New Code Construction Method and High-Speed VLSI Codec Architecture for Repeat-Accumulate Codes

Kaibin Zhang*, Liuguo Yin**, Jianhua Lu*
*Department of Electronic Engineering, Tsinghua University, Beijing, 100084, China
**School of Aerospace, Tsinghua University, Beijing, 100084, China
yinlg@tsinghua.edu.cn

Abstract: Repeat-Accumulate code has been widely used in various practical communication systems due to its special structure that can be encoded with a low complexity turbo-like encoder and decoded with a high-speed parallel algorithm. However, the construction of RA codes with large girth is still a challenging problem. In this paper, the girth upper bound of RA codes is analyzed and a novel RA code construction scheme is proposed. This scheme generates the parity-check matrix with superimposed structured interleavers, guaranteeing the resulting matrix to be globally optimized. Compared with the RA codes constructed with traditional structured interleavers such as π-rotation interleavers, pseudo-random interleavers, combinatorial interleavers, and modified L-type interleavers, the codes constructed with the proposed method exhibit better error-correcting performance. Moreover, a parallel VLSI codec architecture is proposed for the RA codes generated with the proposed method in this paper. Modifying the conventional RA encoder architecture, the codec for RA codes from our proposed method may have the same hardware implementation complexity as that for traditional RA codes, while the data throughput of our codec may achieve up to 450Mbps, which may be of great value for practical applications.

Key–Words: Repeat-Accumulate (RA) codes, low-density parity-check (LDPC) codes, superimposed structured interleaver (SSI), hardware implementation

1 Introduction

Repeat-Accumulate (RA) code is a class of low-density parity-check (LDPC) codes with outstanding error-correcting capability and inherently parallelizable decoding scheme [1][2][3]. Moreover, the RA code has a "turbo-like" structure with two constitute codes being a repetition code and a convolutional code, the encoding of which may thereby be implemented with linear-complexity. As a result, RA code has been accepted as one of the most successful LDPC codes and are widely used in various communication systems such as DVB-S2 [4][5] and WiMax [6][7].

However, the construction of an RA code still has some challenging problems. A parity-check matrix with fully random interleavers may obtain good error-correcting performance, while the hardware implementation complexity may be very high. For practical applications, the parity-check matrix should be constructed with semi-random interleavers that could obtain near Shannon limit error-correcting performance and could be implemented with complexity close to that of structured matrix. In recent years, many efficient semi-random interleavers are proposed for RA codes, such as π-rotation interleavers [8], pseudo-random interleavers [9], and combinatorial interleavers [10]. The parity-check matrixes with these interleavers are generally constructed with a concatenated scheme, which generates the parity-check matrix with two steps: construct a based matrix and then each zero element of the based matrix is expanded to be a zero submatrix and each nonzero element is expanded to be an interleaved identity matrix. This scheme may obtain an LDPC code with good trade-off between coding performance and hardware complexity, while the parity-check matrix is locally optimized result rather than a globally optimized one. In recent years, the construction of RA codes with globally optimized scheme is a hot topic.

In this paper, a novel scheme is proposed for constructing good RA codes. This scheme generates the parity-check matrix with superimposed structured interleavers (SSI), and thereby guaranteeing the resulting matrix globally optimized. Compared with the RA codes constructed with traditional structured interleavers such as π-rotation interleavers, pseudo-random interleavers, combinatorial interleavers and modified L-type interleavers, the codes constructed with the proposed method exhibit better error-correcting performance with the same hardware implementation complexity.

The rest of this paper is organized as follows. In Section 2, the girth upper bound of RA codes is an-

alyzed. In Section 3, the repeat-accumulate codes and its parity-check matrix construction schemes with traditional structure interleavers are described. Then, the proposed code construction method with superimposed structured interleavers is proposed in Section 4. After that, simulation results for codes generated with the proposed algorithm are presented in Section 5. Moreover, the hardware implementation issue of the RA codes with superimposed structure interleavers is discussed in Section 6. Finally, conclusions are drawn in Section 7.

2 The Girth Upper Bound of Repeat Accumulate Codes

Repeat-Accumulate Code is a special class of LDPC code, which is also represented by a Tanner graph. The path $\{v_0, c_0, v_1, c_1, \ldots v_n, c_n, v_0\}$ in the Tanner graph is called a cycle, where v_i and s_i represent the variable node and the parity check node respectively. Cycles in the Tanner graphs of LDPC codes prevent the sum-product algorithm from converging [11][12]. Furthermore, cycles, especially short cycles, degrade the performance of LDPC decoders, because they may affect the independence of the extrinsic information exchanged in the iterative decoding. Hence, LDPC codes with large girth are desired. The length of the shortest cycle in the graph is called girth. The parity matrix of RA codes contains a sub-diagonal matrix H_m, thereby only design the H_c to increase the girth.

For (ρ, λ)-regular LDPC codes, length N, the girth upper bound[13] is determined by :

$$m = \left\lfloor \frac{\log N + \log \frac{\rho\lambda - \rho - \lambda}{2\lambda}}{2\log(\rho - 1)(\lambda - 1)} \right\rfloor \tag{1}$$

$$Girth_{\max} = 4m + 4 \tag{2}$$

where ρ and λ are column weight and row weight.

By taking into account the same density regular (q, a)RA codes, we get the following equations:

$$\begin{cases} a = \lambda - 2 \\ 2(1 - R) + qR = \rho \\ \rho = \lambda(1 - R) \end{cases} \tag{3}$$

where R is the code rate and here the degree-1 node is considered a degree-2 node. H_m is a sub-diagonal matrix and it can't produce a cycle, thereby the girth upper bound evaluation of RA codes approximately equals the girth upper bound of the code of parity matrix H_c.

Using the above equations, the girth upper bound evaluations of (3,6) regular LDPC code and (4,4) regular RA code are demonstrated in Fig.1. The girth upper bounds of RA codes of length several hundreds or hundred thousands are less than that of the regular LDPC codes, while the RA codes of length one thousand to one hundred thousand retain the same bounds. It sideways shows that the RA codes have the good error correcting performance, which are widely applied in practical systems.

The girth upper bound of RA code may decrease when considering H_m. Thus, when codes with larger girth are considered the weight two columns of the accumulator have the effect of reducing the minimum distance bound for RA codes, when compared to LDPC codes with the same girth. Hence it is of great importance to design good interleavers, that equals to optimize H_c.

3 Traditional Construction Schemes for Repeat Accumulate Codes

RA codes can be considered as a special class of Irregular Low Density Parity Check (LDPC) codes [13].

The RA code parity matrix is composed of two sub-matrices, $H = [H_m, H_c]$, where $H_m$is an M by M dual-diagonal square matrix and H_c is an M by $N - M$ matrix, where N is the code length,$M = (1 - R)N$, R is the code rate. An example of H_m is shown as follows.

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

It is clear that H_m is constant for given M and thereby the crucial issue is to optimize the interleavers to improve the coding performance. It equals to optimize H_c.

It has been proved that RA codes with randomly chosen interleavers may obtain good performance. Thereby most RA research has only considered the error rate result of RA codes pseudo-randomly interleavers [1], [14], [15]. However, randomly constructed interleavers may have implementation challenges. Many researchers focus on the trade-off between performance and implementation complexity, and many such interleavers have been proposed.

3.1 RA Code Construction Scheme with π-rotation Interleaver

The algorithm constructs H_c as a composition of $m \times m$ circularly shifted identity matrices, which can

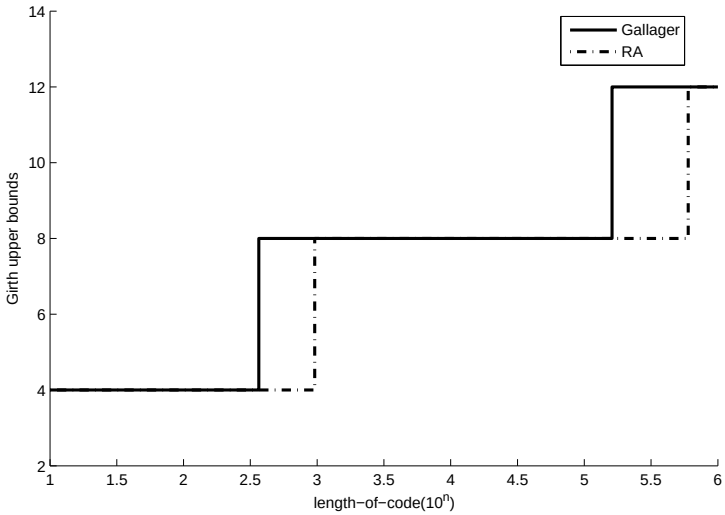

Figure 1: The girth upper bound evaluations of the same density LDPC codes and RA codes

be described by a $1 \times m$ permutation vector. We label the single sub-matrices π_A,π_B,π_C, and π_D, which rotates $\pi/2$ counterclockwise each other. For example, using the $(m = 3)$ permutation vector of [2 3 1], indicating the position of the nonzero element in each row-counting from the left. Thereby we obtain the following four π-rotations.

$$\pi_A = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix} \qquad \pi_D = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

$$\pi_B = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad \pi_C = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$

Then we can create a rate 1/2 H_c from π_A, π_B, π_C and π_D by arranging them as follows, obviously symmetrically.

$$H_c = \begin{bmatrix} \pi_A & \pi_B & \pi_C & \pi_D \\ \pi_B & \pi_C & \pi_D & \pi_A \\ \pi_C & \pi_D & \pi_A & \pi_B \\ \pi_D & \pi_A & \pi_B & \pi_C \end{bmatrix}$$

3.2 RA Code Construction Scheme with Pseudo-Random Interleaver

The construction contains two steps: First, construct a base parity-check matrix with a random generation or PEG algorithm. Second, each nonzero element of the base matrix is expanded to be an L by L sub-matrix, denotes $\pi(i, j)$, in whose k-th row, the element of the $f(\alpha^i \cdot (\alpha^j)^k)$ column is 1, others 0, for $0 \leq k \leq L-1$, $L = 2^p - 1$. α is the primitive element in Galois Field $GF(2^p)$ and its corresponding value denotes $f(\alpha)$, where p is a prime integer. For example, we set the parameter $p = 3, i = 1, j = 2$, then we get,

$$\pi(1,2) = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix}$$

The constructed interleaver is called pseudo-random interleaver and the sub-matrix is structured and determined by i and j. The pseudo-random interleaver can be as follows:

$$H_c = \begin{bmatrix} \pi(i_0, j_0) & 0 & \cdots & \pi(i_4, j_4) \\ \pi(i_1, j_1) & \cdots & \pi(i_3, j_3) & 0 \\ \cdots & \pi(i_2, j_2) & \cdots & \pi(i_5, j_5) \\ \pi(i_7, j_7) & 0 & \pi(i_6, j_6) & \cdots \end{bmatrix}$$

3.3 RA Code Construction Scheme with Combinatorial Interleaver

In [8], it has been proved that for every possible Steiner 2-design there exists a row and column permutation that maps the incidence matrix of the design into an RA code interleaver and accumulator, thus producing a high rate RA code which is 4-cycle free and the combinatorial interleaver can be implemented

easily. But construction 1 and 3 introduced in [8] only construct the RA codes with $q = 3$. Although construction 2 can construct regular RA codes with any a and q, it can not make the irregular RA codes. These three construction methods are not flexible to design a variety of RA codes. Readers can refer to it in details.

3.4 RA Code Construction Scheme with L-type and modified L-type Interleaver

Given K information bits, the number of repetition q, the base interleaver $\prod$ and the interleaver parameter l,the L-type interleaver first selects the K message bits in order and then selects from the message bits again, skipping l bits ahead after each selection. This process is repeated q times, as given as follows.

$$\begin{aligned}\prod = &[\prod_1(1), \ldots, \prod_1(K)] \ldots \\ &[\prod_{q-1}(1) + q - 2, \ldots, \prod_{q-1}(K) + q - 2], \\ &[\prod_q(1) + q - 1, \ldots, \prod_q(K) + q - 1]\end{aligned}$$

where, for $1 \leq i \leq q$

$$\begin{aligned}\prod_1 &= [1, 1 + q, \ldots, 1 + (K-1)q] \\ \prod_i &= [\prod_{i-1}(1), \prod_{i-1}(1 + l), \ldots \prod_{i-1}(1 + K - l)], \\ &\quad [\prod_{i-1}(2), \prod_{i-1}(2 + l), \ldots \prod_{i-1}(2 + K - l)], \\ &\quad \ldots \\ &\quad [\prod_{i-1}(l), \prod_{i-1}(l + l), \ldots \prod_{i-1}(l + K - l)].\end{aligned}$$

Thus the L-type interleaver is constructed. Setting $l = 2a$, a $(3, a)$-regular RA code can be constructed without 6-cycles whenever $K \geq 8a^3$ by using L-type interleaver [16].

In order to eliminate these length-8 cycles, a modified L-type interleaver is proposed as follow. In practice the bits $iK+1$ to $(i+1)K$ of $\prod$, which is denoted by $\prod_{i+1}$, are a type of row-column permutation of bits $(i-1)K+1$ to iK of $\prod$, denoted by $\prod_i$. First the bits are written row-wise into a matrix , P_i with l columns and read out column-wise. Next, the bits from each column are written row-wise into another matrix, Q_j, and read out column-wise. The $j - th$ column of P_i is written into a matrix Q_j with j columns.

In summary, the above RA code construction scheme always contains the following steps: First, construct an M_b by $N_b - M_b$ base matrix, where $M = M_bL$, $N = N_bL$ and L is an integer; Then each non-zero element of the basic matrix is expanded to be an L by L sub-matrix, which is generally a circulant permutation matrix distinct from identity matrix or a random matrix and each zero element is expanded to an L by L zero matrix. It is noted that with the above steps the basic matrix and the expanded matrix are constructed independently, and thereby the resulted parity-check matrix is locally optimized rather than globally optimized.

4 Proposed RA Code Construction Scheme

In [17], parallel edges are permitted in a protograph, whose Tanner graph $G = (V, C, E)$ consists of a set of variable nodes V, a set of check nodes C, and a set of edges E, the mapping $e \rightarrow (v_e, c_e)$ is not necessarily 1:1. The statement shows the sub-matrices can be superimposed in the same location. The above ideas motivate us to come up with designing better interleavers for RA codes from the global view, which we call superimposed structured interleavers(SSI).

Before starting designing SSI, some signs are introduced for convenient description. For simplicity, (i, j, l) means the non-zero element in row l in $\pi(i, j)$. Clearly, the sub-matrix $\pi(i, j)$ is structured and determined by i and j. After introducing the signs, the construction of SSI is proposed as follows:

Step1: According to the code length, and rate, choose the suitable expansion factor $L = 2^p - 1$, where p is a prime integer.

Step2: If all column-blocks are filled, algorithm exits. Otherwise, if weights of all rows are great than the maximum, go to step7; else, randomly select one row under the upper bound.

Step3: If all $\pi(i, j)$ do not meet the requirements, go to step2; Otherwise select one.

Step4: If no sub-matrix has been inserted, go to Step5; Otherwise compare with the inserted sub-matrix, if nonzero elements overlap, go to Step3.

Step5: Given G, If all the nonzero elements satisfy: for any k non-zero elements in H,$1 \leq k \leq G/2$, denoting $(i_0, j_0, l_0), (i_1, j_1, l_1) \ldots (i_{2k-1}, j_{2k-1}, l_{2k-1})$,

$$\begin{cases} l_{2k-2} = l_{2k-1} \\ f(\alpha^{i_{2k-2}} \cdot (\alpha^{j_{2k-2}})^{l_{2k-2}}) = f(\alpha^{i_{2k-1}} \cdot (\alpha^{j_{2k-1}})^{l_{2k-1}}) \end{cases}$$

insert the sub-matrix; otherwise go to Step3.

Step6: If all the sub-matrices have been inserted completely in this column-block, then insert the next, go to Step2; otherwise directly go to Step2.

Step7: delete the formerly inserted sub-matrix, go to Step2.

After that, slightly coordinate the sub-matrices to meet the row-block requirements if necessary, then the formula on behalf of superimposed structured interleaver could be shown in equation (4).

5 Simulations and Numeric Results

With the SSI, a rate 0.5, code length 2032 RA code is constructed with L=127, and the column-weight den-

$$H_c = \begin{bmatrix} \pi(i_0, j_0) \oplus \pi(i_1, j_1) & 0 & \cdots & \pi(i_8, j_8) \\ 0 & \cdots & \pi(i_3, j_3) & 0 \\ \cdots & \pi(i_2, j_2) & \cdots & \pi(i_5, j_5) \oplus \pi(i_4, j_4) \oplus \pi(i_9, j_9) \\ \pi(i_7, j_7) & 0 & \pi(i_6, j_6) & \cdots \end{bmatrix} \tag{4}$$

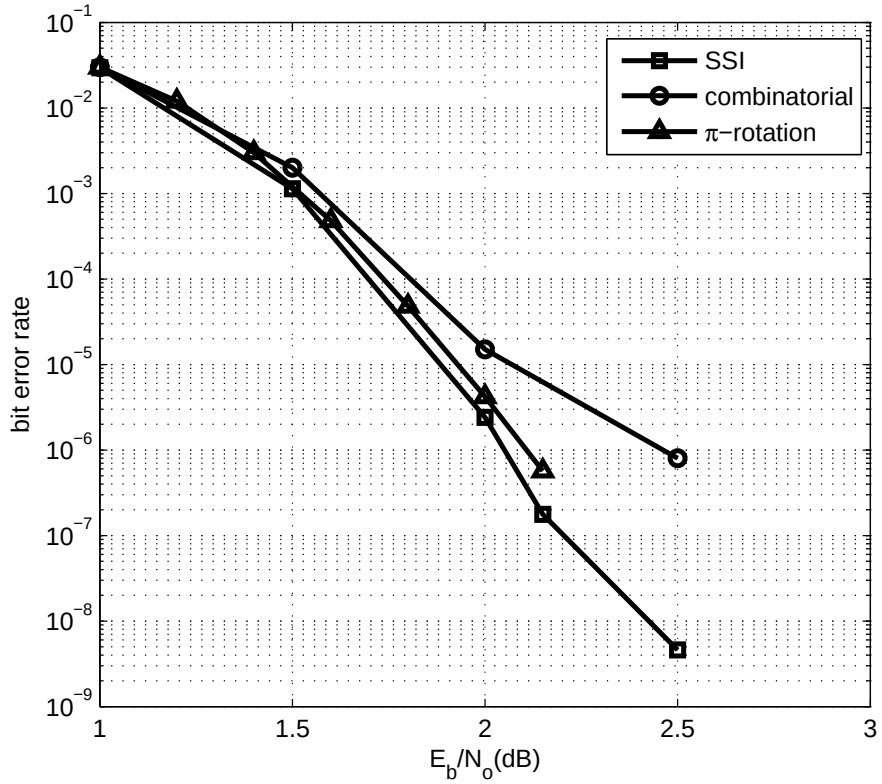

Figure 2: The performance of RA codes with superimposed structured, combinatorial and π-rotation interleavers

sity function and row-weight density function are

$$\lambda(x) = 0.5x + 0.5x^3 \tag{5}$$

and

$$\rho(x) = 0.125x^4 + 0.875x^5 \tag{6}$$

The error correction performance of the code is presented in Figure 2. We are aware of one combinatorial interleaver for RA codes from [8], named modified construction 2. The performance curve of the code with the same degree distribution and rate constructed is plotted, length 2022. The performance of the RA code with π-rotation interleaver and the same parameters are presented, length 2000. In Figure 3, we demonstrate the performance of the RA code with SSI compared to that with pseudo-random interleaver, plotted by round and square lines respectively, length 16352, rate 0.5, weight density function same as shown in (5)(6). In Figure 4, the performance comparison of the RA code of length 9690 with SSI and that of length 10000 with modified L-type interleaver is proposed, plotted by round and square lines respectively.

It is shown in Figure 2 that the performance of the code generated with the proposed method is much better than that generated with the combinatorial algorithm. The performance improvement is about 0.78 dB for output bit error rate of 10^{-7}. It is observed that the code generated with the proposed method outperforms the π-rotation RA code about 0.1 dB and has a lower error floor. It is also seen from Figure 3 that the RA code with SSI significantly outperforms the code with pseudo-random interleaver and exhibit improved error-floor performance. In Figure 4,the performance improvement gets 0.05dB, compared the

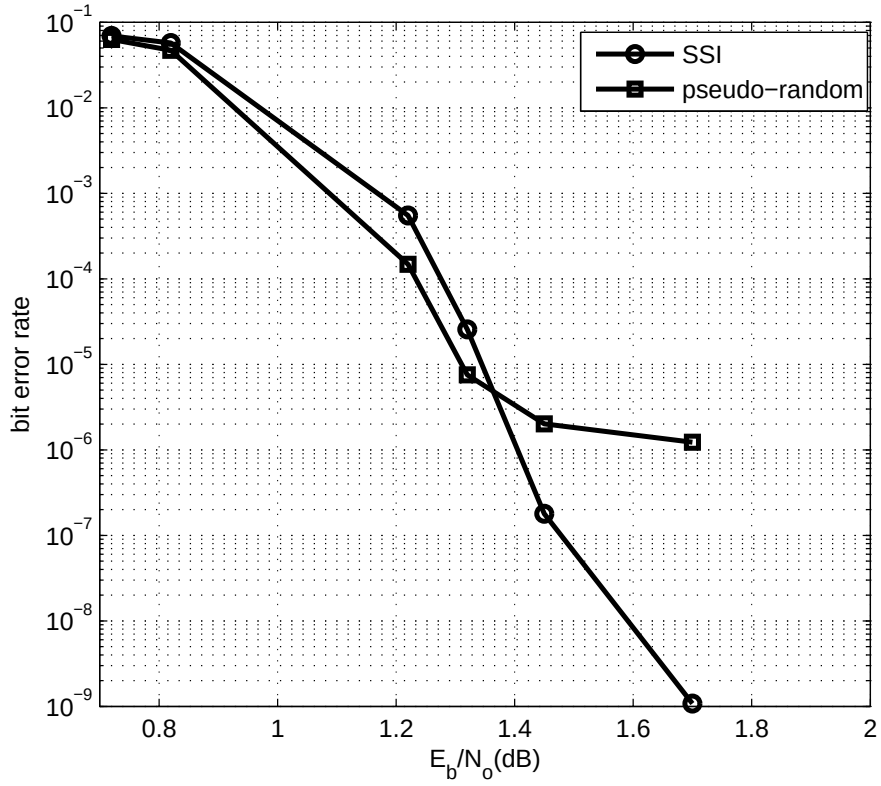

Figure 3: The performance of RA codes with superimposed structured and pseudo-random interleavers

RA code with SSI with that with modified L-type interleaver.

Compared with them, the RA codes with SSI are capable of outperforming the RA codes with three above interleavers, while the code length is almost the same.

6 Implementation Architecture

6.1 Encoder Architecture

RA codes can also be seen as a class of "Turbo-like" codes [18]. The encoding schedule is shown in Figure 4. The K information bits $m = [m_1 \dots m_K]$ are repeated l times, showed as follows:

$$\overrightarrow{b} = [m_1^{(1)}, m_1^{(2)} \dots m_1^{(l)}, \dots, m_K^{(1)}, m_K^{(2)} \dots m_K^{(l)}].$$

Then, the output sequence $\overrightarrow{b}$ is permuted with interleaver $\prod$, and the result of which is $\overrightarrow{d}$, where

$$\prod = [\pi_1, \pi_2, \dots \pi_n], \tag{7}$$

and

$$\overrightarrow{d} = [b_{\pi_1}, b_{\pi_2}, \dots, b_{\pi_n}]. \tag{8}$$

After that the parity bits $\overrightarrow{p}$ may be calculated by equation (6), for $1 \leq i \leq Kl/a$,

$$\begin{cases} g_i = d_{(i-1)a+1} \oplus d_{(i-1)a+2} \oplus \cdots \oplus d_{ia} \\ p_i = p_{i-1} \oplus g_i \end{cases} \tag{9}$$

where a is the parameter of the combiner, which deals with the message before entering into the accumulator. Then the final systematic codeword is

$$\overrightarrow{c} = [\overrightarrow{m}, \overrightarrow{p}] \tag{10}$$

Thus an RA code with length $N = K(1 + l/a)$ and rate $R = a/(a + l)$ is obtained.

Thereby we propose an RA code encoder implement architecture as shown in Figure 5. It consists of an array of memory blocks corresponding to the RA interleaver, denoted as MEMI, and a parity-check-bit-generating unit, denoted as PCBGU[10]. As shown in section 4, the SSI is structured and MEMI is associated with nonzero elements of the interleaver. The output ports of the MEMI are connected to input ports of PCBGU, which generate one parity-check bit each clock cycle. With this encoder, the whole encoding procedure could cost $N_b \cdot L$ clock cycles. Thereby in

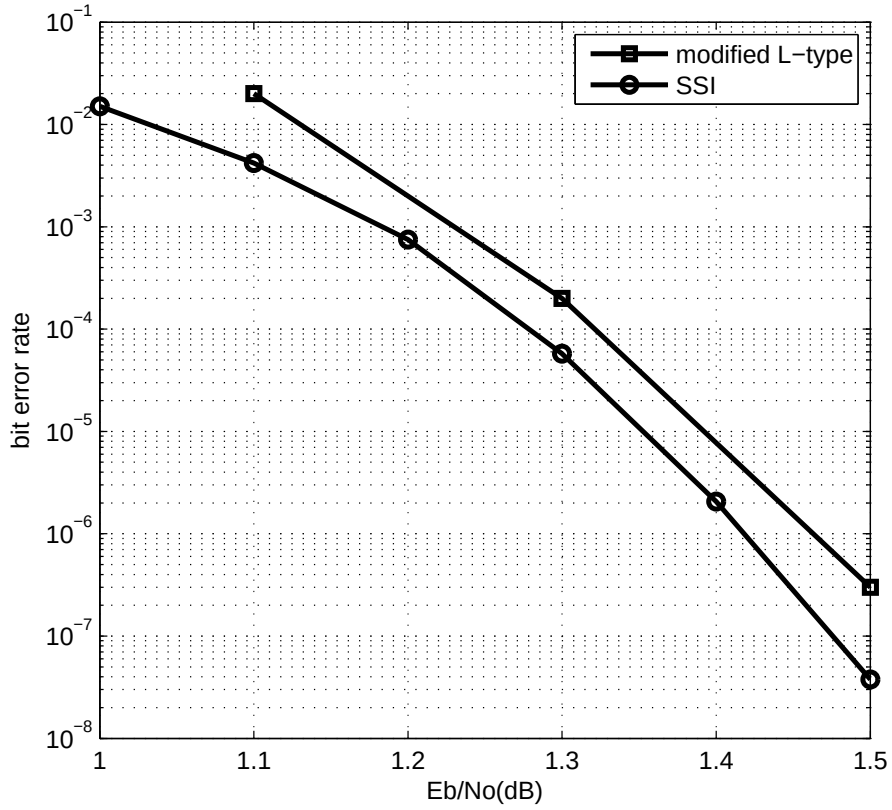

Figure 4: The performance of RA codes with superimposed structured and modified L-type interleavers

the first $(N_b - M_b) \cdot L$ clock cycle, source bits are output, and simultaneously operated in the MEMI. In the $M_b \cdot L$ clock cycles, the parity-check bits are generated by PCBGU. It is clearly shown that the implementation complexity of the RA code with SSI is probably the same as that of the RA code with the pseudo-random interleaver.

6.2 Decoder Architecture

RA codes can be described by a Tanner graph[19]. as shown in Figure 6. The regular RA code described by a Tanner graph in Figure 6 has the same parameters in Figure 4, where the message bit nodes are presented at the top of the figure and the parity check bit nodes at the bottom.

As described in section 3, the RA codeword vector $\overrightarrow{c}$is divided into $\overrightarrow{m}$ and $\overrightarrow{p}$, where $\overrightarrow{m}$ are the information bits and $\overrightarrow{p}$ the parity check bits. Corresponding to $Hc^T = \mathbf{0}$, then

$$H_m \cdot \overrightarrow{p} = H_c \cdot \overrightarrow{m} \tag{11}$$

Then we have,

$$\overrightarrow{p} = H_m^{-1} \cdot H_c \cdot \overrightarrow{m} \tag{12}$$

where H_m^{-1} is the lower triangular matrix. We can calculate $\overrightarrow{p}$ with equation (12) in any arbitrary information vector $\overrightarrow{m}$.

Thereby the RA code can be seen as an LDPC code[20] and the LDPC high-speed parallel decoder architecture could be adopted[21][22][23].

Due to its "turbo-like" structure that can be encoded with a low complexity turbo-like encoder and decoded with a high-speed parallel algorithm, these advantages make the RA codes with SSI valuable for practical communication systems.

6.3 FPGA Implementation

Based on the serial architecture shown in Figure 6, the RA encoder can be implemented easily in FPGA. However, the speed may not be high enough to be applied in many special practical systems, which promotes us to take a novel parallel encoder architecture. It is noted that eliminating some nonzero elements of

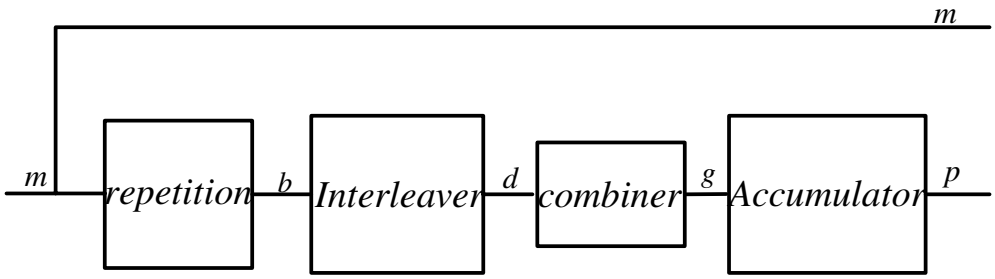

Figure 5: The encoding architecture of a length-N RA code as viewed a turbo code

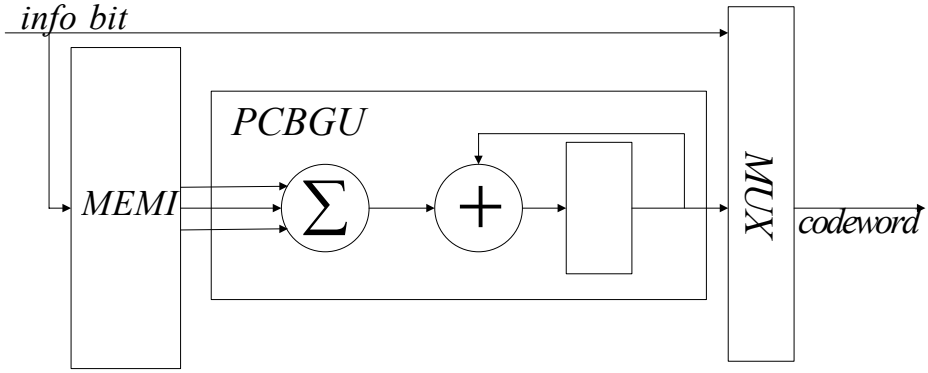

Figure 6: Encoder architecture for RA codes with SSI

H_m splits H into an assemble of parity check matrices for sub-RA codes.

Based on the above idea, $(L, L-1), (2L, 2L-1), ...((N_b-1)L, (N_b-1)L-1)$ are eliminated, where (i, j) means the element of a matrix in $i-th$ row, $j-th$ column, $N_bL = N$. Then the parallel encoder architecture is obtained and N_b information bits can be encoded synchronously.

Based on the parallel architecture, an encoder and decoder implementation of an RA code with SSI with the Xilinx Vertex-IV xc4vlx100 is achieved, where the decoder is taken by 6bit quantization. The code length is 15330 and the rate is 5/6. The variable node degree distribution and parity check node degree distribution are

$$\lambda(x) = 0.0003 + 0.1663x + 0.3334x^2 + 0.5x^3 \quad (13)$$

and

$$\rho(x) = 0.002x^{18} + 0.998x^{19} \quad (14)$$

respectively.

The codec is described in VHDL, and SYNOPSYS FPGA Complier II is used to synthesize the VHDL implementation. The encoder and decoder resource utilization statistics are listed in Table 1 and Table 2, respectively.

Table 1: Encoder FPGA resource utilization statistics

Resource Name	Used	Available	Utilization
Slices	1371	49152	2%
Slice Flip Flops	746	98304	0.75%
Input LUTs	2171	98304	2%
Bonded IOBs	17	772	2%
BRAMs	40	240	16%
GCLKs	3	32	9%

Moreover, the codec can work at the clock frequency of 240MHz in Vertex-IV xc4vlx100 and the data throughput of 450Mbps is achieved, which is very valuable for real high-speed applications.

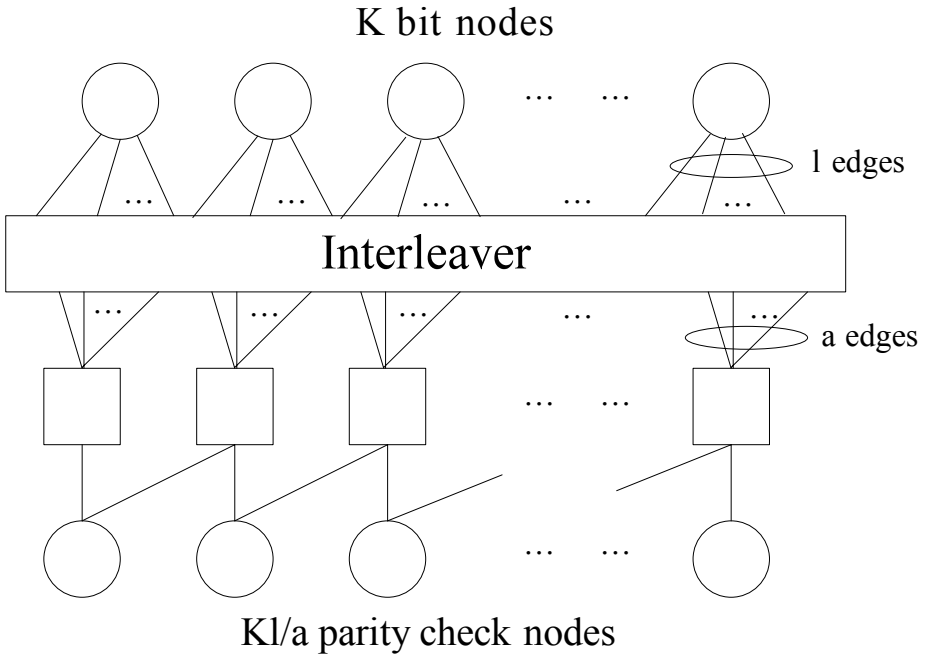

Figure 7: Decoder architecture of a length-N RA code generated with the proposed method

Table 2: Decoder FPGA resource utilization statistics

Resource Name	Used	Available	Utilization
Slices	42273	49152	86%
Slice Flip Flops	48522	98304	49%
Input LUTs	53679	98304	54%
Bonded IOBs	37	772	3%
BRAMs	164	240	68%
BUFGCTRLs	3	32	9%

7 Conclusions

In this paper, a novel scheme of constructing RA codes is proposed. This scheme generates the parity-check matrix with superimposed structured interleavers, guaranteeing it close to global optimum as possible. Compared with the RA codes constructed with traditional structured interleavers such as π-rotation interleavers, pseudo-random interleavers and combinatorial interleavers, the codes constructed with SSI exhibit better error-correcting performance with the same hardware implementation complexity and the data throughput of our codec can achieve up to 450Mbps.

Acknowledgements: This work was supported in part by the National Science Foundation of China (60525107, 60532070) and the National Fundamental Research Program of China (2007CB310600).

References:

[1] D. Divsalar, H. Jin, and R. J. McEliece, "Coding theorems for "turbo-like" codes", in *Proc. 36th Allerton Conf on Communications, Control, and Computing, Allerton, Illinois*, September 1998, pp. 201-210.

[2] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon limit error-correcting coding and decoding: turbo-codes", in *Proc. IEEE Int. Conf. on Communications (ICC93)*, Geneva, Switzerland, May 1993, pp. 1064-1070.

[3] Chung S Y,Forney J G. D,Richardson T, and Urbanke R."On the design of low-density parity-check codes within 0.0045 dB of the Shannon limit", *IEEE Trans.Lett.* 2001, pp. 58-60.

[4] Digital video broadcasting (DVB): Second generation framing structure, channel coding and

modulation systems for broadcasting, interactive services, news gathering and other broad-band satellite applications, EN 302 307, European Telecommunications Standards Institute (ETSI).

[5] Morello A. and Mignone V."DVB-S2: The Second Generation Standard for Satellite Broad-Band Services", Proceedings of the IEEE,Volume 94, Issue 1, Jan. 2006, pp.210-227

[6] ElMahgoub K. and Nafie M. "On the Enhancement of LDPC Codes in the IEEE 802.16 Physical Layer", ICC '07. IEEE International Conference on Communications, 24-28 June 2007, pp. 683-688

[7] Xin-Yu Shih, Cheng-Zhou Zhan, Cheng-Hung Lin and An-Yeu Wu,"A 19-mode 8.29mm2 52-mW LDPC Decoder Chipp for IEEE 802.16e System" *IEEE Symposium on VLSI Circuits*, 14-16, June, 2007, pp. 16-17

[8] Sarah J. Johnson and Steven R. Weller, "Combinatorial Interleavers for Systematic Regular Repeat Accumulate Codes", *IEEE Trans. on Communications*, vol.56, no. 8, August 2008, pp. 1201-1206.

[9] R. Echard and S . C. Chang, "The π-rotation low-density parity-check codes," *in Proc. Global Telecommunications Conference*, San Antonio, TX, Nov.2001, pp. 980-984.

[10] Yukui Pei, Liuguo Yin and Jianhua Lu, "Design of irregular LDPC codec on a single chip FPGA", *Frontiers of Mobile and Wireless Communication*, 31 May-2 June 2004, vol. 1, pp. 221-224.

[11] R. J. McEliece, D. J. C. Mackay, and J. F. Cheng, "Turbo decoding as an instance of Pearl's belief propagation algorithm", *IEEE J.Secleted Areas Commonication*, vol.16,no.2, pp.140-152, Feb 1998.

[12] T. Etzion, A. Trachtenberg, and A. Vardy, "Which codes have cycle-free for magnetic recording", *IEEE Tans.Inform.Theory*, vol.45, no.6, pp.2173-2181, Sep 1999.

[13] R. G. Gallager, "Low-density parity check codes", *IRE Trans. Inform.Theory*, vol. IT-8, January 1962, pp. 21-28.

[14] A. Roumy, S. Guemghar, G. Caire, and S. Verdu, "Design methods for irregular repeat-accumulate codes", *IEEE Trans. Inform. Theory*, vol. 50, no. 8, Aug. 2004, pp. 1711-1727.

[15] M. Yang, W. E. Ryan, and Y. Li, "Design of efficiently encodable moderate-length high-rate irregular LDPC codes", *IEEE Trans. Commun.*, vol. 52, no. 4, Apr. 2004, pp. 564-571.

[16] Johnson, S. J,Weller S. R,"Practical Interleavers for Systematic Repeat-Accumulate Codes", Vehicular Technology Conference, 2006. VTC 2006-Spring. IEEE 63rd, Vol 3,pp. 1358-1362

[17] J. Thorpe, "Low-Density Parity-Check(LDPC) Codes Constructed from Protographs", *IPN Progress Report*, August 15, 2003, pp. 42-154.

[18] M. Yang, W. E. Ryan, and Y. Li, "Design of efficiently encodable moderate-length high-rate irregular LDPC codes", *IEEE Trans. Comms.*, vol. 52, no. 4, pp. 564-571, April 2004.

[19] Etzion T. ,Trachtenberg A. and Vardy A. ,"Which codes have cycle-free Tanner graphs? ", *IEEE Transactions on Information Theory*,Volume 45, Issue 6, Sept. 1999, pp. 2173 - 2181

[20] T. J. Richardson, M. A. Shokrollahi, and R. L. Urbanke, "Design of capacity-approaching irregular low-density parity-check codes", *IEEE Trans. Inform. Theory*, vol. 47, no. 2, pp. 619-637, February 2001.

[21] S. Jin , G. Minglun, ZH. Zhongjin, L. Li and W. Zhongfeng, "A Memory Efficient FPGA Implementation of Quasi-Cyclic LDPC Decoder", Proceedings of the 5th WSEAS Int. Conf. on Instrumentation, Measurement, Circuits and Systems, 2006, pp. 218-223.

[22] W. ANGUS and W. L. LEE, "Efficient VLSI Parallel Implementation for LDPC Decoder", Conference of the 3th WSEAS Int. Conf., Taiwan, 2004.

[23] W. ANGUS and W. L. LEE, "Modified VLSI Implementation for Sequential LDPC Decoder", Conference of the 3thWSEAS Int. Conf., Taiwan, 2004.