International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Index Copernicus Value (2013): 6.14 | Impact Factor (2013): 4.438

A Memory Efficient -Fully Parallel QC-LDPC Encoder

Vishnu Nampoothiri V [1], Sajith Sethu P [2]

[1]M.Tech student, SCT College of Engineering, Pappanamcode, Trivandrum, Kerala

[2]Assistant Professor, SCT College of Engineering, Pappanamcode ,Trivandrum ,Kerala

Abstract: ***Low-Density Parity Check codes are a special class of linear block codes widely used in communication and disk storage systems, due to their Shannon limit approaching performance and their favourable structure. A special class of LDPC codes, called QC-LDPC codes, allows for efficient hardware implementations of encoding and decoding algorithms by exploiting the structure of the Parity Check Matrix (PCM), which is composed of circulant permutation matrices. These codes have encoding advantage over other types of LDPC codes. In this paper an efficient QC-LDPC encoder and decoder are developed. Belief propagation algorithm is used for decoding .Overall system is developed in Matlab and performances are compared for different rates. This work also introduces a memory efficient high throughput VHDL implementation for the encoder. Due to their error correction strength, QC-LDPC codes have been recently adopted in several industrial standards such as wireless local area networks (Wi-Fi, IEEE802.11 n, ac, and ad) and Digital Video Broadcasting- Satellite- Second Generation (DVB-S2).***

Keywords: QC-LDPC codes, Encoding algorithm, Multilevel Expansion ,LU decomposition, Parallel LDPC encoder, Throughput.

1. Introduction

In 1948, Claude E. Shannon demonstrated in his paper [1] that data can be transmitted up to full capacity of the channel and free of errors by using specific coding schemes. The engineers at that time and today were surprised by Shannon's channel capacity that has become a fundamental standard for communication engineers as it gives a determination of what a system can perform and what a system cannot perform. Many of the researchers have formed different coding techniques to rise above the space between theoretical and practical channel capacities. Theses codes can be sorted by simple like repetition codes to a bit multipart codes like cyclic Hamming and array codes and more compound codes like Bose-Chaudhuri-Hocquenghem (BCH) and Reed-Solomon (RS) codes. So most of the codes were well design and most of them make distinct from Shannon's theoretical channel capacity.

A new coding technique, Low Density Parity Check (LDPC) codes are a special class of linear block codes, were introduced by Gallager in 1962 [6] .LDPC codes were carry out to achieved near the Shannon's limit but at that time almost for thirty years the work done by Gallager was ignored until D. Mackay reintroduced these codes in 1996.These codes have remarkable performance near Shannon's limit when perfectly decoded.A special class of LDPC codes, called QC-LDPC codes, allows for efficient hardware implementations of encoding and decoding algorithms by exploiting the structure of the Parity Check Matrix (PCM), which is composed of circulant permutation matrices.So QC-LDPC codes can be encoded efficiently with shift registers. Due to their excellent error correction capability and the availability of parallel decoders, LDPC codes have been lately selected by the digital video broadcasting (DVB-S2) standard and high-throughput wireless local area network (LAN), IEEE 802.16e, IEEE 802.11n, 10Gb Ethernet and magnetic storage.

In the LDPC encoder design, the direct method is to multiply the information bits with the dense generator matrix derived from the sparse PCM. The typically large code word length and density of the generator matrix make this method impractical due to its high complexity. Richardson and Urbanke [7] reduce an LDPC Parity Check Matrix in to an almost lower triangular by row and column reduction thus maintaining the sparsity of the PCM. Neal introduced an encoding method [4], where LU decomposition is used to avoid the computational complexity of multiplication by a dense inverse matrix. In our approach first we apply LU-decomposition on H_2^{-1} and then perform matrix multiplications to calculate the parity bit. In this paper we introduce serial and parallel encoder architecture for QC-LDPC codes. In this paper scheme reduces memory requirements, Here PCM is represented as compressed base shift matrix. That reduces memory requirements. LU decomposition of H_2^{-1} further reduces memory requirements.

Belief propagation (BP) algorithm is used to decode the received code word bits. Bit Flipping (BF) algorithm and the Sum Product Algorithm (SPA) are the hard decision & the soft decision decoding algorithms of BP. Bit flipping algorithm initially a hard decision is taken at the received codeword bits and the sum product algorithm is used to take information of channel proper ties and this is used to find the probabilities of the bit received at the other end, here at the end of SPA a hard decision is taken and soft information is utilized that is related to the received bits. In MATLAB a library of different functions was created to perform encoding, decoding, and performances were compared for different rates. Results give an idea that even at lower SNR these LDPC codes performed much better and have no error floor. BER curves are also with 1.5-2.5dB from Shannon's limit at BER of 10^{-6} for different rates.

2. Literature Survey

2.1 Richardson Encoding Scheme

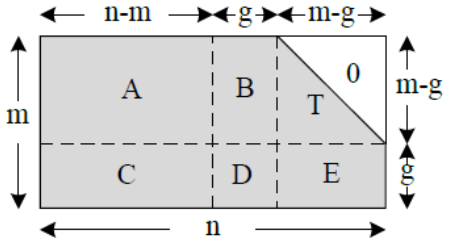

Figure 1: Richardson's proposal for encoding.

LDPC codes are linear codes. Hence, they can be expressed as the null space of a parity-check matrix H.

$$c..H^T = 0 \quad (1)$$

In the LDPC encoder, the direct method is to multiply the information bits with the dense generator matrix derived from the sparse PCM.The typically large code word length and density of the generator matrix make this method impractical due to its high complexity. The actual encoding requires $O(n^2)$operations since, in general, after the preprocessing the matrix will no longer be sparse. Richardon's algorithm makes use of the sparseness of the parity check matrices of LDPC codes and achieve much smaller complexity than conventional encoding algorithms .Richardson proposed their encoding scheme with near $O(n+g^2)$ complexity and is shown in Fig. 1. ,where g is a parameter called the gap of code.

2.2 LDPC Encoding Using Triangular Factorization

LDPC encoding using Triangular factorization[3] was developed by Yuichi Kaji.In this he also consider parity check equation to develop code word.He also divide the H matrix in to H1 and H2 .Using triangular factorization found Lower (L) and Upper(U) triangular matrices. Thus H2 can be represented as

$$H_2 = L \cdot U \quad (2)$$

The parity bits are calculated in two step
Step1: Compute
Step2: Solve
1. Compute v^T=L^{-1} U^T by aback substitution for L,and then
2. Compute p^T= U^{-1} v^T by a substitution for U.
This algorithm is known as TSTF (Two Stage encoding with the Triangular Factorization).But here the problem is that he not mentioned about the QCLDPC codes. He has shown that LU decomposition of leads to less encoding complexity than RU encoding complexity, however, his analysis does not deal with the structure of L^{-1} and U^{-1} in the case of QC-LDPC codes, where they could have less nonzero sub-matrices than H_2^{-1}.

2.3 LDPC Encoding using PCM

In general, a systematic (n, k) binary LDPC code has k information bits and n coded bits with code rate r = k/n. The parity-check matrix H is of size (n − k) × n, where n − k = m is the number of parity bits, and it defines a set of equations by equation (2) ,where c is a codeword. Let H1 and H2 be sub-matrices of H, of size m × k and m × m, respectively, such that H = [H1 H2]. Let codeword c be expressed as c = [s p], where s is a vector of k information bits and p is a vector of the m parity bits. From (1),

$$H_1 \cdot s^T + H_2 \cdot p^T = 0 \quad (3)$$

$$p^T = H_2^{-1} \cdot H_1 \cdot s^T \quad (4)$$

Here H^{-1} is not sparse .The high density of H^{-1} in (3) requires substantial hardware cost for the storage of the matrix. Furthermore, operations with dense matrices increases the computational complexity.In certain cases, the structure of the PCM allows for the reduction of encoding complexity. Encoding architecture of QC-LDPC has low complexity, they take advantage of the fact that the sub-matrices of the QC-LDPC PCM are identity matrices or identity shifted matrices.

3. Methodology

3.1 Iterative PCM Construction

$$\begin{bmatrix} -1 & 81 & -1 & 28 & -1 & -1 & 14 & 25 & 17 & -1 & -1 & 85 & 29 & 52 & 78 & 95 & 22 & 92 & 0 & 0 & -1 & -1 & -1 & -1 \\ 42 & -1 & 14 & 68 & 32 & -1 & -1 & -1 & -1 & 70 & 43 & 11 & 36 & 40 & 33 & 57 & 38 & 24 & -1 & 0 & 0 & -1 & -1 & -1 \\ -1 & -1 & 20 & -1 & -1 & 63 & 39 & -1 & 70 & 67 & -1 & 38 & 4 & 72 & 47 & 29 & 60 & 5 & 80 & -1 & 0 & 0 & -1 & -1 \\ 64 & 2 & -1 & -1 & 63 & -1 & -1 & 3 & 51 & -1 & 81 & 15 & 94 & 9 & 85 & 36 & 14 & 19 & -1 & -1 & -1 & 0 & 0 & -1 \\ -1 & 53 & 60 & 80 & -1 & 26 & 75 & -1 & -1 & -1 & -1 & 86 & 77 & 1 & 3 & 72 & 60 & 25 & -1 & -1 & -1 & -1 & 0 & 0 \\ 77 & -1 & -1 & -1 & 15 & 28 & -1 & 35 & -1 & 72 & 30 & 68 & 85 & 84 & 26 & 64 & 11 & 89 & 0 & -1 & -1 & -1 & -1 & 0 \end{bmatrix}$$

Figure 2: Base Matrix of ieee 802.16e

In QC-LDPC, PCM is represented as compressed base matrix. This matrix consist of w x w circulant permutation sub matrices and zero sub matrices. Circulant permutation sub matrices are nonzero circularly shifted identity sub-matrices. Parity check matrix of the QC-LDPC is constructed by a matrix expansion process. To illustrate multilevel expansion process[5], consider a binary base matrix H (i) in the ith expansion level. The parity check matrix of a QC-LDPC code, H(i+1), is obtained by replacing each element hij of a base binary matrix H (i) by a w(i) × w(i) permutation matrix when hij = 1, or by a w(i) × w(i) zero matrix otherwise.

$$H(i) = Ex_1^v\{Hb(i), w(i)\} \quad (5)$$

where Ex{M,w} represents the operation of the extension of a matrix M by using sub-matrices w × w , i = 1, 2, · · · ,v, v is the number of expansion steps, w(i) wi×wi is the square I_ sub-matrix and H(0) = H_b , where H_b the first binary base matrix in the sequence of the expansion steps. In each expansion level girth of the expanded matrix is calculated .Girth is the shortest cycle presented in tanner graph ,representation of H matrix. Girth four LDPC codes have poor decoding performance. Thus we have to construct girth-four free LDPC code. The fig. shows IEEE 802.16e standard base matrix .In which each element is the shifting factor and it has a dual diagonal H_{b2} sub matrix. There is only one level of expansion is required to construct the PCM. Using this H matrix we can calculate the parity bit.

3.1.1 Matrix Expansion Procedure

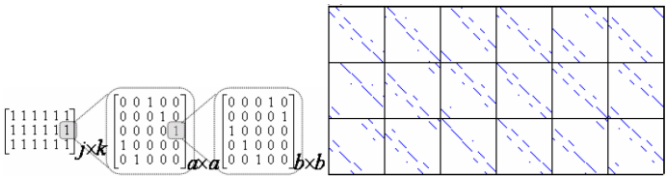

Figure 3: Matrix expansion procedure

1) Construct a base matrix $H_b = [H_{b1}\ H_{b2}]$ where H_{b2} is full-rank and girth -four free.
2) Expand H_{b2} by $w_1 \times w_1$ identity shifted matrices to produce $H_2(1)$,Expand Hb1,by $w_1 \times w_1$ permuted matrices, concatenated with $H_2(1)$ to construct girth-four free $H(1) = [H_1(1)\ H_2(1)]$
3) Expand $H_2(1)$ of step 2 by $w_2 \times w_2$ using identity shifted matrices to produce $H_2(2)$ and expand $H_1(1)$, by $w_2 \times w_2$, concatenated with $H_2(2)$ to construct girth-four free $H = [H_1(2)\ H_2(2)]$.
4) The procedure is repeated until we reach the final PCM.

3.2 Proposed Encoder Architecture

Table 1: Margin specifications

Step	*Operation*
1	$p_1^T = H_1 \cdot s^T$
2	$p_2^T = U \cdot p_1^T$
3	$p^T = L \cdot p_2^T$

Triangular factorization can reduce the complexity due to the multiplication by H_2^{-1} in (3) which can be decomposed into a lower triangular matrix and upper triangular matrix, that is:

$$H_2^{-1} = L \cdot U \quad (6)$$

$$p^T = L \cdot U[H_1 \cdot s^T] \quad (7)$$

If all the principle minors of are non-singular, the LU decomposition may not exist. In this case an extended LU decomposition is possible:[$L.U.P$]=$lu(H_2)$, where P is a permutation matrix such that

$$p^T = U^{-1} \cdot L^{-1}[(H_1 \cdot s^T)] \quad (8)$$

LU decomposition of H_2^{-1} of the proposed codes results to sparser matrices than LU decomposition of H_2 .The proposed encoding method exploits the sparsity of the three matrices. Equation (6) can be implemented by performing a three-step encoding algorithm, listed in Table I. The sub-matrix H_2 is inverted within Galois Field (2) .The obtained result is decomposed into a lower triangular matrix and an upper triangular matrix using Gaussian Elimination. Matrices L and U are precomputed, compressed and stored in appropriated memories.

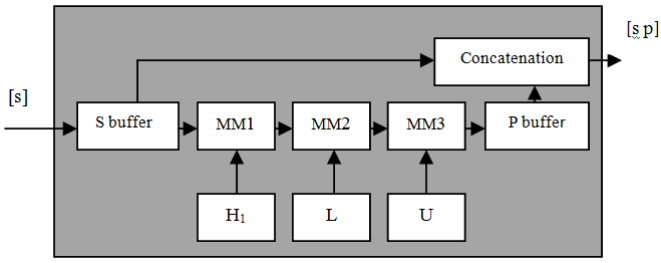

Figure 4: Overview of the proposed hardware encoder.

An overview of the proposed hardware encoder architecture is shown in Fig. 2. First we propose an architecture where the operations are grouped into three identical stages, executed serially. We first propose a single-level compression scheme, subsequently extended to a recursive multi-level scheme. The particular technique to reduce the memory required to store a sparse binary matrix, is to store the location of nonzero elements only . The basic operation performed by each of the three stages of the encoder core is row-vector by matrix multiplication (VMM). We consider that each row vector is composed of sub-row vectors each of length equal to w.

3.3 Serial Encoder Architecture

3.3.1 First Encoding Step

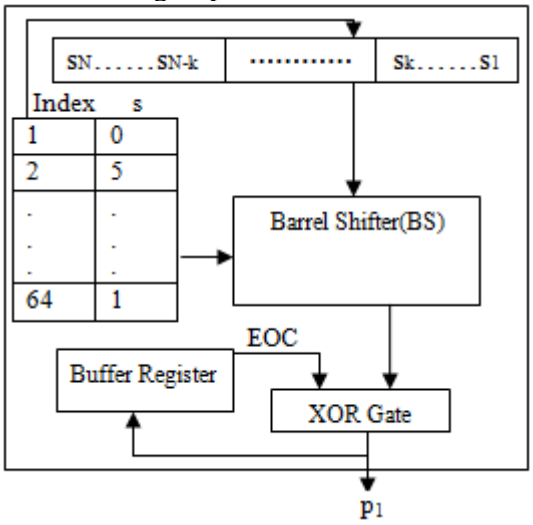

Figure 5: First-stage MM Unit.

In the first step is involved H_1. H_1 matrix is stored as compressed base matrix where its elements represent the values of the sub-matrices it can be stored as look up table. For a code constructed in steps, we use the notation to denote EOC (End Of Column) is a one-bit flag, which, when asserted, denotes that the entry to be subsequently read, refers to the next column. Multiplication of a vector by a circularly shifted matrix is equivalent to a corresponding circular shift of the vector. Therefore we use a Barrel Shifter (BS) modified to perform circular shifts to implement the multiplication of sub-row vectors with the nonzero sub-matrices. All partial products are accumulated by exclusive-or gates and a w-bit register, as shown in Fig. 5. The employed compressing method ensures that the memory required to store the matrices needed for encoding, depends on the density of each matrix rather than its size.

3.3.2 Second and Third Encoding Step

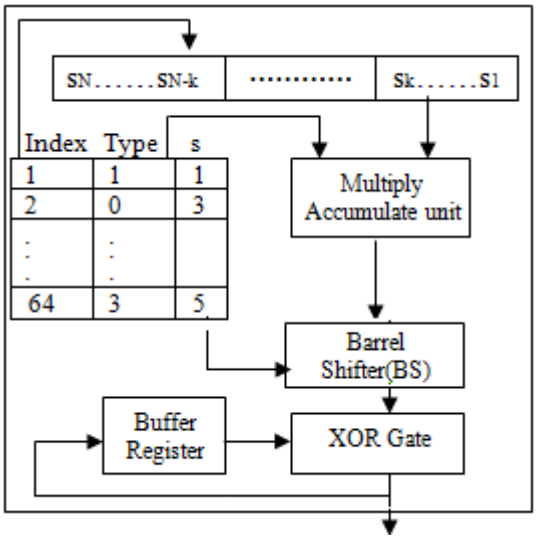

Figure 6: Second and Third stage matrix multiplication unit.

The second and third stage of matrix multiplayer unit is more complex than first stage shown in fig.6 . It consist of an additional Multiply Accumulate unit .It is used to find the multiplication of type of matrix other than identity shifted matrices with the vector. Because this type multiplication cannot find out using a simple shifting operation. Therefore we first to multiply the sub-row vectors with the corresponding sub-matrices using a Modulo-2 row vector-matrix multiplier, called MAC, and subsequently circularly shift the result by the corresponding shifting factor. Matrices L and U do not consist only of identity I ,circularly shifted identity, and zero sub-matrices thus, because of the inversion of H_2 ,they differ from the structure of H_1. The matrices L and U consist of $w_1 \times w_1$ zero sub-matrices and a limited number of types of $w_1 \times w_1$ nonzero sub-matrices. Furthermore, it is interesting to note that the nonzero sub-matrices are obtained by circularly shifting an even more limited set T of nonzero sub-matrices. The particular details of the structure of L and U are exploited to decrease the memory required for their storage. The different types of matrices other than zero matrices and identity shifted matrices present in the inverse matrix are shown in fig.7.

$$\text{Type1} = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \quad \text{Type2} = \begin{pmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{pmatrix}$$

Figure 7: Type of non zero sub matrix in Land U

3.4 Parallel Encoder Architecture

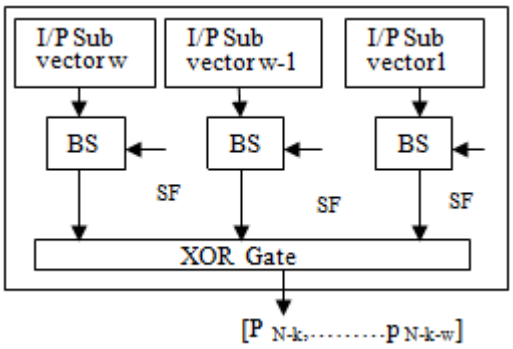

Figure 8: Parallel architecture for one column of Base matrix

The serial encoding system multiplies row vectors by the corresponding compressed matrices, one w-column at a time. To decrease the encoding latency, we can parallelize each multiplication. Parallel architecture for multiplication of input vector with one column of base matrix is shown in fig 8.In this architecture the input is divided into sub vectors of length w and applied to separate barrel shifter. Shifting factor is also applied to the BS and output is XORed simultaneously. This block is repeated for all columns of the base matrix. Parity bit is the concatenated version of all these parallel block .We get code word by concatenating the input and parity bit.The fully parallel architecture of Fig. 6 execute the three-step encoding algorithm in a single clock.So the through put is very high with the expense of hard ware complexity .The memory requirement for parallel encoder is same as that of serial architecture.

4. Result Analysis

4.1 VHDL Implementation Results

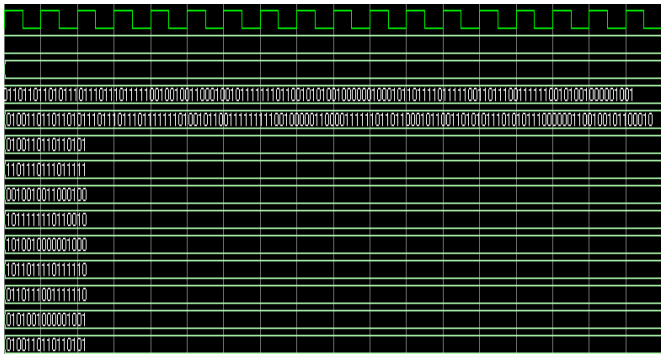

Figure 9: ModelSim output waveform

The serial and parallel encoders are developed in Model-Sim. ModelSim output waveform is shown in fig.9. Xilinx ISE 8.1i software is used to find the slice utilization of serial and parallel encoders. The proposed encoder architectures have been implemented onto hardware for various codes.Li [3] exploit the circulants in the structure of the matrices involved in encoding using the Shift-Register-Add-Accumulator (SRAA) circuit as a building block for a variety of architectures. The SRAA circuits require w clocks to multiply a w-bit vector by wxw circulant matrix. Our architecture utilizes BS to perform the same basic operation in a single clock, at the expense of a longer maximum delay path.

Table 2: Comparison of Serial and Parallel architecture

Architecture	Slices	Throughput
Serial	424	52Mb/s
2 parallel	517	102Mb/s

The occupied slices refer to the proposed serial architecture using single-level compression. The architecture is found to be efficient for several QC-LDPC codes .The proposed parallel architecture increases the encoding throughput with a moderate increase of area complexity. Table 2 reports the encoding complexity of code for two cases, a serial one and a parallel one. The particular parallel architecture uses distinct MM units and not shared MMs.The proposed encoder is memory efficient and low complex in comparison with ordinary LDPC in which large number of components and memory are required . Degree-2 parallel architecture increases throughput by a factor of two, while area complexity is only slightly increased in comparison with serial architecture. In N parallel architecture throughput is very high in the range of GB (Giga Byte) with moderate hardware complexity.

5. Conclusion

We have introduced a memory efficient Serial encoder and Parallel encoder for QC-LDPC codes. The recursive code construction allows flexibility of creation of codes with good performance and low error floors comparable to codes of IEEE802.16e standards. The architectures are flexible in the sense of supporting wide range of code lengths and rates. The proposed Parallel architecture has high throughput compared to serial architecture with the expense of moderate hardware complexity. Proposed encoder is found to reduce complexity compared with existing methods.

References

[1] C.E. Shannon, "A Mathematical Theory of Communication," Bell System Technical journal, July 1948.
[2] Ahmed Mahdi and Vassilis Paliouras, "A Low Complexity-High Throughput QC-LDPC Encoder" IEEE Trans on Signal Processing, Vol. 62, no. 10, May 2014
[3] Z. Li, L. Chen, L. Zeng, S. Lin, andW. Fong, "Efficient encoding of quasi-cyclic low-density parity-check codes," *IEEE Trans. Commun.*,vol. 54, no. 1, pp. 71–81, Jan. 2006.
[4] Y. Kaji, "Encoding LDPC codes using the triangular factorization,"*IEICE Trans. Fundamentals Electron., Commun. Comput.Sci., vol.* E89-A, pp. 2510–2518, Oct. 2006
[5] Ahmed Mahdi and Vassilis Paliouras, "Simplified Multi-level Quasi-Cyclic LDPC codes for Low-Complexity Encoder" IEEE Workshop on Signal Processing Systems Signal Processing, 2012
[6] R. Neal, IMA Programs on Codes, Syst., Graphical Models, "Sparse matrix methods and probabilistic inference algorithms—Part I: Faster encoding for low density parity check codes using sparse matrix methods," Aug. 1999.
[7] R. Gallager, "Low-density parity-check codes," *IRE Trans. Inf. Theory*,vol. 8, no. 1, pp. 21–28, Jan. 1962.
[8] T. Richardson and R. Urbanke, "Efficient encoding of low-density parity-check codes," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp.638–656, Feb. 2001.

Author Profile

Vishnu Nampoothiri V is currently doing M.Tech. Degree in Signal Processing with the Department of Electronics and Communication Engineering, SCT College of Engineering, Pappanamcode, Trivandrum, Kerala. He received the B.Tech degree from the University of Kerala, Thiruvananthapuram, in 2013 in Electronics and Communication Engineering.His research interests include areas in communication and coding theory.

Sajith Sethu P has been working as an Assistant Professor under Electronics and Communication Department in SCT College of Engineering, Pappanamcode since 2000. He received the B.Tech and M.tech degree from the University of Kerala, Thiruvananthapuram. He is currently pursuing his Ph.D. Degree from the University of Kerala.His research interests include areas in Information theory and coding.