US 20080222499A1

(19) **United States**

(12) **Patent Application Publication** (10) **Pub. No.: US 2008/0222499 A1**

HUNG et al. (43) **Pub. Date: Sep. 11, 2008**

(54) **CYCLIC COMPARISON METHOD FOR LOW-DENSITY PARITY-CHECK DECODER**

(76) Inventors: **Jui-Hui HUNG**, Tainan City (TW); **Jui-Hung Hung**, Tainan City (TW); **Sau-Gee Chen**, Hsinchu City (TW)

Correspondence Address:
SINORICA, LLC
528 FALLSGROVE DRIVE
ROCKVILLE, MD 20850 (US)

(21) Appl. No.: **11/741,693**

(22) Filed: **Apr. 27, 2007**

(30) **Foreign Application Priority Data**

Mar. 9, 2007 (TW) 96108173

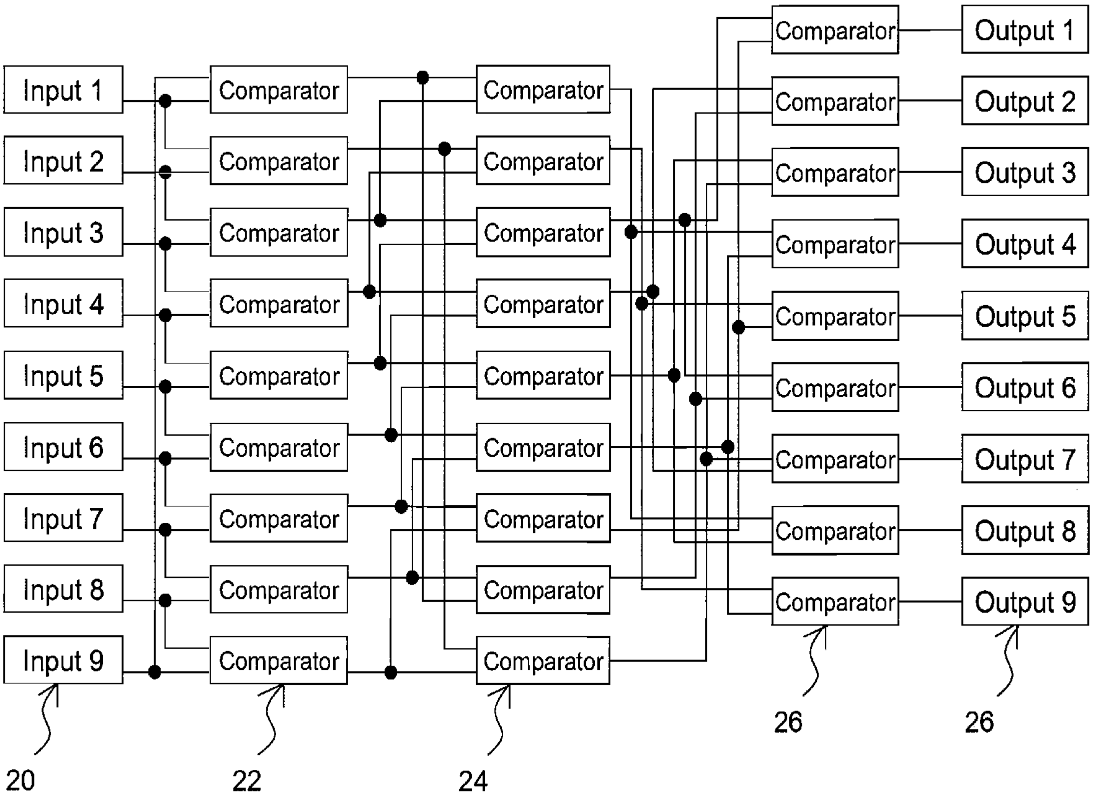
Publication Classification

(51) **Int. Cl.**
H03M 13/09 (2006.01)

(52) **U.S. Cl.** **714/801**

(57) **ABSTRACT**

The present invention discloses a cyclic comparison method for an LDPC decoder, which applies to the comparators used in an LDPC decoder. According to the cyclic comparison algorithm of the present invention, the nth element of the input k elements, wherein n=1, . . . , k, is sequentially removed by the corresponding comparator to obtain k first series. Next, pairs of two elements selected from the k elements are used to form k second series. The preceding step is repeated k×log$_2$(k−1) times to obtain k completion series. Either of one first series and one completion series contains (k−1) elements. The first series are compared with the completion series to determine whether they are identical. If they are identical, the process stops. If they are not identical, the abovementioned step is repeated to obtain new completion series. The cyclic comparison method of the present invention needs only k×log$_2$(k−1) comparisons to obtain completion series. Thus, the present invention can reduce the number of basic operations and can apply to any input number. Further, the present invention can be easily programmed and can achieve the optimal solution.



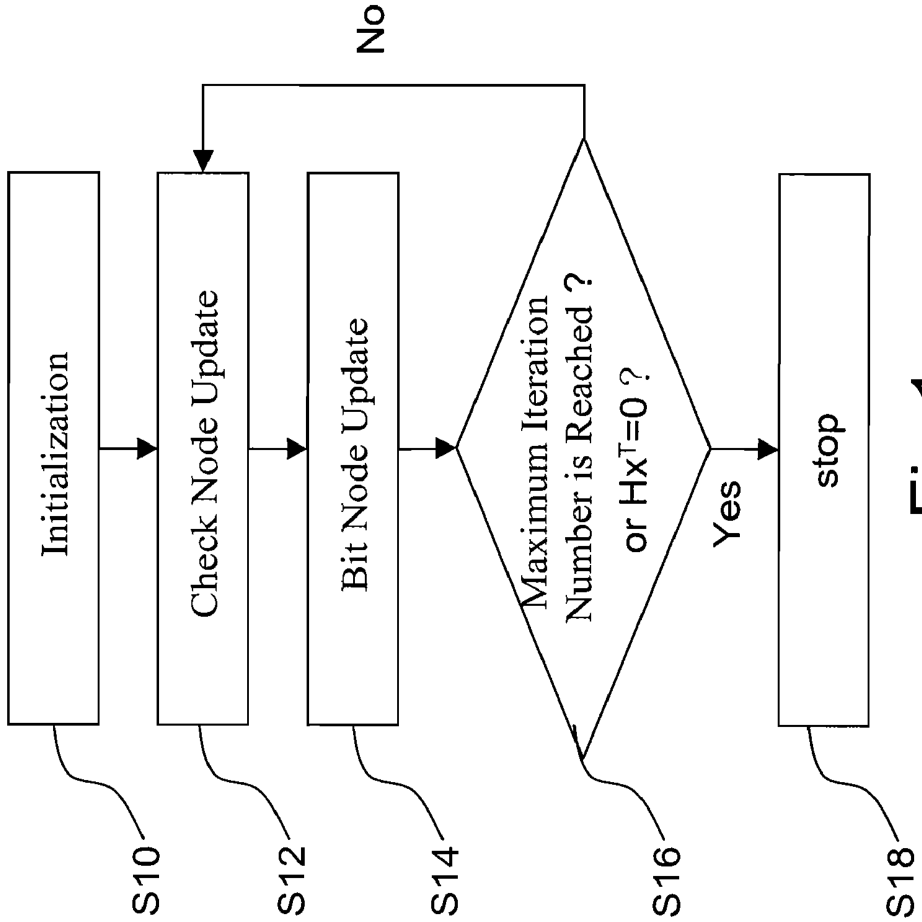

Initialization
Check Node Update
Bit Node Update
Maximum Iteration Number is Reached ? or $Hx^T=0$?
No
Yes
stop
S10
S12
S14
S16
S18

Fig.1

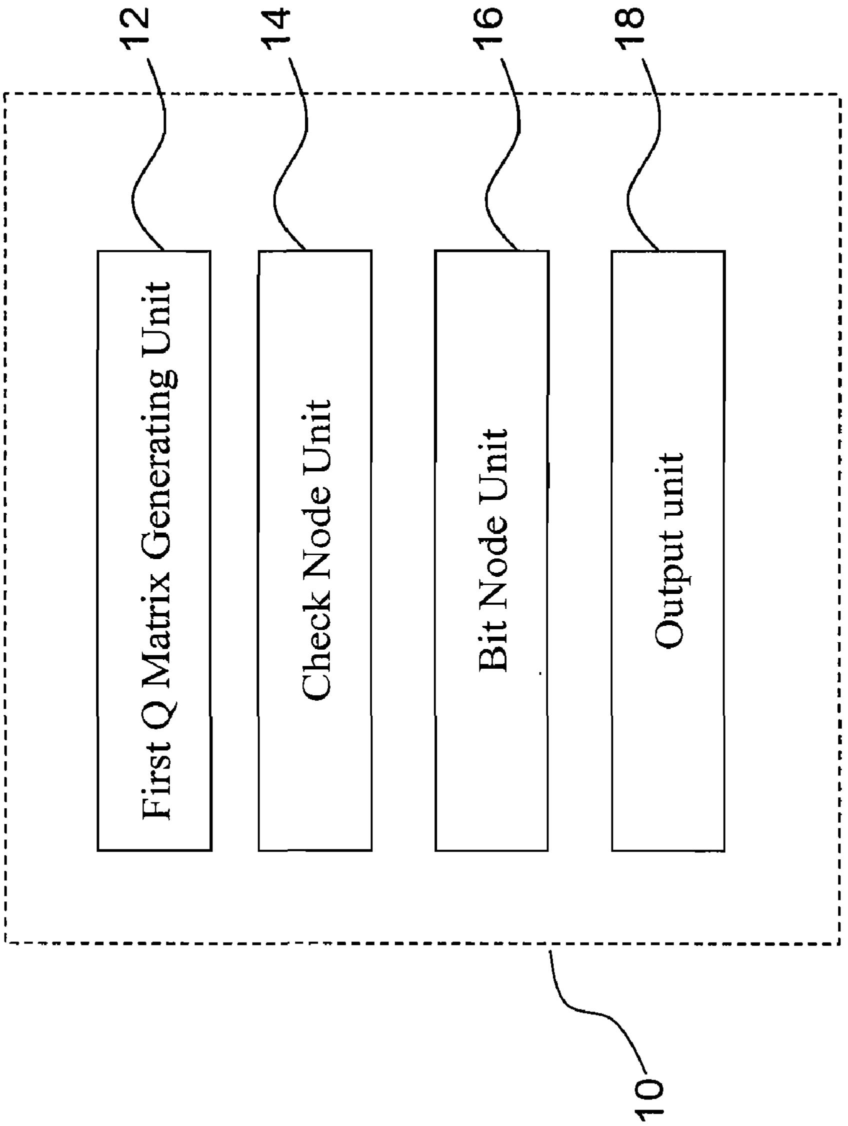

12
14
16
18
First Q Matrix Generating Unit
Check Node Unit
Bit Node Unit
Output unit
10

Fig.2

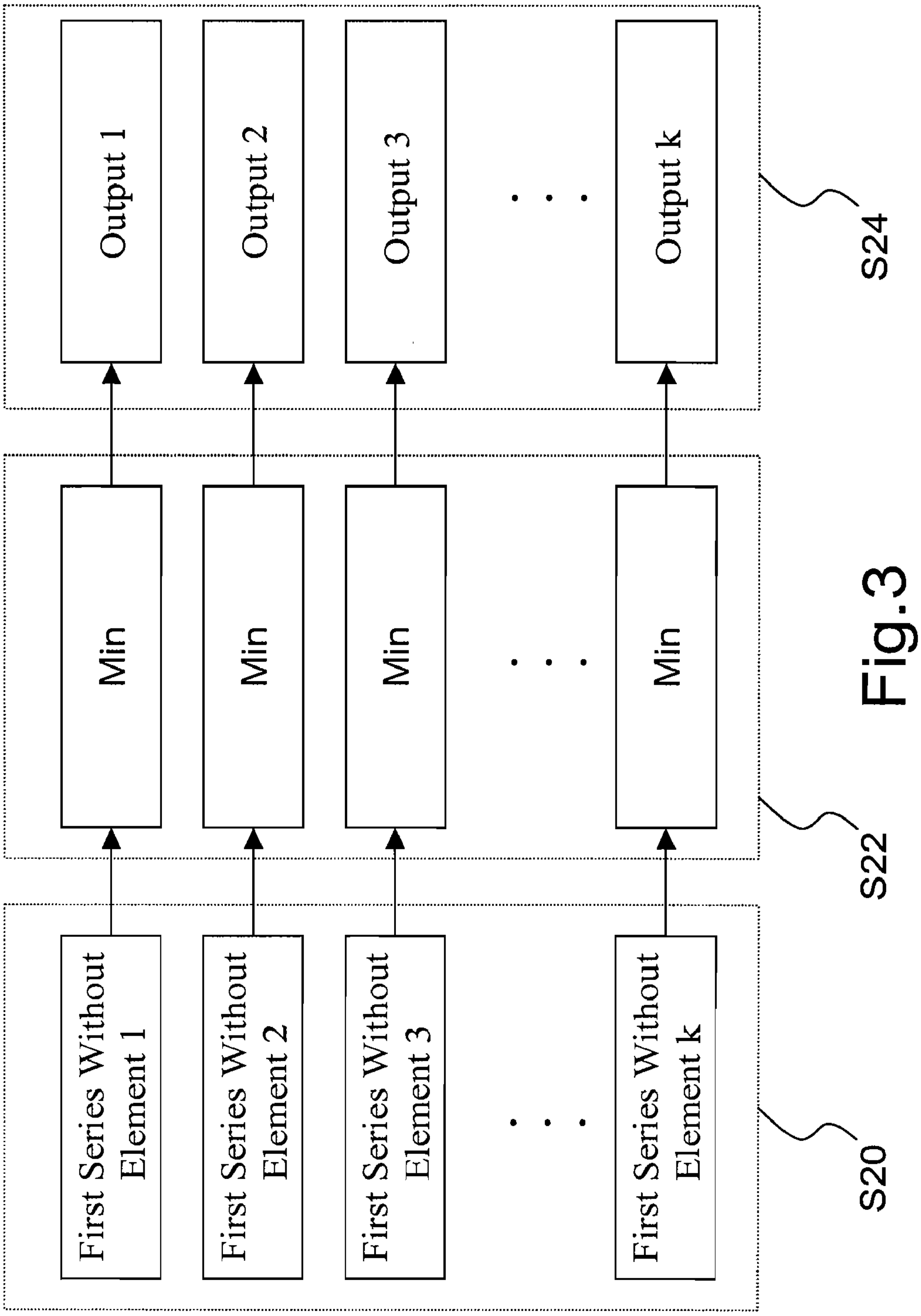

First Series Without Element 1
First Series Without Element 2
First Series Without Element 3
. . .
First Series Without Element k
S20
Min
Min
Min
. . .
Min
S22
Output 1
Output 2
Output 3
. . .
Output k
S24

Fig.3

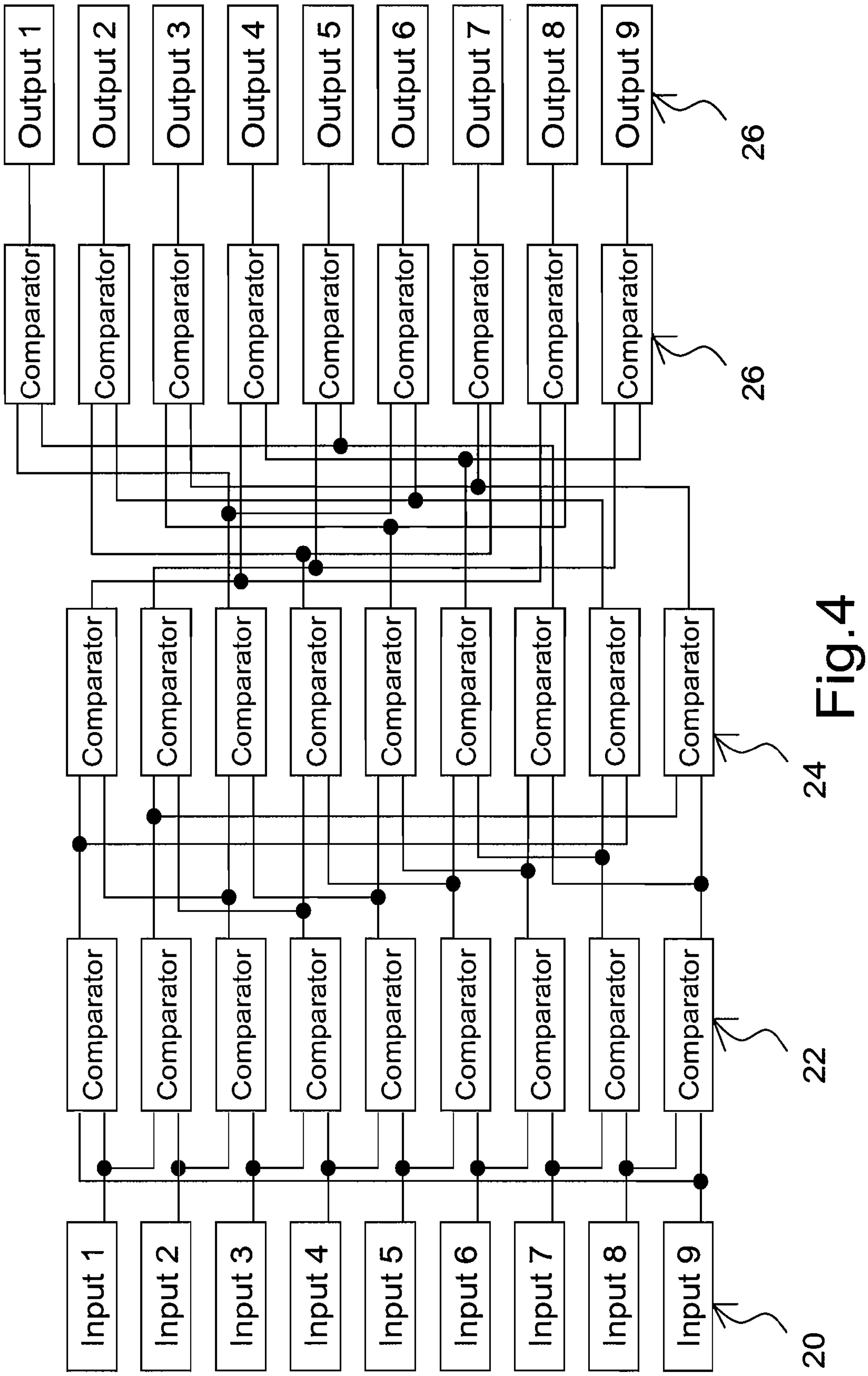

Output 1
Output 2
Output 3
Output 4
Output 5
Output 6
Output 7
Output 8
Output 9
26
Comparator
Comparator
Comparator
Comparator
Comparator
Comparator
Comparator
Comparator
Comparator
26
Comparator
Comparator
Comparator
Comparator
Comparator
Comparator
Comparator
Comparator
Comparator
24
Comparator
Comparator
Comparator
Comparator
Comparator
Comparator
Comparator
Comparator
Comparator
22
Input 1
Input 2
Input 3
Input 4
Input 5
Input 6
Input 7
Input 8
Input 9
20

Fig.4

CYCLIC COMPARISON METHOD FOR LOW-DENSITY PARITY-CHECK DECODER

BACKGROUND OF THE INVENTION

[0001] 1. Field of the Invention

[0002] The present invention relates to a technology of a low-density parity-check (LDPC) code for a transmission channel, particularly to a cyclic comparison method for an LDPC decoder.

[0003] 2. Description of the Related Art

[0004] The low-density parity-check (LDPC) code is a promising and very useful encoding technology with highly complicated calculation. The Min-Sum algorithm thereof can greatly reduce the hardware complexity of a decoder. However, a little performance loss may occur in the operation thereof. Therefore, it is an important subject how to reduce operational complexity via the Min-Sum algorithm without the penalty of performance.

[0005] The operation of an LDPC decoder is an iterative process. Refer to FIG. **1** a flowchart of the decoding process of an LDPC decoder. As shown in FIG. **1**, the decoder is initialized firstly in Step S**10**; next, check nodes and bit nodes are respectively updated in Step S**12** and Step S**14**; then, in Step S**16**, it is checked whether the iteration number exceeds a predetermined maximum iteration number or whether the decoded information bits satisfies the check matrix constraint: $Hx^T=0$; if the result is positive, the decoded bits are output, and the process ends, as shown in Step S**18**; it the result is negative, the process returns to Step S**12** to repeat the updates of the check nodes and the bit nodes. From the foregoing description, it is known that the check node unit (CNU) has a great influence on the decoder. The conventional technologies cannot apply to an arbitrary input number of a CNU but can only get an optimal result for a specified input number. In the conventional technologies, the greater the input number, the more the comparisons. Thus, the comparators need increasing, and the CNU operation speed slows down, which will consume more resource. Besides, the conventional technologies did not propose the detailed method for optimizing comparators, which makes them hard to outstand.

[0006] Accordingly, the present invention proposes a cyclic comparison method for an LDPC decoder to solve the abovementioned problems.

SUMMARY OF THE INVENTION

[0007] The primary objective of the present invention is to provide a cyclic comparison method for an LDPC decoder, wherein the cyclic comparison algorithm can be easily programmed, can approach the optimal solution and can accept any number of inputs in a low-density parity-check process.

[0008] Another objective of the present invention is to provide a cyclic comparison method for an LDPC decoder, wherein via the large set intersection between the compared series, k input elements need only $k\times\log_2(k-1)$ comparisons; thereby, the number of calculations can be reduced and the efficiency of the system is promoted.

[0009] Further another objective of the present invention is to provide a cyclic comparison method for an LDPC decoder, wherein when the number of the input elements is k, each element of the input elements is sequentially removed from the input elements to form k first series, and each first series has (k−1) elements; pairs of two elements selected from the k elements are used to form k series, and pairs of two series selected from the k series are used to form a plurality of series; the preceding step is repeated until obtaining k completion series with each completion series has (k−1) elements; the first series and the completion series are compared to determine whether they are identical; thereby, a fast comparison is achieved.

[0010] To achieve the abovementioned objectives, the present invention proposes a cyclic comparison method for an LDPC decoder, wherein k elements are input to a CNU for comparison. The cyclic comparison method for an LDPC decoder of the present invention comprises the following steps: (a) sequentially removing the elements respectively at from the first position to the kth position to obtain k first series with each first series having (k−1) elements, and outputting the first series and the minimum value of each first series; (b) utilizing pairs of two elements sequentially selected from the k elements input to the CNU to form k second series, and utilizing pairs of two second series sequentially selected from the k second series to form k third series; (c) utilizing pairs of two third series selected from the k third series to form a plurality series, and repeating the step until obtaining k completion series with each completion series containing (k−1) elements; and (d) comparing the completion series obtained in Step (c) and the first series obtained in Step (a) to determine whether they are identical; if they are identical, stopping the process and outputting the minimum value of each completion series; if they are not identical, doing Step (c) again to obtain new completion series.

[0011] Below, the embodiments are to be described in detail to make the objectives, technical contents, characteristics and accomplishments of the present invention easily understood.

BRIEF DESCRIPTION OF THE DRAWINGS

[0012] FIG. **1** is a flowchart of the decoding process of an LDPC decoder;

[0013] FIG. **2** is a diagram showing the architecture of an LDPC decoder;

[0014] FIG. **3** is a flowchart showing the process of generating first series of an LDPC decoder according to the cyclic comparison method of the present invention; and

[0015] FIG. **4** is a diagram schematically showing that the cyclic comparison method of the present invention is applied to an LDPC decoder.

DETAILED DESCRIPTION OF THE INVENTION

[0016] The present invention pertains to a cyclic comparison method for an LDPC decoder, which reduces the number of comparison operations via working out the largest common comparison set under the condition without changing the shortest critical path of the comparison operation, and which can also fast work out the result for any input number.

[0017] Refer to FIG. **2** a diagram showing the architecture of an LDPC decoding device **10**, which utilizes c-bit codewords transmitted via a channel and an LDPC matrix to perform LDPC decoding. The LDPC decoding device **10** comprises a first Q matrix generating unit **12**, a CNU (Check Node Unit) **14**, a (Bit Node Unit) BNU **16** and an output unit **18**. The first Q matrix generating unit **12** receives a codeword and a pxc parity-check matrix; when one element in one row of the parity-check matrix has the value of 1, the first Q matrix generating unit **12** substitutes the element with the code-bit value of the c-bit codeword at the corresponding position to form a first Q matrix. The CNU **14** receives the first Q matrix generated by the first Q matrix generating unit **12** and generates an R matrix. After receiving the R matrix, the BNU **16** generates a second Q matrix via substituting each non-zero element of the R matrix with the sum of the values of the non-zero elements except the non-zero element at the same column of the codeword; then, the BNU **16** outputs the second Q matrix to the CNU **14**. The output unit **18** receives the R matrix and sums up the elements of each column of the R matrix to determine a code bit and then outputs a decoded codeword with c code bits.

[0018] The cyclic comparison method for an LDPC decoder of the present invention is applied to the CNU **14**. K elements are input to the CNU for comparison. Refer to FIG. **3**. In Step **20**, the elements respectively at from the first position to the kth position are sequentially removed to obtain k first series with each first series having (k−1) elements. In Step **22**, the minimum value of each first series is obtained. In Step **24**, the first series and the minimum value of each first series are output. When k=9, the output 9 first series are {2,3, . . . ,8,9}, {1,3,4, . . . ,8,9}, {1,2,4,5, . . . ,8,9}, {1,2,3,5,6, . . . ,8,9}, {1,2,3,4,6,7,8,9}, {1,2, . . . ,5,7,8,9}, {1,2, . . . ,6,8,9}, {1,2, . . . ,6,7,9}, and {1,2,3, . . . ,7,8}. The minimum values ofthe first series are Min{2,3, . . . ,8,9}, Min{1,3,4, . . . ,8,9}, Min{1,2,4,5, . . . ,8,9}, Min{1,2,3,5,6, . . . ,8,9}, Min{1,2,3, 4,6,7,8,9}, Min{1,2, . . . ,5,7,8,9 }, Min{1,2, . . . ,6,8,9}, Min{1,2, . . . ,6,7,9}, and Min{1,2,3, . . . ,7,8}.

[0019] After the abovementioned k outputs have been generated from the k elements input to the CNU, pairs of two elements sequentially selected from the k elements input to the CNU are used to form k second series. When k=9, pairs of two elements sequentially selected from the first input **20** to the ninth input **20** are used to form 9 second series in the first-layer comparators **22**, including Min{1,2}, Min{2,3}, Min{3,4}, Min{4,5}, Min{5,6}, Min{6,7}, Min{7,8}, Min{8,9}, and Min{9,1}. Next, pairs of two second series sequentially selected from the 9 second series are used to form 9 third series in the second layer comparators **24**, including Min{9,1,2,3}, Min{1,2,3,4}, Min{2,3,4,5}, Min{3,4,5, 6}, Min{4,5,6,7}, Min{5,6,7,8}, Min{6,7,8,9}, Min{7,8,9, 1}, and Min{8,9,1,2}. Next, pairs of two third series selected from the 9 third series are used to form 9 fourth series, and each fourth series has 8 elements. Then, the fourth series are compared with the first series output beforehand to determine whether they are identical; if they are identical, the process stops; if they are not identical, the preceding step repeats.

[0020] In the abovementioned steps, all the values of the elements of each series are non-zero. The number of the elements of the second series, the third series or the succeeding series is equal to 2 to the nth power, wherein n is the number of the combinations for the series. Therefore, the completion series must possess an even number of elements. At the same time, the completion series are obtained via combining two third series and repeating the step until obtaining the series containing (k−1) elements. When the input number k is an even number, the lastly-output completion series should have (k−1) elements, and (k−1) is an odd number, which conflicts with that the completion series should possess an even number of elements. Therefore, when k is an even number, two third series containing a repeated element are combined to obtain one completion series having an odd number of elements. For example, when k=8, two third series {2,3,4,5} and {5,6,7,8} are combined to form the completion series {2,3,4,5,6,7,8}. In the present invention, the number of the required comparators is k×$\log_2$(k−1), and the number of the layers of comparators is $\log_2$(k−1), i.e. the number of the combinations each completion series has to pass through is $\log_2$(k−1).

[0021] In summary, the present invention proposes a cyclic comparison method for an LDPC decoder, wherein the cyclic comparison algorithm can be easily programmed, can approach the optimal solution and can accept any number of inputs in a low-density parity-check process. Further, via the large set intersection between the compared series, k input elements need only k×$\log_2$(k−1) comparisons; thereby, the number of calculations can be reduced, and the efficiency of the system is promoted.

[0022] Those described above are the preferred embodiments to exemplify the present invention. However, it is not intended to limit the scope of the present invention. Any equivalent modification or variation according to the spirit of the present invention is to be also included within the scope of the present invention.

What is claimed is:

1. A cyclic comparison method for a low-density parity-check decoder, which inputs k elements to a check node unit and compares said elements, comprising the following steps:

(a) sequentially removing one said element respectively at from first position to kth position to obtain k first series with each said first series having (k−1) elements, and outputting said first series and a minimum value of each said first series;

(b) utilizing two said elements sequentially selected from said k elements input to said check node unit to form k second series, and utilizing pairs of two said second series sequentially selected from said k second series to form k third series;

(c) repeating utilize pairs of two said third series to form a plurality series until obtaining k completion series with each said completion series containing (k−1) elements; and

(d) comparing said completion series obtained in Step (c) and said first series obtained in Step (a) to determine whether they are identical;

if they are identical, stopping process and outputting a minimum value of each said completion series; if they are not identical, doing Step (c) again to obtain new completion series until said new completion series are identical to said first series.

2. The cyclic comparison method for a low-density parity-check decoder according to claim **1**, wherein values of said elements are greater than 0.

3. The cyclic comparison method for a low-density parity-check decoder according to claim **1**, wherein either of said second series and said third series has an even number of said elements.

4. The cyclic comparison method for a low-density parity-check decoder according to claim **1**, wherein when k is an even number, two series having a repeated element are used to form one said completion series having an odd number of said elements.

5. The cyclic comparison method for a low-density parity-check decoder according to claim **1**, wherein in said first series, values of said elements are set to be 1 except said element with a second minimum value, and said minimum value substitutes said second minimum values.

6. The cyclic comparison method for a low-density parity-check decoder according to claim **1**, wherein a plurality of said first series forms a check matrix.

7. The cyclic comparison method for a low-density parity-check decoder according to claim **1**, wherein combinations and comparisons of said second series, said third series, said completion series, and the series formed in process generating said completion series are implemented with a plurality of comparators.

8. The cyclic comparison method for a low-density parity-check decoder according to claim **7**, wherein number of said comparators is $k \times \log_2(k-1)$.

9. The cyclic comparison method for a low-density parity-check decoder according to claim **7**, wherein said comparators output said minimum values.

10. The cyclic comparison method for a low-density parity-check decoder according to claim **1**, wherein number of the combination/comparison operations to obtain one said completion series is $\log_2(k-1)$.

* * * * *