An Efficient Implementation of LDPC Decoder with Partial Parallel Algorithm for DVB-S2 System

Suksoon Choi[1], Minhyuk Kim[2], Jongtae Bae[3], Taedoo Park[4], Namsoo Kim[5], and Jiwon Jung[6]
Department of Radio Communication Engineering, Korea Maritime University
#1, Dongsam-dong, Youngdo-gu, Busan, Korea
E-mail: [1]next_jeonpa@hhu.ac.kr, [2]squru@hotmail.com, [3]ms43bjt@hhu.ac.kr,
[4]bokddori@hhu.ac.kr, [5]skatn1983@hhu.ac.kr, [6]jwjung@hhu.ac.kr

Abstract— In this paper, we investigate the encoding and decoding method of the irregular LDPC (Low Density Parity Check) codes that offer diverse coding rates from 1/2 to 9/10 defined in the Digital Video Broadcasting (DVB-S2) standard. We study the efficient memory assignment and the implementing method in the LDPC codes.

1. Introduction

The high definition television (HDTV) satellite standard, known as the Digital Video Broadcasting (DVB-S2) transmission system employs a LDPC (Low Density Parity Check) coding technique as a channel coding scheme. The DVB-S2 transmission system requires the large block sizes (N=64800, 16200) and numerous iterations. There are ten code rates (R=1/4, 1/3, 1/2, 3/5, 2/,3, 3/4, 4/5, 8/9, 9/10) and four modulation methods (QPSK, 8-PSK, 16-APSK, 32-APSK). Therefore, this paper makes an analysis of the encoding structure and the decoding algorithm proposed by the HNS. In order to design the FPGA and ASIC, we were compared to implement the way regarding the completely parallel scheme and partially parallel scheme of the decoding structure concerned with the number of memory.

2. Encoder Structure in DVB-S2 Standard

Although the parity check matrix of the LDPC codes is sparse, the general generator matrix is required to encode. Therefore, we apply to the parity check matrix that consists of the following way. $H_{(N-K)xN} = \lfloor A_{(N-K)xK} B_{(N-K)x(N-K)} \rfloor$. B denoted the scheme of the Fig. 1 and the Low Triangular parity check Matrix.

$$B = \begin{bmatrix} 1 & & & & & & & & \\ 1 & 1 & & & & 0 & & & \\ & 1 & 1 & & & & & & \\ & & 1 & \bullet & & & & & \\ & & & & \bullet & & & & \\ & & & & & \bullet & & & \\ & 0 & & & & & \bullet & & \\ & & & & & & & 1 & \\ & & & & & & & 1 & 1 \end{bmatrix}$$

Figure 1. General UEP system block diagram.

Matrix A minimizing the cycle-6 avoids the cycle-4 and composes the Matrix sparsely. It distributes the respective initial value introduced [1] to do randomly. The LDPC codes encode the size of the information block. The N is the size of the encoder and the K is the size of the information bits. The M is the size of the redundant bits and has the number of N-K.

The method of encoder refers to the following procedure.

Stage 1. All parity bits are initialized.
Stage 2. Every address of parity bits is added.
Stage 3. To calculate the B sub matrix of Low Triangular parity check Matrix, it is applied to the Eq(1).

$$p_n = p_n + p_{n-1} \; (n = 1,2,\ldots) \qquad (1)$$

In the stage 2, the M_Address represents the address that retains the 1's position of respectively information bits. Table 1 shows the N and K that offer each of the code rates given the DVB-S2. Table. 1 shows the N,K that offer each of the code rates given the DVB-S2.

Table 1. Simulation parameters

N=16200					
R	N	K	Maximum Column-weight	Maximum Row-weight	q
1/4	16200	3240	12	4	36
1/3	16200	5400	12	5	30
2/5	16200	6480	12	6	27
1/2	16200	7200	8	7	25
3/5	16200	9720	12	11	18
2/3	16200	10800	13	10	15
3/4	16200	11880	12	13	12
4/5	16200	12600	3	13	10
5/6	16200	13320	13	19	8
8/9	16200	14400	3	27	5
9/10	N/A	N/A	N/A	N/A	N/A

N=64800					
R	N	K	Maximum Column-weight	Maximum Row-weight	q
1/4	64800	16200	12	4	135
1/3	64800	21600	12	5	120
2/5	64800	25920	12	6	108
1/2	64800	34200	8	7	90
3/5	64800	38880	12	11	72
2/3	64800	43200	13	10	60
3/4	64800	48600	12	14	45
4/5	64800	51840	11	18	36
5/6	64800	54000	13	22	30
8/9	64800	57600	4	27	20
9/10	64800	58320	4	30	18

This work was supported by Defense Acquisition Program Administration and Agency for Defense Development under the contract UD070054AD

3. Decoder Structure in DVB-S2 Standard

We can divide calculation into 3 steps. Step 1 is initialization that calculates channel estimation value with received bits, Step 2 is CNU that calculates Check Node, Step 3 is BNU that calculates Bit Node Update.

Step 1. Initialization as shown in Fig. 2(a) and using Eq(2).

$$u_n = -L_c \cdot r_n (L_c = \frac{2}{\sigma^2}), \qquad n = 0,1,\ldots,N-1, \tag{2}$$

Step 2. Check Node Update (CNU)

Fig. 2(b) shows that it is calculated probability of the incoming messages to one check node.
If it has dc row weight, bits' probability incoming to each check node is as same as Eq(3).

$$w_{k\to n_1} = g(v_{n_1\to k}, v_{n_2\to k}, \ldots, v_{n_{dc}\to k}) \tag{3}$$

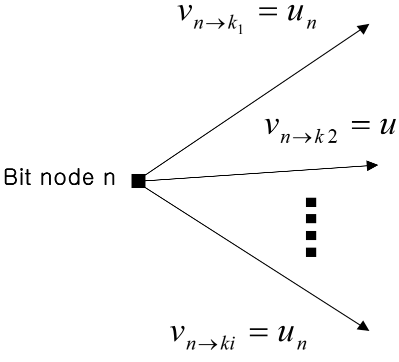
$$\begin{aligned} g(a,b) &= sign(a)\times sign(b)\times\{\min(|a|,|b|)\} + LUT_g(a,b) \\ &= sign(a)\times sign(b)\times\{\min(|a|,|b|) - LUT(|a-b|)\} \end{aligned}$$

$$LUT(|a-b|) = \begin{cases} 2 & |a-b| \le 2 \\ 1 & 3 \le |a-b| \le 6 \\ 0 & |a-b| \ge 7 \end{cases}$$

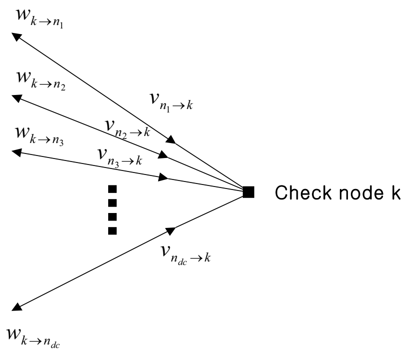
It shows $LUT_g(a,b) = \log(1+e^{-|a+b|}) - \log(1+e^{-|a-b|})$

Step 3. Bit Node Update (BNU)
After each check node has been updated using dv bits which are connected to check node, we must calculate bit node's probability proper for each column.

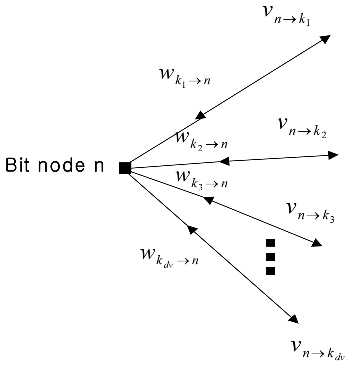
$$v_{n\to k_i} = u_n + \sum_{j\neq i} w_{k_j\to n} \tag{4}$$



(a) Initialization



(b) BNU



(c) CNU

Figure 2. LDPC decoding procedure

4. Partial Parallel Algorithm for LDPC Decoder

In this section, we propose a decoder design scheme to construct the rate - 1/2 LDPC codes. The decoder implementation is classified into largely using the FPGA board or the ASIC. Such fully parallel decoder could achieve tremendous high decoding speed. However, the primary disadvantage of fully parallel design leads to increasing not only the code length but also high implementation complexity causing the excess in the capacity of a chip. Therefore, we can know that the partial parallel scheme is the efficient selection to achieve appropriate trade-offs between hardware complexity and decoding speed. Fig. 3 illustrates the entire block diagram for designing the decoder on the FPGA.

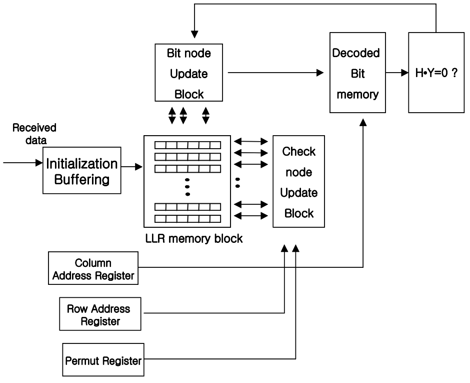


Figure 3. LDPC decoder structure

The partially parallel decoder consists of entirely the assigned block that multiplied received data by channel transmit probability in memory and the CNU (Check Node Update) block and the BNU (Bit Node Update) block. LLR memory structure that is equivalent to the column address proposed by HNS [1] in DVB-S2 comprises the total 90 blocks in which each block has 360 data as shown in Fig. 4.

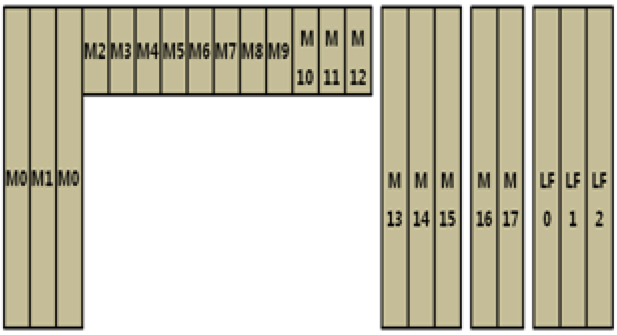


Figure 4. LLR memory structure

To facilitate CNU processing, the Rom_Index denotes whether what memories, among vertical 90 blocks, have same column address to calculate CNU. At the memories, Address_Index directs what block has same column address. And Permut_Index informs the point where block is started to read. Fig. 5 shows the inner block structure that LLR memory structure in Fig. 5 enlarges.

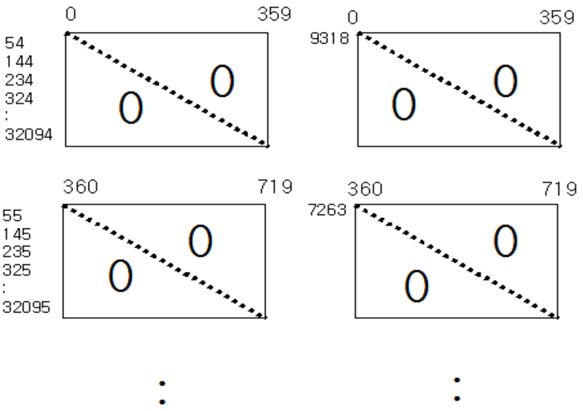


Figure 5. Inner block structure

5. The partially parallel structure

We store row block in different memory in principle. The column block saves the same memory which the number is equivalent to the total 'eight' like the number of Column weight in LDPC codes (N=64800,rate-1/2). The number of entire data using one of the memories is composed to 32400 x 6 bits. Saving 144 bits at one address, we require 1350 addresses. As the memory of chip implemented is able to process 144 bits x 4K. In the CNU mode, decoder performs the computation of all the check nodes. Calculating the CNU is able to access 8 memories in parallel. The Decoder scheme is denoted in detail in Fig. 6 from memory access point of view given in Fig. 4.

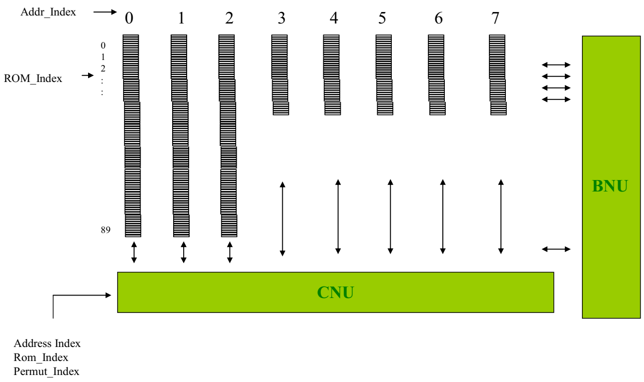


Figure 6. Decoder structure for implementation of partial parallel

Address Index block is divided into 90 blocks that save 360 data. Consequently 90 memory blocks can save the total of 32400 received data. This method is able to store easily the initialization and buffering block. As Fig. 7, describes the scheme calculating the BNU, we need to know the Address_Index, the Rom_Index and the Permut_Index to process the scheme of the CNU.

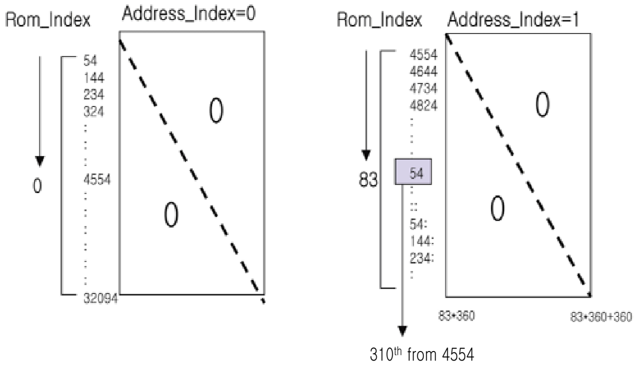


Figure 7. Example (CNU excess)

Both the CNU and the BNU block are divided into each block having the 360 data(Figure 8.). They are calculated as the number of row weight at once. Finishing the one CNU block, 90 CNU blocks is processed one by one in serial. To calculate the CNU block is required to acknowledge the respective block index and the index that displays the beginning point in each block. The operation performed in the BNU is very simple that it has just the block index owing to knowing at the beginning point in each block.

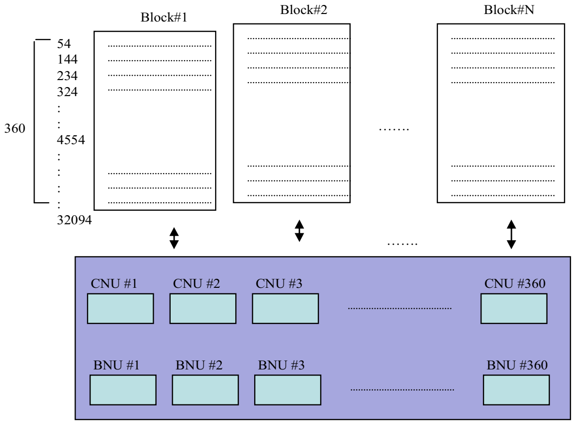


Figure 8. Partially parallel decoder scheme

The number of CNU and BNU block processing the partially parallel algorithm and the fully parallel algorithm are listed in Table 2.

TABLE 2. The number of block according to partially parallel

R	A matric	B matric q*2 (q=M/360)	Receive bir save block	Total block
1/4	270	135*2=270	180(64800/360)	720
1/3	360	120*2=240	180	180
2/5	432	108*2=216	180	822
1/2	54*3+36*8=450	90*2=18	180	810
3/5	548	72*2=144	180	872
2/3	12*13+48*3+60*3=480	60*2=120	180	780
3/4	540	45*2=90	180	810
4/5	567	36*2=72	180	819
5/6	600	30*2=60	180	840
8/9	480	20*2=40	180	700
9/10	540	18*2=36	180	756

6. Conclusion

In this paper, we analyzed LDPC codes of encoding and decoding algorithm at the respective code rates in DVB-S2 standard proposed HNS[1]. On the basis of analysis we compared the performance. Also we have studied the two methods such as the fully parallel decoder and the partially parallel decoder concerning decoding method required the number of memory to design implementation on the FPGA or ASIC. In reference to analysis consequence, we are able to implement the LDPC codes on FPGA.

References

[1] Draft ETSI EN 302 307, Digital Video Broadcasting(DVB); News Gathering and other broadband satellite applications, 2004. 6.

[2] R. G. Gallager, "Low-Density Parity-Check Codes," IRE trans. information theory, vol.8, PP.21-28, 1962.

[3] D. J. C. Mackay and R. M. Neal, "Near Shannon Limit Performance of Low-Density Parity-Check Codes," Electron. Letter, Vol.32, PP. 1645-1646, Aug. 1996.

[4]M. Sipser and D.A.Spielman, "Expander Codes, "IEEE Trans. Information Theory, vol.42, pp.1720-1722, Aug. 1996