A Multi-Level Hierarchical Quasi-Cyclic Matrix for Implementation of Flexible Partially-Parallel LDPC Decoders

Authors Name
Affiliation
Address
Email ID

***Abstract*—A novel technique for constructing a multi-level Hierarchical Quasi-Cyclic (HQC) matrix for Low-Density Parity-Check (LDPC) decoder is presented in this paper. A unique multi-level structure of the proposed matrix provides flexibility in generating different code lengths and code rates for various applications such as WiMAX, WLAN and DVB-S2. In addition, different combinations of permuted sub-matrices are embedded in layers, to provide virtual randomness in the LDPC matrix. Simulation results show that the HQC matrices generated by using the proposed technique has bit error rate (BER) performance very close to that of unstructured random matrices. A prototype model of partially-parallel decoder architecture has been designed by using the various matrix configurations available in the proposed technique. FPGA implementation results show that the designed architecture leads to a resource efficient LDPC decoder.**

***Keywords*- Digital communication, error correction codes, flexible structures, cyclic codes, logic design, field programmable gate array**

I. INTRODUCTION

Low-Density Parity-Check (LDPC) codes have emerged as one of the most popular forward error correcting (FEC) technique that can achieve bit error rate (BER) performance close to Shannon Limit [1]. The inherent structure of the LDPC matrix provides high degree of parallelism and flexibility for designing a decoder for various applications – Worldwide Interoperability for Microwave Access (WiMAX), Wireless Local Area Network (WLAN) and Digital Video Broadcasting - Satellite - Second Generation (DVB-S2) [2]. A fully-parallel architecture implementation of an LDPC decoder provides very high throughput but requires large hardware resources to achieve this performance [3]. Also, the complexity of the decoder increases drastically with longer code lengths. Therefore, an alternate solution to this problem is to use resource efficient partially-parallel architecture [4]. This architecture uses only a few number of decoding nodes and reuses them iteratively in the process. Unlike that in a fully-parallel decoder, it also utilizes block memories (in an FPGA) to store and access intermediate extrinsic messages. However, the advantages of a partially-parallel architecture are achieved by sacrificing the throughput of the decoder due to additional clock cycles required for processing [5].

A partially-parallel decoding architecture provides a trade-off between hardware requirements and throughput. The number of parallel nodes (check node and variable node) required by the decoder is based on the partition size of the matrix (also known as the base matrix). Also, the complexity of the addressing scheme required for handling intermediate messages substantially depends on the structure of the LDPC matrix. Therefore, the hardware requirement of a partially-parallel architecture based decoder predominantly relies on the structure and complexity of the LDPC matrix [6]. In order to alleviate the complexity of the decoder, structured Quasi-Cyclic (QC) [7] based matrix construction methods are widely used. This technique constructs an LDPC matrix by using an array of cyclically-shifted base matrices [8]. The parallelism factor of partially-parallel decoder architecture is normally defined by the size of the base matrix. Hierarchical QC (HQC) [9] matrixes are constructed with several levels of sub-matrices, with the last level corresponding to the base matrix. HQC based technique has the flexibility for constructing LDPC matrices of variable code lengths and code rates [10]. However, not all QC based matrix leads to comparable decoding performance (BER and average iterations) to that of unstructured matrices [8]. Therefore, constructing an LDPC matrix that reduces the complexity of partially-parallel decoder and also achieve optimum decoding performance is a challenge.

This paper presents a 3-Level HQC (3L-HQC) matrix construction technique. The 3-Levels of hierarchy in the matrix provide flexibility of generating LDPC codes of different code lengths and code rates. The matrix can also be easily configured for applications such as WiMAX, WLAN and DVB-S2. The proposed 3L-HQC consists of a permuted matrix in the level-2 of the hierarchical structure. Different combinations of permuted matrices are inserted in layers of the LDPC matrix to provide randomness in the matrix structure. Simulation results show that the proposed matrix has a marginal degradation in BER performance compared to the unstructured random matrices. It also outperforms the 2-Level HQC based LDPC decoders [9]. FPGA implementation of a partially-parallel architecture using the proposed 3L-HQC matrix leads to significant reduction in memory requirements of the decoder.

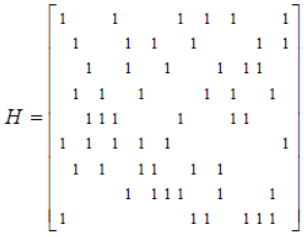
The rest of the paper is organized as follows. A brief overview of unstructured and structured LDPC matrices along with decoder implementation complexity is presented in section II. In section III, the construction and performance analysis of the proposed matrix is presented. Applicability of the proposed matrix for various applications is also presented. It is then followed by a partially-parallel decoder implementation in section IV.

II. STRUCTURE OF LDPC MATRICES

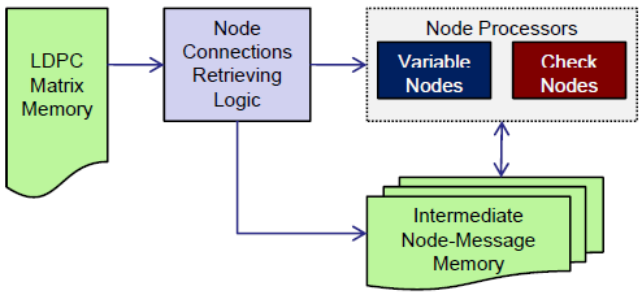
As stated earlier, the structure of the matrix plays a vital role in performance and implementation complexity of an LDPC decoder, especially for partially-parallel architectures [6]. The following sub-sections present an overview and comparison of the structured and unstructured matrices:

A. Unstructured Matrices

An unstructured LDPC matrix is a sparse matrix having the non-zero elements arranged in a random pattern. A sample of an unstructured matrix is shown in Fig. 1(a). Progressive Edge Growth (PEG) is one of the popular algorithms used to construct unstructured matrices [11]. The algorithms just aim to obtain matrices with large girths and can construct matrices of virtually any size. Due to the possibility of generating matrices with large girths, unstructured matrices exhibit good decoding performance [12]. These matrices can be used to design both fully-parallel and partially-parallel architectures for LDPC decoder. However, due to the randomness (complex node inter-connections) in the structure of the matrix, the implementation of both the architectures becomes substantially complex, especially for large code lengths. A simple block diagram of typical partially-parallel architecture of an LDPC decoder using unstructured matrix is shown in Fig. 1(b). Note that an exclusive memory is required to store the inter-connect information (LDPC matrix) for decoding.



(a) Sample of an unstructured LDPC matrix



(b) A simple block diagram of a partially-parallel architecture using unstructured matrix

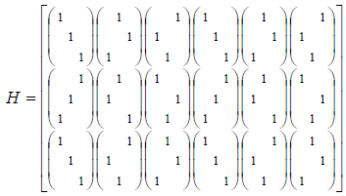
Fig. 1. Example of an unstructured matrix and an LDPC decoder

B. Structured Matrices

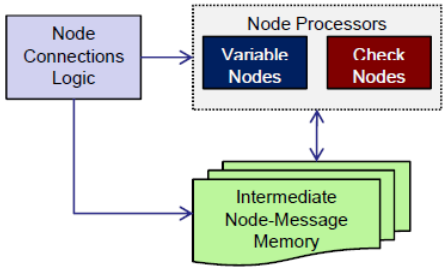
A structured LDPC matrix is constructed in such a way that the position of non-zero elements in the matrix is deterministic. In other words, the organization of the matrix follows a logical pattern. Quasi-Cyclic (QC) based matrices are the most popular class of structured matrices [8]. QC based LDPC matrices are constructed by cyclic shift of the sub-matrices. The sub-matrix is normally an identity matrix. A sample of a QC based structured matrix is shown in Fig. 2 (a). QC based LDPC matrices can be used for fully-parallel decoders, but are highly suitable for partially-parallel decoder architectures [13]. A simple block diagram of a typical partially-parallel decoder architecture using structured matrix is shown in Fig. 2(b). The size of the sub-matrix determines the parallelism factor of the partially-parallel decoder architecture. That is, the number of check nodes and variable nodes required to process in parallel. Since the non-zero elements in the QC matrix follow a logical pattern, the addressing scheme for handling intermediate node messages can be implemented logically. It eliminates the memory requirement to store the inter-connect information in case of an unstructured random LDPC matrix. Hence, the partitioning of the QC matrix using sub-matrices and the logical organization of the non-zero elements substantially alleviates the complexity of the partially-parallel LDPC decoder.

Although structured QC matrices provide a number of advantages for implementing an LDPC decoder, the decoding performances is limited when compared to unstructured random matrices [8]. This is due to the limitation in constructing QC matrices with large girth. However, the performance can be improved by using PEG based QC matrix [14] and Hierarchical QC (HQC) LDPC matrices [9].

The HQC based technique provides greater flexibility for partitioning the matrix into sub-matrices when compared to PEG based QC method. Also the former has completely deterministic pattern of non-zero elements arranged in the LDPC matrix. A 2-Level HQC (2L-HQC) matrix consists of a core matrix expanded by two matrices forming a 2-Level hierarchical structure. Construction of a 2L-HQC matrix is presented in [9].



(a) Sample of a structured LDPC matrix



(b) A simple block diagram of a partially-parallel architecture using structured matrix

Fig. 2. Example of a structured matrix and an LDPC decoder

III. PROPOSED 3-LEVEL HQC LDPC MATRIX

QC based techniques are less flexible for constructing matrices of variable sizes, when compared to unstructured matrices. This limitation is due to the use of array of sub-matrices that are fixed in size. The proposed technique is flexible for constructing matrices by exploiting the advantages of using HQC methods. It uses 3-Level hierarchy to efficiently organize the structure and construct flexible matrices with variable code lengths/rates. Also, Permuted sub-matrices are inserted in layers of the LDPC matrix. This introduces a sort of virtual randomness in the matrix, similar to that of unstructured matrices, to improve the decoding performance. The following sub-sections present a detailed explanation on the construction and analysis of the proposed technique:

A. *Construction of the Matrix*

In order to illustrate the matrix construction process, a ½ rate (3, 6) regular LDPC matrix is considered in this example. A simple structure of the proposed 3L-HQC matrix is shown in Fig. 3.

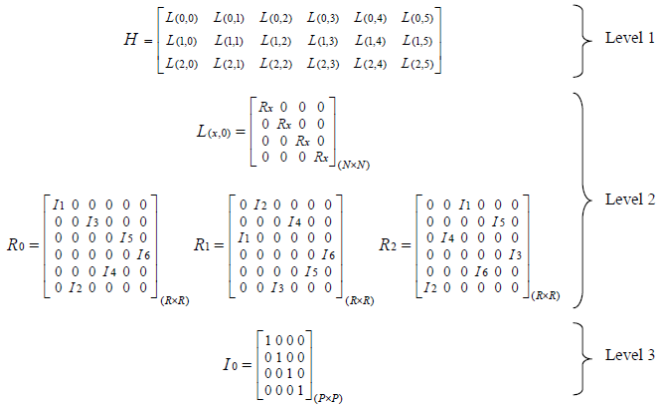


Fig. 3. Illustration for constructing the proposed 3L-HQC matrix

Level-1: The proposed matrix has 3-Levels of hierarchy. The first level of matrix in the hierarchy is termed as the Core matrix. This level is responsible for maintaining the rate and regularity of the LDPC matrix. For example, in case of ½ rate (3, 6) regular LDPC code configuration, the Core matrix (H) consists of 3 rows and 6 columns (see Fig. 3). Further down the matrix construction process, each of the elements in the Core matrix that are expanded maintains a regularity of (1, 1). This retains the overall regularity of (3, 6) in the LDPC matrix.

Level-2: The second level of the matrix is obtained by expanding each of the elements in the Core matrix with a circularly shifted identity matrix (L) of size 'N'. This matrix (L) is again expanded by placing an array of circularly shifted Permuted matrices (R_x) of size 'R'. A Permuted matrix is constructed by placing a positive integer value randomly in the matrix. Examples of integer values are shown as subscript of 'I' in Fig. 3. This level of the matrix structure is predominantly responsible for expansion and construction of LDPC matrices with various code lengths for a particular application.

Note that different combinations of Permuted matrices are used in layers (each rows of Core matrix) of the LDPC matrix. The subscripts in each of the elements in the Core matrix (H) illustrate the layering of the Permuted matrix. For example, a subscript of (x, y) indicate that an 'x^{th}' combination of Permuted matrix is used for expansion of that particular element in the Core matrix with a circular shift of 'y'.

Level-3: In the third level, each of the non-zero elements in the Permuted matrix is expanded by a Base matrix (I). This matrix is a circularly shifted identity matrix of size 'P'. The number of circular shifts in a Base matrix depends on the elements in the Permuted matrix. This is indicated by the subscript of 'I' in the Permuted matrix, as shown in Fig. 3. The size of the Base matrix defines the parallelism factor (P) of the LDPC decoder. That is, the number of check nodes and variable nodes required for parallel processing.

B. *Configurations of the Matrix*

The proposed technique can be configured to generate LDPC matrices with different code lengths by varying 'N', 'R' and 'P' parameters. Some of the possible configurations that are suitable for WiMAX [15], WLAN [16] and DVB-S2 [17] applications are shown in Table I.

TABLE I
CONFIGURATIONS OF THE PROPOSED MATRIX FOR VARIOUS APPLICATIONS

WiMAX (P=16)				WLAN (P=18)				DVB-S2 (P=27)			
CL	CR	R	N	CL	CR	R	N	CL	CR	R	N
576	1/2	6	1	648	1/2	6	1	16200	1/3	5	20
672	1/2	7	1	1296	1/2	6	2	16200	2/3	5	20
768	1/2	8	1	1944	1/2	6	3	64800	1/2	8	50
864	1/2	9	1	648	2/3	6	1	64800	1/3	8	50
960	1/2	10	1	1296	2/3	6	2	64800	2/3	8	50
1056	1/2	11	1	1944	2/3	6	3	64800	5/6	8	50
1152	1/2	6	2	648	5/6	6	1	-	-	-	-
1728	1/2	6	3	1296	5/6	6	2	-	-	-	-
2304	1/2	6	4	1944	5/6	6	3	-	-	-	-

Note: CL = Code Length; CR = Code Rate;

Note that a number of decoders have been proposed that uses a flexible multi-rate and multi-length LDPC matrix [10, 18-20]. However, the proposed matrix is much flexible for constructing LDPC matrices for multiple applications (as shown in Table I) without compromising the decoding performance.

C. *Performance Analysis*

To analyze the decoding performance of the proposed matrix (3L-HQC), simulations were carried out and compared against 2L-HQC and PEG based matrices. A software simulation model was developed using C programs and executed in the MatLab environment [21]. A ½ rate (3, 6) regular 1152-bit LDPC code (WiMAX) using Min-Sum algorithm with 4-bit quantization was used to assess the BER and average iterations for different matrices. For the

simulations, the encoded data is assumed to have Binary Phase Shift Keying (BPSK) modulated and passed over an Additive White Gaussian Noise (AWGN) channel. The maximum number of iterations for the algorithm was set to 10.

The BER performance and average iterations obtained from simulations are shown in Fig. 4 and Fig. 5 respectively. From Fig. 4, it is clear that the proposed matrix outperforms the 2L-HQC by 0.3 dB at a BER of 10^{-5}. The PEG based random matrix has a marginal performance gain of less than 0.1 dB over the proposed matrix at a BER of 10^{-6}. However, in the case of average iterations, each of the matrices have difference of approximately 0.25 iterations over 2.25-3.25 dB E_b/N_o range as shown in Fig. 5. The proposed matrix requires fewer average iterations compared to 2L-HQC.

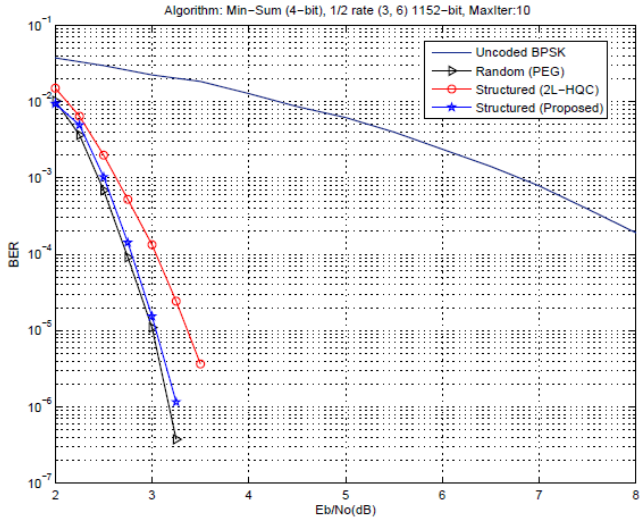


Fig. 4. Simulation of BER performance for various matrices

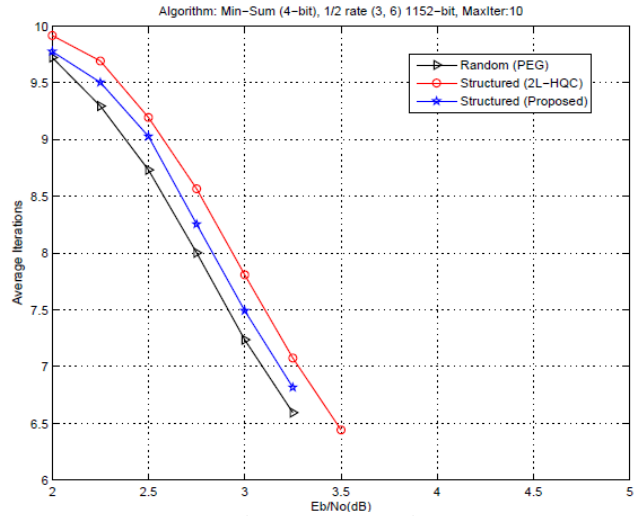


Fig. 5. Simulation of average iterations for various matrices

IV. Design and Analysis of Hardware Model

A. *Hardware Design of the Decoder*

A prototype hardware model of an LDPC decoder using the proposed matrix has been designed using Verilog Hardware Description Language (HDL). In order to verify the feasibility of hardware implementation of the proposed matrix, the decoder model uses simple and straight-forward partially-parallel architecture.

The decoder model is designed for a ½ rate (3, 6) regular LDPC code with code lengths 576, 1152 and 2304, compliant with WiMAX application (see Table I). A Modified Min-Sum (MMS) algorithm is used to reduce the complexity of the decoder [3]. This algorithm uses 4-bit intrinsic messages and 2-bit extrinsic messages with simplified check nodes. Hence reduces the hardware and memory requirements of the decoder.

The top-level simple block diagram of the prototyped hardware model of the LDPC decoder is shown in Fig. 6. The decoder consists of two major blocks: Decode Controller (DC) and Decode Processor (DP). The DC is responsible for controlling the decoding process and responding to external control signals. It also organizes and sequences the input data for decoding and to output the decoded data. The DP is responsible of the decoding process. It consists of Variable Node Processing Unit (VNPU), Check Node Processing Unit (CNPU), Variable Nodes (VN), Check Nodes (CN), Intrinsic Message Block (IMB) and the Permuted Matrix Memory Block (PMMB). Based on parallelism factor (P) for this configuration of the decoder, the VN and CN blocks consist of chain of 16 variable nodes and check nodes respectively (see Table I). The Permuted matrix information is stored in the form of Look-Up Tables (LUT) in PMMB. The VNPU and CNPU use these LUTs for accessing and storing messages at appropriate locations in the Block RAMs (BRAM). To start with the decoding process, the VNPU first accesses the intermediate message decoding data (extrinsic messages) from the BRAM and passes on to the VN. The VN process this data along with the intrinsic message from IMB. The updated message is then passed to CNPU to stored back in the BRAM. This cycle continues till all the variable nodes are processed for the entire code length of the decoder. Next, a similar message updating process is performed by CNPU and CN. This processing cycle of VNPU and CNPU completes a single decoding iteration of the decoder. The decoding process is stopped by DC when the maximum iteration is reached or the parity check is satisfied.

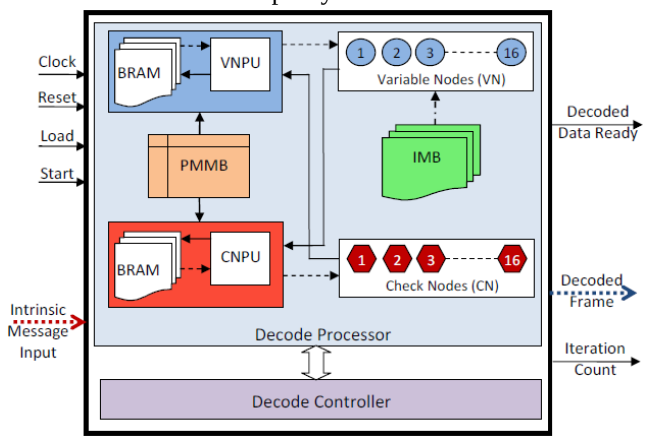


Fig. 6. Top level block diagram of the prototyped LDPC decoder

B. *Analysis of Implementation Results*

The hardware model of the decoder has been implemented on Xilinx Virtex 5 FPGA (XC5VLX110T-3FF1136). Synthesis and implementation of the design was carried out using Xilinx ISE tool. The BER performance and average iterations of the implemented decoder is shown in Fig. 7 and Fig. 8 respectively. As expected, the BER performance improves and average iterations increases as the code length of the decoder increases.

A comparison of hardware requirements and performance of the proposed decoder with that of other partially-parallel decoders are shown in Table II. Among many partially-parallel decoder architectures [13, 22, 23], only a selected few decoders that have configuration close to the proposed decoder is compared. Also the comparison is mainly to highlight the importance of using the proposed matrix.

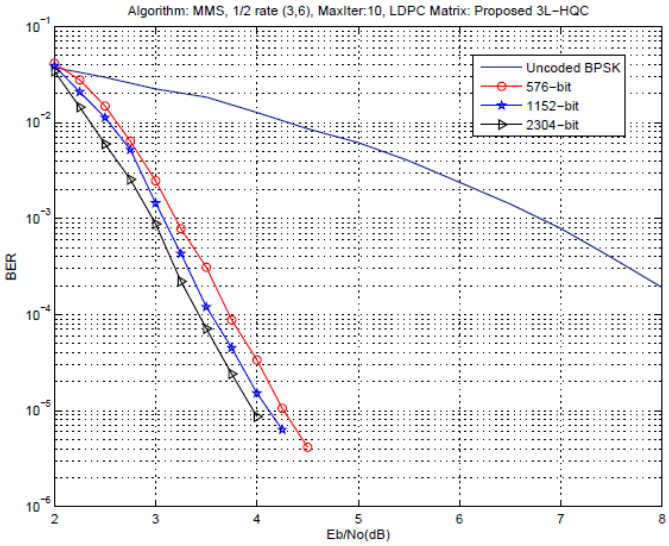


Fig. 7. BER performance of the implemented LDPC decoder

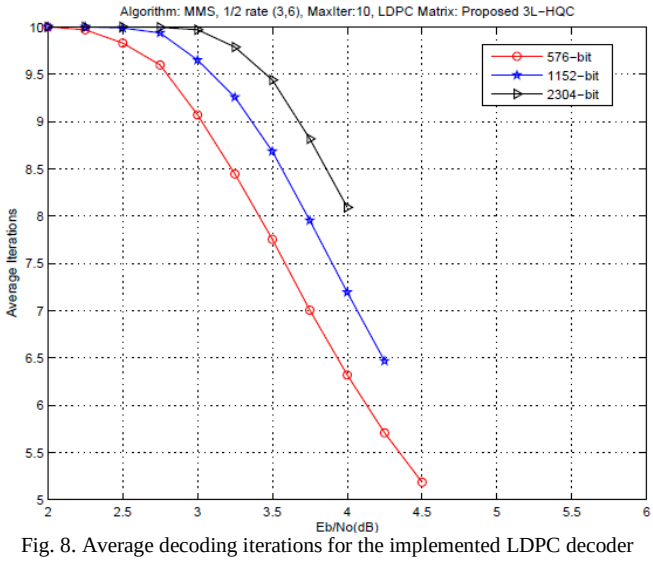


Fig. 8. Average decoding iterations for the implemented LDPC decoder

From Table II, it is clear that using the proposed matrix results in a highly flexible decoder. The logic resources (LUTs and Registers) remain constant irrespective of the LDPC code length. The number of block RAMs are also unaltered. However the memory requirement of the Block RAM increases with an increase in code length.

The Throughput (T) of the implemented decoder is computed using the formula shown in (1). The parallelism factor (P) of the implemented LDPC decoder is 16 and the code lengths are 576, 1152 and 2304. Therefore, the total number of clock cycles (N_{it}) required for processing a decoding iteration is the sum of clock cycles required for processing variable nodes, check nodes and the latency of the decoder. The latency for this architecture of the decoder is 6 and always constant for each of the decoding iteration. For a 576-bit decoder, N_{it} = 60 clock cycles (36+18+6). Similarly, for 1152 and 2304, N_{it} is 114 and 222 respectively.

$$T = \frac{CodeRate \times CodeLength \times MaxOperatingFrequency}{DecodingIterations \times Nit} \quad (1)$$

At a maximum operating frequency of 64 MHz (obtained from the implementation results), the average throughput of the decoders (576, 1152 and 2304) using average iterations (see Fig. 8) at 4.5 dB, 4.25 dB and 4 dB E_b/N_o are shown in Table II. The throughput of the proposed decoder is easily scalable by increasing the parallelism factor (P). However, this results in an increase in the corresponding hardware and memory requirements.

TABLE II
COMPARISON OF FPGA IMPLEMENTATION RESULTS

	Proposed			[24]	[25]
Algorithm	MMS			Min-Sum	Min-Sum
Code Structure	3L-HQC			QC	PEG
Code Length	576	1152	2304	2304	2304
FPGA Device	Virtex 2			Virtex 2	Stratix 2
Slices/ALUTs	2778			6568	17259
4-input LUTs	5188			11028	-
Registers	857			6330	6598
Block RAMs	29			100	-
Total Memory	9,792	19,584	39,168	60288	271,104
Avg. Throughput (Mbps)	59	50	42	61	232.5

V. CONCLUSION

This paper presents a novel technique to construct LDPC matrices with different code lengths suitable for multiple applications. The technique exploits the flexibility of matrix construction using Hierarchical Quasi-Cyclic (HQC) based approach. It is shown that using multi-level hierarchy and layered permutation leads to (1) flexibility in code construction, (2) decoding performance close to Progressive Edge Growth (PEG) based matrices, (3) reduced hardware implementation complexity, (4) better controllability of parallelism factor and (5) scalable throughput. These

advantages of the proposed matrix have been verified by implementing a prototype model of simple partially-parallel decoder architecture. Therefore, the proposed technique provides a potential solution for implementing a highly flexible multi-application LDPC decoder.

REFERENCES

[1] D.J.C. MacKay and R.M. Neal, "Near Shannon limit performance of low density parity check codes," *Electronics Letters*, vol. 33, no. 6, pp. 457-458, 13 March 1997.

[2] G.L.L. Nicolas Fau, *LDPC (Low Density Parity Check) - A Better Coding Scheme for Wireless PHY Layers* Design and Reuse Industry Article, 2008.

[3] V.A. Chandrasetty and S.M. Aziz, "FPGA Implementation of High Performance LDPC Decoder Using Modified 2-Bit Min-Sum Algorithm," *Proceedings of the 2nd International Conference on Computer Research and Development*, Kuala Lumpur, pp. 881-885, 7-10 May 2010.

[4] M. Karkooti and J.R. Cavallaro, "Semi-parallel reconfigurable architectures for real-time LDPC decoding," *Proceedings of the International Conference on Information Technology: Coding and Computing*, pp. 579-585, 5-7 April 2004.

[5] E. Liao, Y. Engling, and B. Nikolic, "Low-density parity-check code constructions for hardware implementation," *Proceedings of the IEEE International Conference on Communications*, pp. 2573-2577, 20-24 June 2004.

[6] R. Zarubica and S.G. Wilson, "A solution for memory collision in semi-parallel FPGA-based LDPC decoder design," *Proceedings of the 41st Asilomar Conference on Signals, Systems and Computers*, Pacific Grove, CA, pp. 982-986, 4-7 November 2007.

[7] Y. Kou, S. Lin, and M.P.C. Fossorier, "Low density parity check codes: construction based on finite geometries," *Proceedings of the IEEE Global Telecommunications Conference*, San Francisco, CA, pp. 825-829, 27 November-01 December 2000.

[8] Y. Xiao and M.H. Lee, "Construction of good quasi-cyclic LDPC codes," *Proceedings of the Wireless, Mobile and Multimedia Networks, 2006 IET International Conference on*, Hangzhou, China, pp. 1-4, 6-9 November 2006.

[9] C. Yi-Hsing and K. Mong-Kai, "A High Throughput H-QC LDPC Decoder," *Proceedings of the IEEE International Symposium on Circuits and Systems*, New Orleans, LA, pp. 1649-1652, 27-30 May 2007.

[10] N. Bonello, S. Chen, and L. Hanzo, "Multilevel Structured Low-Density Parity-Check Codes for AWGN and Rayleigh Channels," *Proceedings of the IEEE International Conference on Communications*, Beijing, pp. 485-489, 19-23 May 2008.

[11] H. Xiao-Yu, E. Eleftheriou, and D.M. Arnold, "Progressive edge-growth Tanner graphs," *Proceedings of the IEEE Global Telecommunications Conference*, San Antonio, TX, pp. 995-1001, 25-29 November 2001.

[12] X. Hua and A.H. Banihashemi, "Improved progressive-edge-growth (PEG) construction of irregular LDPC codes," *IEEE Communications Letters*, vol. 8, no. 12, pp. 715-717, December 2004.

[13] X. Zhang and F. Cai, "Partial-parallel decoder architecture for quasi-cyclic non-binary LDPC codes," *Proceedings of the International Conference on Acoustics Speech and Signal Processing*, pp. 1506-1509, 14-19 March 2010.

[14] L. Zongwang and B.V.K.V. Kumar, "A class of good quasi-cyclic low-density parity check codes based on progressive edge growth graph," *Proceedings of the 38th Asilomar Conference on Signals, Systems and Computers*, pp. 1990-1994, 7-10 November 2004.

[15] IEEE Stdandard 802.l6e, "Air interface for fixed and mobile broadband wireless access systems . Amendment 2: Physical and medium access control layers for combined fixed and mobile operation in licensed bands", IEEE, December 2005.

[16] IEEE Std. 802.11n, "Wireless LAN medium access control (MAC) and physicallayer (PHY) specifications: enhancementsfor higher throughput", IEEE, September 2009.

[17] Europian Standard DVB-S2, "Digital Video Broadcasting (DVB); Second generation framing structure, channel coding and modulation systems for Broadcasting, Interactive Services, News Gathering and other broadband satellite applications (DVB-S2)", European Broadcasting Union, August 2009.

[18] L. Yang, M. Shen, H. Liu, and C.-J.R. Shi, "An FPGA implementation of low-density parity-check code decoder with multi-rate capability," *Proceedings of the conference on Asia South Pacific design automation*, Shanghai, China, pp. 760-763, 18-21 Jan. 2005.

[19] Y. Lei, L. Hui, and C.J.R. Shi, "Code construction and FPGA implementation of a low-error-floor multi-rate low-density Parity-check code decoder," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 53, no. 4, pp. 892-904, April 2006.

[20] F. Charot, C. Wolinski, N. Fau, and F. Hamon, "A New Powerful Scalable Generic Multi-Standard LDPC Decoder Architecture," *Proceedings of the 16th International Symposium on Field-Programmable Custom Computing Machines*, Palo Alto, CA, pp. 314-315, 14-15 April 2008.

[21] V.A. Chandrasetty and S.M. Aziz, "A reduced complexity message passing algorithm with improved performance for LDPC decoding," *Proceedings of the 12th International Conference on Computers and Information Technology*, Dhaka, pp. 19-24, 21-23 December 2009.

[22] D. Yongmei, C. Ning, and Y. Zhiyuan, "Memory Efficient Decoder Architectures for Quasi-Cyclic LDPC Codes," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 55, no. 9, pp. 2898-2911, October 2008.

[23] B. Dan, et al., "Programmable Architecture for Flexi-Mode QC-LDPC Decoder Supporting Wireless LAN/MAN Applications and Beyond," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 57, no. 1, pp. 125-138, January 2010.

[24] K.K. Gunnam, G.S. Choi, W. Weihuang, K. Euncheol, and M.B. Yeary, "Decoding of Quasi-cyclic LDPC Codes Using an On-the-Fly Computation," *Proceedings of the 40th Asilomar Conference on Signals, Systems and Computers*, Pacific Grove, CA, pp. 1192-1199, 29 October-1 November 2006.

[25] H. Ding, S. Yang, W. Luo, and M. Dong, "Design and Implementation for High Speed LDPC Decoder with Layered Decoding," *Proceedings of the WRI International Conference on Communications and Mobile Computing*, Yunnan, pp. 156-160, 6-8 January 2009.