution of the codes has been of very high curiosity in recent times. The recent interest in different decoding algorithm has directed to the encounter of LDPC Codes. R. Gallager invented LDPC Codes in 1960. Initially it was invented but it was too intricate in implementation, so it was not popular initially. The term LDPC comes from the property of their parity check matrix (PCM) which accomplishes only few numbers of 1’s in comparison to the numbers of 0’s. Their main advantage is that they deliver a superior performance that is very close to the Shannon’s capacity for many different channels. The LDPC codes, defined by a sparse PCM, are also known as linear block codes, and can be powerfully represented by the bipartite graph. The types of PCM are: quasi-cyclic matrix and the pseudorandom matrix. Furthermore, the encoding complexity rely on block size and is extensively useful in systems such as WLANs like 802.11n, 802.16e, terrestrial broadcasting applications and storage system. The various decoder architectures have been demonstrated for various applications based on iterative decoding approaches. Generally, this architecture would fall into: 1) better performance; 2) higher throughputs; 3) smaller area; and 4) more flexible configurability.

In paper 1, the author focuses on power aware LDPC Decoder architecture and mainly accomplishes Quasi-Cyclic LDPC codes are extensively useful in telecommunication systems. However, the decoders are still complicated to be implemented for real time applications due to area overhead and large dynamic power, particularly in the wireless devices. This paper presents a superior iterative decoder architecture, which can largely reduce their power. In paper 2, the author highlights an efficient low latency decoder architecture for QC-LDPC Codes. Architectural level optimizations are included in order to decrease the path delay. Improved
partially parallel architectures are designed to boost the throughput of conservative partially parallel decoders by accomplishing a small percentage of extra hardware. In Paper 3, author focuses on Non-binary low-density parity-check codes having high decoding intricacy which is a major barrier to their applications. In addition, they proposed a decoding algorithm with abridged complexities and lesser storage requirements for NB-LDPC codes. In paper 4, author concluded that NB-LDPC codes are vigorous to different channel impairments. The extreme computational complexity and storage usage of the current decoder designs are noticeably costly for real time applications. As compared to existing works, their design has some advantages. In presented quasi cyclic layered architecture of LDPC, with arrays of circulant sub-matrices generating the PCMs. They had given brief information about the layered decoding algorithm. They had also given a case study on China Mobile Multimedia Broadcasting LDPC decoder design and DVB-S2 decoder design. They compared various parameters such as Matrix Kernel Size, Clock frequency, Scaled clock frequency, Core Area, Scaled area, Throughput, Scaled throughput and area metric. The LDPC codes can be decoded using decoding scheme, and for equalization/detection. A fully parallel decoder is designed by directly instantiating the Belief Propagation algorithm to hardware. An area efficient Min Sum Algorithm based Decoder that is proposed which mainly improved the Check Node Unit. FPGA implementation LDPC decoder on Virtex FPGA was proposed which focuses on low complexity and high speed.

2. Preliminary of LDPC Codes

The difficulty of employing LDPC codes is high as compared to existing codes like Convolutional codes, turbo codes. In recent days, it is used because of well-known coding method for communication system due to its flexibility to choose various code parameters such as code length and code size.

2.1 Representation of LDPC Codes

LDPC codes can be graphically represented by bipartite graphs in which elements of row represents the check nodes and elements of column of parity check matrix represents the variable nodes. For every 1’s in the Parity Check Matrix, there exists an edge between VNU and CNU. The recital of the LDPC decoder is frequently calculated in terms of the ensemble average of codes, an outcome of the random selection of edges linking check and variable nodes for a given block length. Degree of each node depends on the number of elements other than zero in each row or column of PCM. Therefore, there are two classes of LDPC Codes as regular and irregular. Figure 1 shows the bipartite graph representation for Parity Check Matrix.

2.2 Graphical Demonstration of Parity Check Matrix (PCM)

For N code length and M data bits, the rate R is (M / N). All data message bits analogous to every 1-components of a sub matrix are stored in one memory storage location. To access every process unit to route messages matching to blocks units at numerous columns or rows of the sub-matrix in every cycle is to store many messages in each memory location. For example, as shown in Figure 2.

3. Proposed Decoder Architecture

We present the portrayal of architecture of LDPC decoder. LDPC decoder architecture consists of a number of vital blocks. They are namely: Controller design, Soft Decoding Algorithm Processing Element (SDAPE), Router or Shifter, PCM for random or unstructured LDPC codes, Memory blocks (RAMs or ROMs).
Figure 3 shows the LDPC decoder architecture. Operation of decoding starts by selecting the suitable block length from the coded message block. By using these initial values, the precise values of PCM are chosen from different ROMs. Now these values from special memory locations are used to create the addresses of inherent Checks and APPs memories. These APP values are delivered to the shifter block. Work of shifter/router is to provide the values from different APPs to the correct SDAPE units. In different SDPAEs the received values from router is valued and retrieve the original message. Through all processing units, the obtained result is passed through detection unit to checked with the original message which is stored in the memory to check that the correct codeword is found or not. Then, these attained values are written into APPs memories and again execution of other data word starts again. If detection unit does not originate the correct code word after an all-out limit of iterations and then decoder fails to get the original data. Using different control signals like enable, reset, clock, load and some hand-shaking signals Controller block controls the data flow between various parts of decoder architecture. The architecture contains a number of memory blocks for storing the messages, the shifter blocks to provide the path for the data to various nodes (SDAPE), address generator block to pick the data from various Checks and APPs memories and a control unit.

3.1 Soft Decoding Algorithm Processing Element (SDAPE)

Decoder architecture uses a number of SDAPEs block to provide parallelism in the design. In the decoder architecture, they are accountable for the Checks and APPs messages and to produce the new values of Checks and APPs. For example, for 972 code lengths, 12 processing nodes with sub-matrix size block of 54×54 work concurrently. A typical decoding processing unit blocks accomplishes several block Processing Units (PU) and many global memory modules as shown in Figure 3 and Figure 4. The SDAPEs and the global memory modules communicate with each other via a permuter network. The permuter network could be one or more
shared buses. Each SDAPE module accomplishes a
restricted memory module, a block processing element
unit, which accomplishes a scalar unit, Shifter, Counter,
XOR, Latch, Compare block as shown in Figure 4.

Figure 4. SDPAE block.

4. Partial Layered Parallel Soft
Decoding Algorithm

Parallel means by grouping numbers of blocks and
processed concomitantly. The decoder architecture
proposed in this paper is nearly two times faster than the
existing decoders and uses the layered parallel algorithm.
It uses parallel algorithm by selecting a convinced number
of Check and Variable nodes into a group for parallel
processing. Consider \( P_{mn} \) signify the check node message
that is sent from the check nodes \( m \) to the variable nodes
\( n \). The \( S(t_n) \) \( (n = 1, 2, 3 \ldots N) \) denotes the posteriori
probability messages. For each variable node \( n \) inside the
present horizontal layer, \( S(t_{mn}) \) which corresponds to a
specific check message equation \( m \) are calculated as:

\[
S(t_{mn}) = S(t_n) - P_{mn}
\]  

For every check node , messages \( P_{mn} \) corresponds
to the variable nodes \( n \) that appears in a specific check
equation, which are calculated as:

\[
P_{mn} = \prod_{n' \in N(m) \setminus \{n\}} \text{sign}(S(t_{mn})) \phi \left[ \sum_{n' \in N(m) \setminus \{n\}} \phi(S(t_{mn}')) \right]
\]  

Updation of APP messages in the present horizontal
layer are:

\[
S(t_n) = S(t_{mn}) + P_{mn}
\]  

Depending on the sign of \( S(t_n) \), hard decisions are
decided after updating of each horizontal layer. If all the
check equations are fulfilled, stop the message-passing
algorithm.

4.1 Pipelining Principle Stage

The decoding of every layer of the PCM were done in three
steps namely reading from APPs, processing and writing
back in APPs after updating of new values according to
equations 1, 2 and 3 respectively. With the intention of
improving the decoding speed and throughput, all the
three steps must be performed simultaneously. The tasks
are as follows:

1) Divide every block units into intermediate-
block units where all the message bits corresponding to
1-components at adjacent rows of an intermediate matrix
are stored in different segments;

2) Store data analogous to adjoining rows of an
intermediate -matrix in one memory whereas utilizing
additional buffers to solve the memory access variance;

3) Execute steps 1 and 2.

4.2 Layered Partially Parallel Soft Decoding

As shown in Figure 2, the PCM is observed as a group
of appended horizontal layers. In reading stage, APP
messages are read from the various horizontal layers
\( (L^n) \). In addition, variable node messages are taken from
same layer \( (L^n) \) and updated as equation”(1)”. All check
node messages are simultaneously updated as equation
no “(2),” constructed on the updated check bits from the
\( (L-1)^{th} \) layer. It must be noticed that each recently updated
APP messages are not used straight. So, APP messages are
updated as:

\[
S(t_n^{(k,L-2)}) = S(t_n^{(k,L-2)-P}) - P_{mn}^{(k-1,L-2)} + P_{mn}^{(k,L-2)}
\]

As we have already discussed that throughput of LDPC
Codes, be determined by constraints: code word length,
code Rate, total iterations for decoding each message,
total clock cycles for each iteration and maximum clock
frequency. Figure 5 shows the Flow chart of Proposed
Algorithm.
5. FPGA Implementation and Measurement Result

On platform of Xilinx Virtex 5XC5VLX85 FPGA, we presented a decoder with block length 972 and 54 shared row and column processors work concurrently. Table 1 shows the Statistics of the FPGA Utilization. After simulation, the maximum frequency achieved is 371 MHz, the number of iterations required is eight to decode each message, and number of clocks required for each iteration is twenty. For 648-code length the throughput of the proposed decoder is 1.5 Gbps.

Table 1. Statistics of the FPGA utilization

| LDPC Decoder          | Family: FPGA Xilinx Virtex | Device: 5XC5VLX85 |
|-----------------------|-----------------------------|-------------------|
| Resource Used         |                             |                   |
| Slice Registers       | 2096                        |                   |
| Slice LUTs            | 1272                        |                   |
| LUT-FF pairs          | 768                         |                   |
| Bonded IOBs           | 1839                        |                   |
| Buffers               | 1                           |                   |

6. Performance Analysis and Comparison

Table 2 compares this decoder with different existing LDPC decoders. There are number of parameters taken in consideration while comparisons are made with other Decoder Architectures like throughput, code length, code rate, iterations, and resources at different platform of FPGA. Table 2 shows the comparison of this work with Existing Architectures. The design is implemented in Verilog HDL, synthesized, placed using Xilinx development EDA Tool ISE13.4

Table 2. Comparison of this work with Existing architecture

| Parameter   | [8] Platform | [9] Vertical 2 | Proposed |
|-------------|--------------|----------------|----------|
| Code length | 1008         | 8176           | 648      |
| Code rate   | 1/2          | 7/8            | 2/3      |
| Iteration   | 6            | 15             | 8        |
| Throughput  | 142 Mbps     | 170 Mbps       | 1.5 Gbps |

7. Conclusion

The partially parallel structure and VLSI architecture of a 648-bit 2/3 code rate LDPC decoder is designed in this paper. The hardware efficiency of proposed approach in this paper is better than the existing decoder architectures. As a concern, this design of a LDPC decoder symbolizes an inspiring and momentous case of study. In proposed
architecture, we presented a decoder that supports variable block size structured LDPC irregular codes. Partially parallel decoder architecture is accomplished in order to balance between the speed and complexity. Furthermore, this architecture is suitable for VLSI technology, and it can be well mapped to the partial decoder configuration. We proposed a pipelined structure that is helpful to accomplish an improved throughput by using layered partial parallel soft decoding algorithm. This architecture is implemented on FPGA Xilinx Virtex 5XC5VLX85. The decoder for code length of 648 at clock frequency of 371 MHz using eight iterations can achieve a throughput of 1.5 Gbps. Because these designs can achieve a very high throughput and pipelining can be applied to reduce their timing constraints although it still maintaining sufficient throughput for high-speed communication applications like WLANs.

8. References

1. Xiang B, Shen R, Pan A, Bao D, Zeng X. An area-efficient and low-power multirate decoder quasi-cyclic low-density parity-check codes. IEEE Transactions on Very Large Scale Integration Systems. 2010 Oct; 18(10):1447–60.
2. Wang Z, Cui Z. Low-complexity high-speed decoder design for quasi-cyclic LDPC codes. IEEE Transactions on Very Large Scale Integration Systems. 2007 Jan; 15(1):104–14.
3. Lin J, Yan Z. An efficient fully parallel decoder architecture for nonbinary LDPC codes. IEEE Transactions on Very Large Scale Integration (VLSI) Systems. 2014 Dec; 22(12):2649–660.
4. Xiaoheng C, Wang C-L. High-Throughput Efficient Non-Binary LDPC Decoder Based on the Simplified Min-Sum Algorithm. IEEE Transactions on Circuits and Systems-1: Regular Papers. 2012 Nov, 59(11):2784–94.
5. Sun Y, Cavallaro JR. VLSI architecture for layered decoding of QC-LDPC codes with high circulant weight. IEEE transactions on Very Large Scale Integration systems. 2012 Oct; 21(10):1960–4.
6. Kasiselvanathan M, Kumar NS. BER performance analysis and comparison for large scale MIMO receiver. Indian Journal of Science and Technology. 2015; 8(35).
7. Zhang T. A 54 Mbps (3, 6) –regular FPGA LDPC decoder. IEEE Workshop on Signal Processing Systems; 2002. p. 127–32.
8. Hang J, Xu C, Zhong Q, Zhong G. A kind of low complexity LDPC decoder. Second International Symposium on Computer Science and Computational Technology; 2009 Dec. p. 102–5.
9. Cui Z, Wang Z. A 170Mbps (8176, 7156) Quasi-cyclic LDPC decoder implementation with FPGA. 2006 IEEE International Symposium on Circuits and Systems. Island of Kos, 2006 May, pp. 5095-98.
10. Karkooti M, Radosavljevic P, Cavallaro JR. Configurable, high throughput, irregular LDPC decoder architecture: Tradeoff analysis and implementation. IEEE 17th International Conference on Application-Specific Systems, Architectures and Processors (ASAP'06); 2006 Sep. p. 360–7.