

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/225027141>

# A Highly Flexible LDPC Decoder using Hierarchical Quasi-Cyclic Matrix with Layered Permutation

Article · March 2012

DOI: 10.4304/jjnw.7.3.441-449

---

CITATIONS

13

---

READS

1,116

2 authors:



[Vikram Chandrasetty](#)

Western Digital Corporation

26 PUBLICATIONS 241 CITATIONS

SEE PROFILE



[Syed Mahfuzul Aziz](#)

University of South Australia

133 PUBLICATIONS 2,026 CITATIONS

SEE PROFILE

# A Highly Flexible LDPC Decoder using Hierarchical Quasi-Cyclic Matrix with Layered Permutation

Vikram Arkalgud Chandrasetty and Syed Mahfuzul Aziz

School of Electrical & Information Engineering, University of South Australia,  
Mawson Lakes, SA 5095, Australia

Email: vikramac@ieee.org, mahfuz.aziz@unisa.edu.au

**Abstract**—Hardware implementation of partially-parallel Low-Density Parity-Check (LDPC) decoders using unstructured random matrices is very complex and requires huge hardware resources. To alleviate the complexity and minimize resource requirements, structured LDPC matrices are used. This paper presents a novel technique for constructing a multi-level Hierarchical Quasi-Cyclic (HQC) structured matrix for LDPC decoder. 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 inserted in layers at different levels of matrix hierarchy to provide virtual randomness in the LDPC matrix. Simulation results show that the HQC matrices generated using the proposed technique have a marginal loss of less than 0.1 dB at a bit error rate (BER) performance  $10^{-5}$  compared to unstructured random matrices. A hardware model of the proposed matrix structure has been developed and synthesized on Xilinx FPGA to verify the flexibility features, hardware requirements and to analyze the performance of the LDPC decoder.

**Index Terms**—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) techniques that can achieve bit error rate (BER) performance close to the Shannon Limit [1]. The inherent structure of the LDPC matrix provides high degree of parallelism and flexibility for designing a decoder [2] for various applications such as – Worldwide Interoperability for Microwave Access (WiMAX), Wireless Local Area Network (WLAN) and Digital Video Broadcasting - Satellite - Second Generation (DVB-S2) [3]. A fully-parallel implementation of a LDPC decoder provides very high throughput but requires large amount of hardware resources [4-7]. Also, the complexity of the decoder increases drastically with longer code lengths. An alternative solution is to use more resource efficient partially-parallel architectures [8],

where the number of decoding nodes are much reduced. These reduced nodes are used repeatedly to achieve the full decoding functionality. Unlike a fully-parallel decoder, a partially-parallel decoder utilizes the block memory of an FPGA to store and access intermediate extrinsic messages. The obvious price to pay for the smaller hardware size of a partially-parallel architecture is a reduction in decoder throughput due to reduction in parallelism [9].

Nevertheless partially-parallel decoding architectures enable trade-off between hardware size and throughput. The number of parallel nodes (check nodes and variable nodes) required by the decoder depends on the partition size of the matrix (also known as the base matrix). Also, the complexity of the addressing scheme for storing and reading intermediate messages depends largely on the structure of the LDPC matrix. Therefore, the hardware requirement of a partially-parallel decoder predominantly relies on the structure and complexity of the LDPC matrix [10]. In order to reduce the hardware complexity of the decoder, structured Quasi-Cyclic (QC) [11] based matrix construction methods are widely used. This technique constructs an LDPC matrix by using an array of cyclically-shifted base matrices [12]. The parallelism factor of partially-parallel decoder architecture is normally defined by the size of the base matrix. Hierarchical QC (HQC) [13] matrices are constructed with several levels of sub-matrices, with the lowest level corresponding to the base matrix. HQC based technique has the flexibility for constructing LDPC matrices of variable code lengths and code rates [14]. However, not all QC based matrices lead to decoding performance (BER and average iterations) comparable to that of unstructured matrices [12]. 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 unique 3-Level HQC (3L-HQC) matrix construction with a novel Layered Permutation (LP) technique [15]. The three 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 various applications such as WiMAX, WLAN and DVB-S2. In addition, the proposed matrix consists of a permuted matrix in 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. This technique is explained in detail in Section III. Simulation results show that the proposed matrix has a marginal degradation in BER performance compared to unstructured random matrices. It also outperforms the 2-Level HQC based LDPC decoders [13]. FPGA synthesis of a partially-parallel architecture using the proposed matrix leads to significant reduction in memory requirements.

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 in this section. It is then followed by a partially-parallel decoder synthesis in section IV.

## II. STRUCTURE OF LDPC MATRICES

As stated earlier, the structure of the matrix plays a vital role in determining the performance and implementation complexity of an LDPC decoder, especially for partially-parallel architectures [10]. This section presents an overview and comparison of the structured and unstructured matrices.

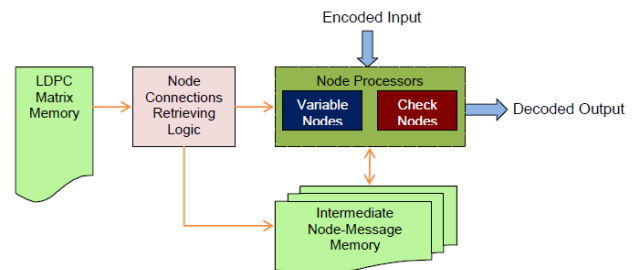
### A. Unstructured Matrices

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 [16]. The algorithms aim to obtain matrices with large cycles in the LDPC matrix (also known as Girth) and construct matrices of virtually any size. Due to the possibility of generating matrices with large girths unstructured matrices exhibit good decoding performance, such as BER [17]. These matrices can be used to design both fully-parallel and partially-parallel LDPC architectures. However, due to the randomness (complex node inter-connections) in the structure of the matrix, the implementation of either architecture is substantially complex, especially for large code lengths. A simple block diagram of typical partially-parallel architecture of a LDPC decoder using unstructured matrix is shown in Fig. 1(b). The decoder consists of few check nodes and variable nodes depending on the parallelism factor. The node interconnect information is stored in a large LDPC matrix memory block. In typical partially-parallel architectures, the intermediate messages need to be stored in separate memory blocks. The writing and reading of these messages at appropriate memory locations is performed based on the LDPC matrix memory. Hence there is a need for an exclusive memory

to store interconnects information for decoders based on unstructured LDPC matrices.

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

(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 a LDPC decoder

Note that an exclusive memory is required to store the inter-connect information in the LDPC matrix for decoding.

### 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 due to their flexibility in designing resource efficient partially-parallel decoder architectures [12]. QC based LDPC matrices are constructed by cyclic shift of the sub-matrices. The sub-matrix is normally an identity matrix for simplicity and ease of LDPC decoder design. 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 due to their breakable structure for parallel processing [18]. A simple block diagram of a typical partially-parallel decoder architecture using structured matrix is shown in Fig. 2(b). Note that the decoder based on structured LDPC matrices eliminates the need for an exclusive memory to store interconnects information. The size of the sub-matrix determines the parallelism factor of the partially-parallel decoder, 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 deterministic pattern the addressing scheme for handling intermediate node messages can be implemented using logic blocks. This

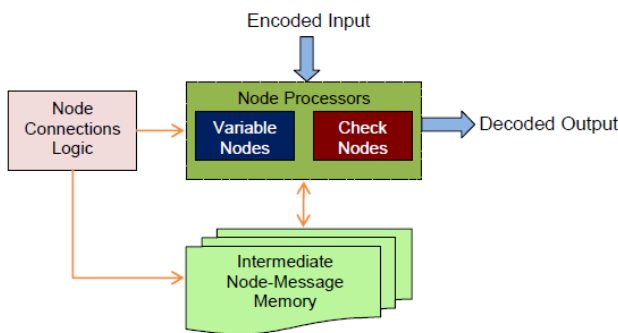
eliminates the need for exclusive inter-connect memory. Hence, the partitioning of the QC matrix using sub-matrices and the logical organization of the non-zero elements alleviates the complexity of the partially-parallel LDPC decoder.

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 [12]. This is due to the limitation in constructing QC matrices with large girth. However, by using PEG based QC matrix [19] and Hierarchical QC (HQC) based LDPC matrices [13] the performance can be improved and are comparable to that of unstructured matrices.

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 that is expanded by two other matrices forming a 2-Level hierarchical structure. Construction of a 2L-HQC matrix is presented in [13].

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

(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 a LDPC decoder

### III. PROPOSED 3-LEVEL HQC LDPC MATRIX: DESIGN AND ANALYSIS

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 HQC methods [13]. It uses a unique 3-

Level hierarchy to efficiently organize the structure and construct flexible matrices with variable code lengths/rates. In addition, permuted sub-matrices are inserted in layers of the LDPC matrix. This novel Layered Permutation technique introduces virtual randomness in the matrix similar to that of unstructured matrices, to improve the decoding performance. The following subsections present a detailed explanation on the construction and analysis of the proposed technique:

#### A. Structure of the Matrix

In order to illustrate the matrix construction process, a  $\frac{1}{2}$  rate (3, 6) regular LDPC matrix is considered. A simple structure of the proposed 3L-HQC matrix with Layered Permutation is shown in Fig. 3.

**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  $\frac{1}{2}$  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'. Unlike that is presented in [14], the 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'.

$$H = \begin{bmatrix} L(0,0) & L(0,1) & L(0,2) & L(0,3) & L(0,4) & L(0,5) \\ L(1,0) & L(1,1) & L(1,2) & L(1,3) & L(1,4) & L(1,5) \\ L(2,0) & L(2,1) & L(2,2) & L(2,3) & L(2,4) & L(2,5) \end{bmatrix} \quad \left. \vphantom{\begin{matrix} L(0,0) \\ L(1,0) \\ L(2,0) \end{matrix}} \right\} \text{Level 1}$$

$$L(x,y) = \begin{bmatrix} R_x & 0 & 0 & 0 \\ 0 & R_x & 0 & 0 \\ 0 & 0 & R_x & 0 \\ 0 & 0 & 0 & R_x \end{bmatrix}_{(N \times N)} \quad \left. \vphantom{\begin{matrix} R_x \\ R_x \\ R_x \\ R_x \end{matrix}} \right\} \text{Level 2}$$

$$R_0 = \begin{bmatrix} I_1 & 0 & 0 & 0 & 0 \\ 0 & 0 & I_3 & 0 & 0 \\ 0 & 0 & 0 & I_5 & 0 \\ 0 & 0 & 0 & 0 & I_6 \\ 0 & 0 & 0 & I_4 & 0 \\ 0 & I_2 & 0 & 0 & 0 \end{bmatrix}_{(R \times R)} \quad R_1 = \begin{bmatrix} 0 & I_2 & 0 & 0 & 0 \\ 0 & 0 & I_4 & 0 & 0 \\ I_1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & I_6 \\ 0 & 0 & 0 & I_5 & 0 \\ 0 & 0 & I_3 & 0 & 0 \end{bmatrix}_{(R \times R)} \quad R_2 = \begin{bmatrix} 0 & 0 & I_1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & I_5 & 0 \\ 0 & I_4 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & I_3 \\ 0 & 0 & 0 & I_6 & 0 & 0 \\ I_2 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}_{(R \times R)}$$

$$I_0 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}_{(P \times P)} \quad \left. \vphantom{\begin{matrix} I_1 \\ I_2 \\ I_3 \\ I_4 \\ I_5 \\ I_6 \end{matrix}} \right\} \text{Level 3}$$

Fig. 3. Simple structure of the proposed 3L-HQC matrix with LP

*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. Construction Technique

The proposed matrix is constructed using a software model in the MatLab environment. The technique used for construction of the matrix is illustrated in Fig. 4. All the key parameters such as code length, code rate and regularity of the expected LDPC matrix are provided to the software model. The model uses one of the possible combinations of matrix parameters (N, R and P) and the permutation of the random matrices (R). The resulting LDPC matrix is verified for girth. Matrices with girth less than or equal to 4 are eliminated due to their poor decoding performance and the matrix is re-constructed by varying the permuted matrix structure and circular shift parameters. Whereas matrices with girth greater than 4 is chosen [20] and simulations are carried out to ascertain the BER, FER and average iteration performance. LDPC matrices with performance close/comparable to that of unstructured matrices is selected and the configuration of the matrix parameters are saved. Matrices with poor performance are rejected and the matrix re-construction procedure is repeated.

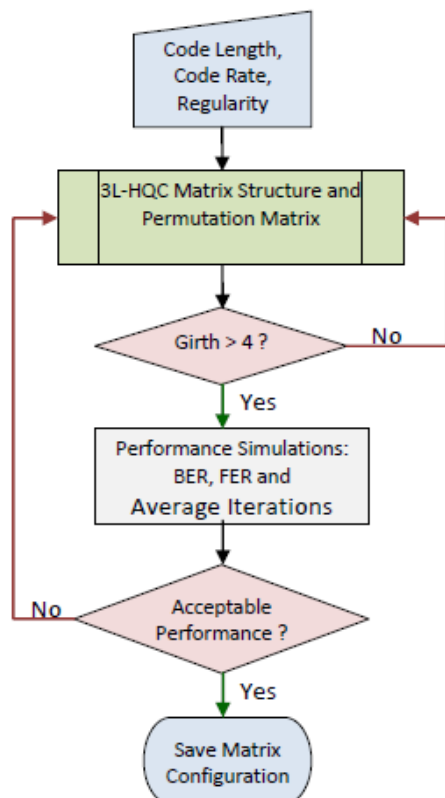


Fig. 4. Illustration of the proposed matrix construction technique

#### C. Standard configurations of the Matrix

The proposed technique allows generation of LDPC matrices with different code lengths by varying 'N', 'R' and 'P' parameters. Some of the possible configurations that are suitable for WiMAX [21], WLAN [22] and DVB-S2 [23] application standards 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	-	-	-	-

CL = Code Length;

CR = Code Rate;

P = Size of the base matrix that determines the parallelism factor

R = Size of the random permuted matrix

N = Size of the identity matrix for expansion of LDPC matrix. The diagonal elements of an identity matrix is replaced by the random permuted matrix

A number of decoders have been proposed that uses a flexible multi-rate and multi-length LDPC matrix [14, 24-26]. However, the proposed matrix is flexible for constructing LDPC matrices with multi-rate and multi-length for multiple applications (as shown in Table I) without compromising the decoding performance. This flexibility is mainly achieved by introducing an additional level (3<sup>rd</sup>) in the hierarchy of QC matrix compared to 2L-HQC.

#### D. Visual Analysis

The matrices constructed using the proposed technique is verified for girth. The matrices are also visually analyzed for distribution of non-zero elements. A 1/2 rate (3, 6) 1296-bit LDPC matrix (for WLAN) constructed using PEG (unstructured random), 2L-HQC and the proposed technique (3L-HQC with LP) is shown in Fig. 5(a), 5(b) and 5(c) respectively.

From Fig. 5(a) it is clear that the PEG based random matrix follows a particular pattern but the position of non-zero elements are not deterministic. However, the HQC based matrices in Fig. 5(b) and Fig. 5(c) exhibit a structured pattern. The proposed 3L-HQC matrix with LP has finer distribution compared to 2L-HQC. This is due to smaller Base matrices in the 3-level hierarchical structure. Note that the Layered Permutation feature of the proposed matrix is not very clearly visible in the Fig. 5(c). However, the decoder performance gains achieved from this feature is discussed in the next section.



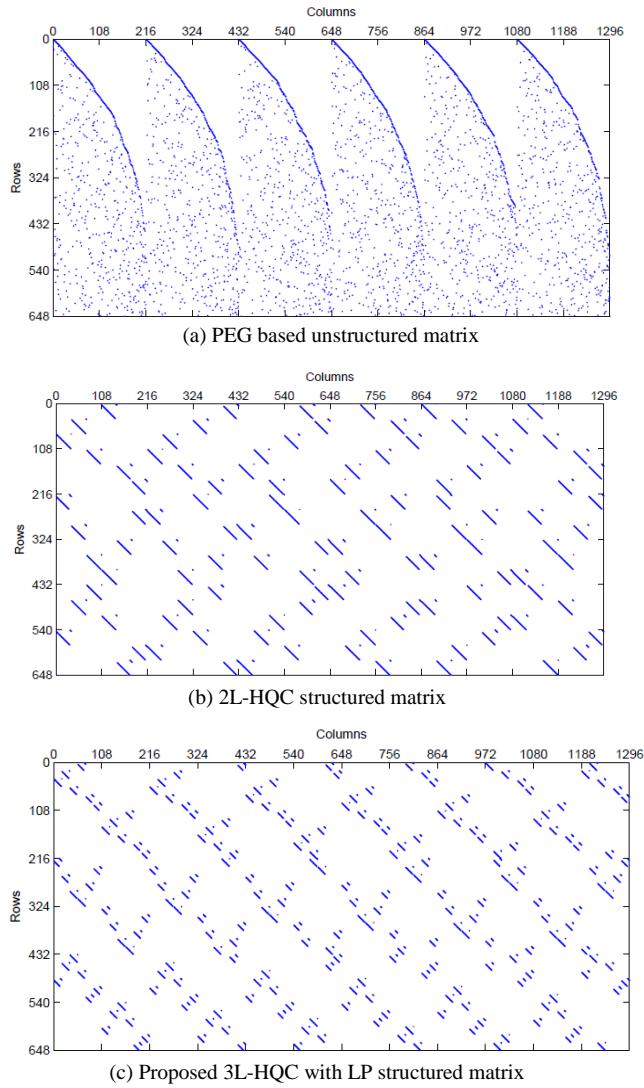


Fig. 5. Visual analysis of distribution of non-zero elements in the matrix

### E. Performance Analysis

To analyze the decoding performance of the proposed matrix, 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 [2]. A  $\frac{1}{2}$  rate (3, 6) regular 1296-bit LDPC code (WLAN) using Min-Sum algorithm with 4-bit quantization was used to assess the BER, frame error rate (FER) and average iterations. For the simulations, the encoded data was assumed to be Binary Phase Shift Keying (BPSK) modulated and passed over an Additive White Gaussian Noise (AWGN) channel. The maximum number of iterations for the algorithms was set to 10.

The BER and FER performance along with the average iterations are shown in Fig. 6, Fig. 7 and Fig. 8 respectively. The figures include performances of 3L-HQC matrices with and without Layered Permutation.

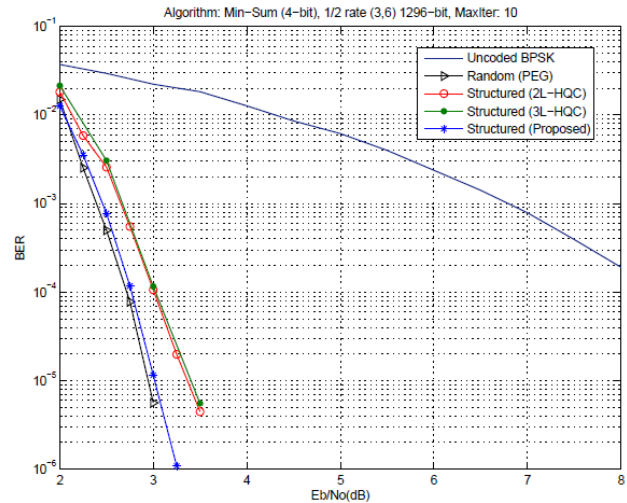


Fig. 6. Simulation of BER performance for various matrices

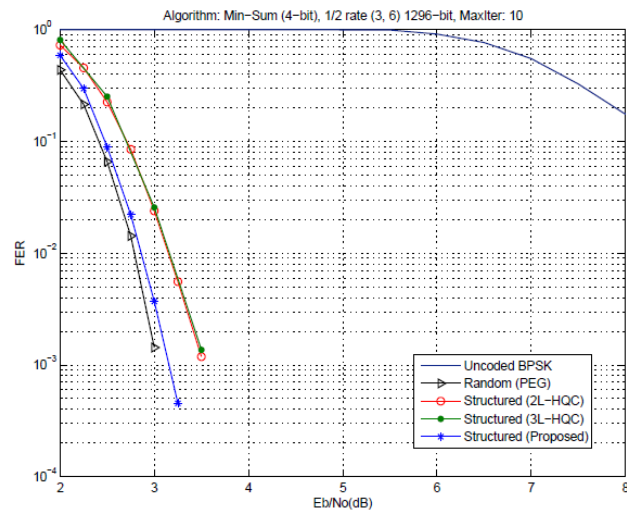


Fig. 7. Simulation of FER performance for various matrices

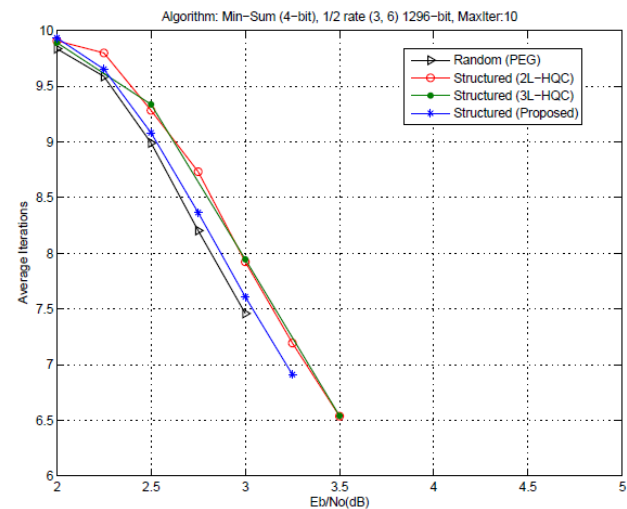


Fig. 8. Simulation of average iterations for various matrices

From Fig. 6-8, the 3L-HQC matrix has performance similar to that of 2L-HQC. It is clear that introducing an additional level (3<sup>rd</sup>) in the QC LDPC matrix hierarchy has no impact on decoding performance. As stated earlier in Section III (C), the 3L-HQC provides better flexibility in generating LDPC matrices with various configurations

for different applications compared to 2L-HQC. However, the performance improvement is achieved by using the Layered Permutation technique. The proposed matrix (3L-HQC with LP) outperforms the 2L-HQC and 3L-HQC by 0.3 dB at a BER of  $10^{-5}$  and  $\sim 0.35$  dB at a FER of  $10^{-3}$  (approx.) respectively. The PEG based random matrix has a marginal performance gain of less than 0.1 dB over the proposed matrix at a BER of  $10^{-5}$  and FER of  $10^{-3}$  (approx.). However, in the case of average iterations, the proposed matrix and 2L-HQC/3L-HQC requires 7.6 and 6.8 iterations to achieve BER performance of  $10^{-5}$  at 3 dB and 3.4 dB respectively as shown in Fig. 6 and Fig. 8. The proposed matrix requires fewer average iterations compared to 2L-HQC/3L-HQC for a given signal-to-noise ratio (SNR) range. From the simulation results it is clear that using Layered Permutation technique improves the decoding performance over 2L-HQC/3L-HQC.

#### IV. DECODER ARCHITECTURE DESIGN AND ANALYSIS

##### A. Hardware Design of the Decoder

A Register Transfer Level (RTL) model of an LDPC decoder using the proposed 3L-HQC matrix with LP has been developed using Verilog Hardware Description Language (HDL). In order to verify the feasibility of the decoder's hardware implementation, the Verilog model uses simple partially-parallel architecture. The model is designed for a  $1/2$  rate (3, 6) regular LDPC code with code lengths 648, 1296 and 1944, compliant with WLAN (IEEE Std. 802.11n) application (see Table I). Min-Sum algorithm with 4-bit quantization of messages is used to obtain optimum BER performance of the decoder [27] and also to efficiently utilize the memory and the logic blocks in the FPGA.

The top-level block diagram of the hardware model of the LDPC decoder is shown in Fig. 9.

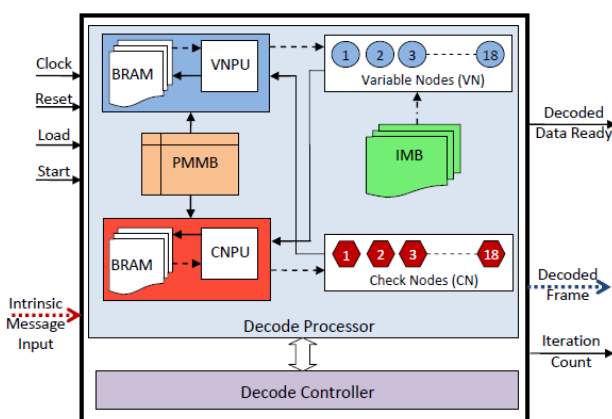


Fig. 9. Top level block diagram of the LDPC decoder

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 and outputs the decoded data. The DP is responsible for 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 the parallelism factor (P) for this configuration of the decoder, the VN and CN blocks consist of chain of 18 variable nodes and 18 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 the decoding process, the VNPU first accesses the intermediate messages (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 for storing in the BRAM. This cycle continues until all the variable nodes are processed for the entire code length. 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 count is reached or the parity check is satisfied.

##### B. Analysis of Synthesis Results

The hardware model of the decoder has been designed for Xilinx Virtex 2 FPGA (XC2V8000-5FF152). Synthesis, placement and routing (PAR) of the design was carried out using Xilinx ISE tool. The BER, FER performance and average iterations of the implemented decoder is shown in Fig. 10, Fig. 11 and Fig. 12 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 those of other partially-parallel decoders are shown in Table II. Among the many partially-parallel decoder architectures reported to date [18, 28-32], the ones that have configuration close to the proposed decoder are compared [30-32]. The comparison is mainly to highlight the benefits of using the proposed matrix.

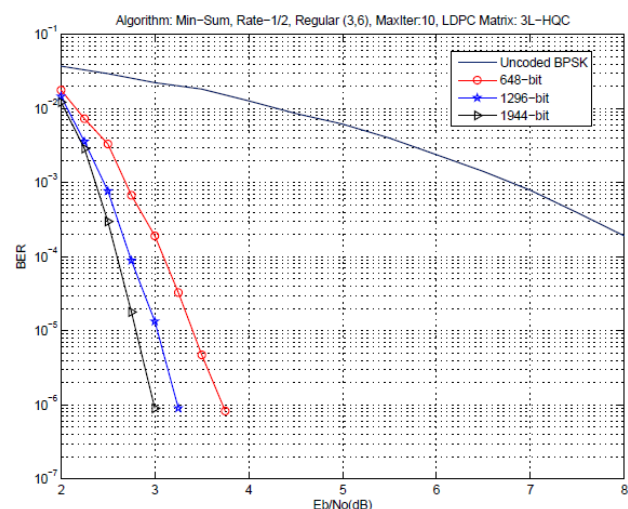


Fig. 10. BER performance of the proposed LDPC decoder

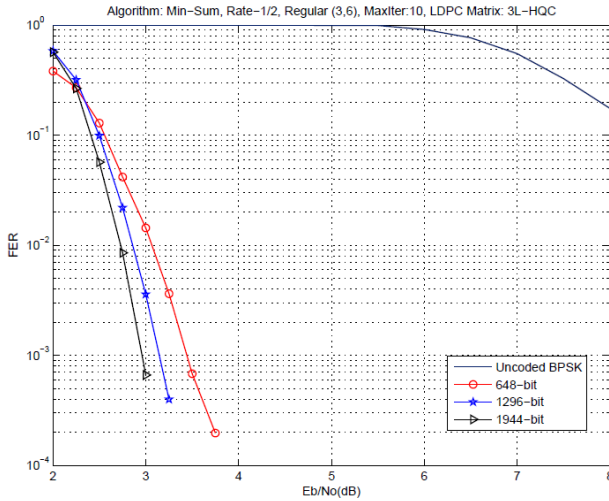


Fig. 11. FER performance of the proposed LDPC decoder

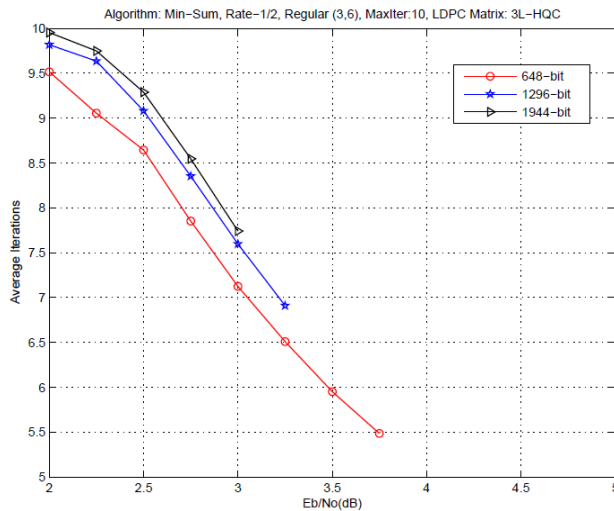


Fig. 12. Average decoding iterations for the proposed LDPC decoder

TABLE II.  
COMPARISON OF FPGA IMPLEMENTATION RESULTS

	Proposed			[30]	[31]	[32]
Algorithm	Min-Sum					
Architecture	Partially-Parallel					
Code Structure	3L-HQC			QC	QC	PEG
Code Length	648	1296	1944	1944	1536	2304
FPGA Device	Virtex2					Stratix2
Slices/ALUTs	5908			5514	11352	17259
4-input LUTs	10816			9352	20374	NA
Registers	1604			5341	NA	6598
Block RAMs	47			133	66	NA
Total Memory (bits)	18792	37584	56376	55344	NA	271104
Avg. Throughput (Mbps)	54	45	32	119	127	232.5

NA = Not Available

From Table II, it is clear that the proposed matrix results in a highly flexible decoder with variable code lengths. 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 proposed decoder is computed using the formula shown in (1). The parallelism factor (P) of the implemented LDPC decoder is 18 and the code lengths are 648, 1296 and 1944. 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 decoder is 6 clock cycles for accessing and storing processed data from the nodes to the memory at the end of each of the decoding iterations. For a 648-bit decoder,  $N_{it} = 60$  clock cycles ( $36+18+6$ ). Similarly for 1296 and 1944,  $N_{it}$  is 114 and 222 respectively.

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

At a maximum operating frequency of 128 MHz (obtained after synthesis, placement and routing), the average throughput of the decoders (648, 1296 and 1944) using average iterations (see Fig. 12) at 3.75 dB, 3.25 dB and 3 dB  $E_b/N_0$  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.

## 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 inherent in the Hierarchical Quasi-Cyclic (HQC) based approach. It is shown that using multi-level structure in the matrix leads to flexibility in code construction for various application standards and better controllability of parallelism factor. Also, the unique Layered Permutation technique used in the proposed matrix helps in significant improvement in decoding performance close to Progressive Edge Growth (PEG) based matrices. These advantages of the proposed matrix have been verified through performance simulations and comparing with various LDPC matrix structures. The flexibility of the decoder (supporting multiple code lengths, throughput and hardware requirements) is also verified by designing a prototype hardware model of partially-parallel architecture suitable for IEEE Std. 802.11n (WLAN). The throughput of the decoder is easily scalable by selecting appropriate parallelism factor. Therefore, the proposed technique provides an attractive solution for implementing a highly flexible partially-parallel 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] V.A. Chandrasetty and S.M. Aziz, "A reduced complexity message passing algorithm with improved performance for LDPC decoding," *Proc. of the 12th International Conference on Computers and Information Technology*, Dhaka, pp. 19-24, 21-23 December 2009.
- [3] G.L.L. Nicolas Fau, *LDPC (Low Density Parity Check) - A Better Coding Scheme for Wireless PHY Layers* EE Times Network 2008.
- [4] S.S. Tehrani, S. Mannor, and W.J. Gross, "An Area-Efficient FPGA-Based Architecture for Fully-Parallel Stochastic LDPC Decoding," *Proc. of the IEEE Workshop on Signal Processing Systems*, Shanghai, China, pp. 255-260, 17-19 October 2007.
- [5] V.A. Chandrasetty and S.M. Aziz, "FPGA Implementation of a LDPC Decoder using a Reduced Complexity Message Passing Algorithm," *Journal of Networks*, Academy Publisher, vol. 6, no. 1, pp. 36-45, January 2011.
- [6] V.A. Chandrasetty and S.M. Aziz, "Analysis of Performance and Implementation Complexity of Simplified Algorithms for Decoding Low-Density Parity-Check codes," *Proc. of the IEEE Globecom Workshop on Complex and Communication Networks*, Miami, USA, pp. 445-450, 6-10 December 2010.
- [7] C. Zhixiang, et al., "A macro-layer level fully parallel layered LDPC decoder SOC for IEEE 802.15.3c application," *Proc. of the International Symposium on VLSI Design, Automation and Test*, Hsinchu, pp. 1-4, 25-28 April 2011.
- [8] K. Shimizu, T. Ishikawa, N. Togawa, T. Ikenaga, and S. Goto, "Partially-parallel LDPC decoder based on high-efficiency message-passing algorithm," *Proc. of the IEEE International Conference on Computer Design: VLSI in Computers and Processors*, pp. 503-510, 2-5 October 2005.
- [9] E. Liao, Y. Engling, and B. Nikolic, "Low-density parity-check code constructions for hardware implementation," *Proc. of the IEEE International Conference on Communications*, pp. 2573-2577, 20-24 June 2004.
- [10] R. Zarubica and S.G. Wilson, "A solution for memory collision in semi-parallel FPGA-based LDPC decoder design," *Proc. of the 41st Asilomar Conference on Signals, Systems and Computers*, Pacific Grove, CA, pp. 982-986, 4-7 November 2007.
- [11] Y. Kou, S. Lin, and M.P.C. Fossorier, "Low density parity check codes: construction based on finite geometries," *Proc. of the IEEE Global Telecommunications Conference*, San Francisco, CA, pp. 825-829, 27 November-01 December 2000.
- [12] Y. Xiao and M.H. Lee, "Construction of good quasi-cyclic LDPC codes," *Proc. of the Wireless, Mobile and Multimedia Networks, 2006 IET International Conference on*, Hangzhou, China, pp. 1-4, 6-9 November 2006.
- [13] C. Yi-Hsing and K. Mong-Kai, "A High Throughput H-QC LDPC Decoder," *Proc. of the IEEE International Symposium on Circuits and Systems*, New Orleans, LA, pp. 1649-1652, 27-30 May 2007.
- [14] N. Bonello, S. Chen, and L. Hanzo, "Multilevel Structured Low-Density Parity-Check Codes for AWGN and Rayleigh Channels," *Proc. of the IEEE International Conference on Communications*, Beijing, pp. 485-489, 19-23 May 2008.
- [15] V.A. Chandrasetty and S.M. Aziz, "Construction of a Multi-Level Hierarchical Quasi-Cyclic Matrix with Layered Permutation for Partially-Parallel LDPC Decoders," *Proc. of the 13th International Conference on Computers and Information Technology*, Dhaka, pp. 131-136, 23-25 December 2010.
- [16] H. Xiao-Yu, E. Eleftheriou, and D.M. Arnold, "Progressive edge-growth Tanner graphs," *Proc. of the IEEE Global Telecommunications Conference*, San Antonio, TX, pp. 995-1001, 25-29 November 2001.
- [17] 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.
- [18] X. Zhang and F. Cai, "Partial-parallel decoder architecture for quasi-cyclic non-binary LDPC codes," *Proc. of the International Conference on Acoustics Speech and Signal Processing*, pp. 1506-1509, 14-19 March 2010.
- [19] 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," *Proc. of the 38th Asilomar Conference on Signals, Systems and Computers*, pp. 1990-1994, 7-10 November 2004.
- [20] W. Xiaofu, Y. Xiaohu, and Z. Chunming, "A necessary and sufficient condition for determining the girth of quasi-cyclic LDPC codes," *IEEE Transactions on Communications*, vol. 56, no. 6, pp. 854-857, June 2008.
- [21] IEEE Standard 802.16e, "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.
- [22] IEEE Standard 802.11n, "Wireless LAN medium access control (MAC) and physical layer (PHY) specifications: enhancements for higher throughput", IEEE, September 2009.
- [23] European 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.
- [24] 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," *Proc. of the conference on Asia South Pacific design automation*, Shanghai, China, pp. 760-763, 18-21 January 2005.
- [25] 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.
- [26] F. Charot, C. Wolinski, N. Fau, and F. Hamon, "A New Powerful Scalable Generic Multi-Standard LDPC Decoder Architecture," *Proc. of the 16th International Symposium on Field-Programmable Custom Computing Machines*, Palo Alto, CA, pp. 314-315, 14-15 April 2008.
- [27] R. Zarubica, R. Hinton, S.G. Wilson, and E.K. Hall, "Efficient quantization schemes for LDPC decoders," *Proc. of the IEEE Military Communications Conference*, San Diego, CA, pp. 1-5, 16-19 November 2008.
- [28] 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.
- [29] 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.

- [30] 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," *Proc. of the 40th Asilomar Conference on Signals, Systems and Computers*, Pacific Grove, CA, pp. 1192-1199, 29 October-1 November 2006.
- [31] M. Karkooti and J.R. Cavallaro, "Semi-parallel reconfigurable architectures for real-time LDPC decoding," *Proc. of the International Conference on Information Technology: Coding and Computing*, pp. 579-585, 5-7 April 2004.
- [32] H. Ding, S. Yang, W. Luo, and M. Dong, "Design and Implementation for High Speed LDPC Decoder with Layered Decoding," *Proc. of the WRI International Conference on Communications and Mobile Computing*, Yunnan, pp. 156-160, 6-8 January 2009.



**Vikram Arkalgud Chandrasetty** received Bachelor Degree in Electronics and Communication Engineering from Bangalore University (India) in 2004 and Master Degree in VLSI System Design from Coventry University (UK) in 2008. He was working with Core Networks Division at Motorola India as

Software Engineer (2005-2007), where he was part of the billing and call processing R&D team of Motorola Soft-Switch (MSS) for Mobile Switching Centres (MSC). He also worked for SoftJin Technologies as Senior Software Engineer (2007-2008) focusing on Electronic Design Automation (EDA) and FPGA applications design. He was involved in the design and development of Programmable Synthesis Engine (PSE) for custom FPGA architectures and structured ASICs. He was also working on software modelling and FPGA implementation of Motion Estimation algorithms for H.264 Advance Video Coder.

Mr Vikram is currently working towards his doctoral thesis at the School of Electrical and Information Engineering, University of South Australia. He is exploring low complexity algorithms for decoding LDPC codes and investigating efficient architectures for hardware implementation. His research is mainly focused on implementing high performance LDPC decoders on reconfigurable devices.



**Syed Mahfuzul Aziz** received Bachelor and Masters Degrees, both in electrical & electronic engineering, from Bangladesh University of Engineering & Technology (BUET) in 1984 and 1986 respectively. He received a Ph.D. degree in electronic engineering from the University of Kent (UK) in 1993 and a Graduate Certificate in higher education

from Queensland University of Technology, Australia in 2002.

He was a Professor in BUET until 1999, and led the development of the teaching and research programs in integrated circuit (IC) design in Bangladesh. He joined the University of South Australia in 1999, where he is currently an associate professor. In 1996, he was a visiting scholar at the University of Texas at Austin when he spent time at Crystal Semiconductor Corporation designing advanced CMOS integrated circuits. He has been involved in numerous industry projects in Australia and overseas, and has attracted funding from reputed research organisations such as the Australian Research Council (ARC), Australian Defence Science and Technology Organisation (DSTO), and Cooperative Research Centre, Australia. He has authored over 110 refereed research papers. His research interests include digital CMOS IC design and testability, modelling and FPGA implementation of high performance processing systems, biomedical engineering and engineering education.

Prof Aziz is a senior member of IEEE and a member of Engineers Australia. He has received numerous professional and teaching awards including the Prime Minister's Award for Australian University Teacher of the Year (2009). He has served as member of the program committees of many international conferences. Among the journals he has reviewed in the last five years are the *IEEE Transactions on Computer*, *IEEE Transactions on Image Processing*, *IEEE Transactions on Education*, *IEEE Communications Letters*, *Electronics Letters*, *Computers & Electrical Engineering – An International Journal*.