



## Article

# Reconfigurable Low-Density Parity-Check (LDPC) Decoder for Multi-Standard 60 GHz Wireless Local Area Networks

Cheng-Hung Lin <sup>1,2,\*</sup> , Hsin-Hao Su <sup>2</sup>, Tang-Syun Chen <sup>2</sup> and Cheng-Kai Lu <sup>3</sup> <sup>1</sup> Biomedical Engineering Research Center, Yuan Ze University, Taoyuan City 32003, Taiwan<sup>2</sup> Department of Electrical Engineering, Yuan Ze University, Taoyuan City 32003, Taiwan; s1024655@mail.yzu.edu.tw (H.-H.S.); s1064626@mail.yzu.edu.tw (T.-S.C.)<sup>3</sup> Department of Electrical Engineering, National Taiwan Normal University, Taipei City 10610, Taiwan; cklu@ntnu.edu.tw

\* Correspondence: chlin@saturn.yzu.edu.tw

**Abstract:** In this study, a reconfigurable low-density parity-check (LDPC) decoder is designed with good hardware sharing for IEEE 802.15.3c, 802.11ad, and 802.11ay standards. This architecture flexibly supports 12 types of parity-check matrix. The switching network adopts an architecture that can flexibly switch between different inputs and achieves a low hardware complexity. The check node unit adopts a switchable 8/16/32 reconfigurable structure to match different row weights at different code rates and uses the normalised probability min-sum algorithm to simplify the structure of searching for the minimum value. Finally, the chip is implemented using the TSMC 40 nm CMOS process, based on the IEEE 802.11ad standard decoder, extended to support the IEEE 802.15.3c standard, and upwardly compatible with the next-generation advanced standard IEEE 802.11ay. The chip core size was 1.312 mm × 1.312 mm, the operating frequency was 117 MHz when the maximum number of iterations was five with the power consumption of 57.1 mW, and the throughput of 5.24 Gbps and 3.90 Gbps was in the IEEE 802.11ad and 802.15.3c standards, respectively.



**Citation:** Lin, C.-H.; Su, H.-H.; Chen, T.-S.; Lu, C.-K. Reconfigurable Low-Density Parity-Check (LDPC) Decoder for Multi-Standard 60 GHz Wireless Local Area Networks. *Electronics* **2022**, *11*, 733. <https://doi.org/10.3390/electronics11050733>

Academic Editor: José Santa

Received: 28 January 2022

Accepted: 25 February 2022

Published: 26 February 2022

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

**Keywords:** error correction code; low-density parity-check code; min-sum algorithm; ASIC implementation

## 1. Introduction

With the rapid development of multimedia equipment and the advancement of technology, ultra-high-quality equipment with a resolution of 3840 × 2160 (4K2K) pixels, such as ultra-high-definition television (UHDTV) projectors, has been developed. Most products use high-definition multimedia interface (HDMI) lines as transmission media, which is expensive and has length limitations; therefore, wireless transmission is an ideal solution. Equipment for augmented reality (AR) or virtual reality (VR), mirroring mobile devices, etc., also tend to use wireless transmission. Thus, 60 GHz wireless transmission plays an important role in the fifth-generation (5G) era, where a high transmission rate, large data volume, and low latency are emphasised.

In communication systems, forward error correction (FEC) is used to protect data from errors caused by noise interference during transmission. After the data are encoded by the error correction code, even if noise interference occurs in the transmission channel during transmission, the error message can be recovered at the receiving end through the decoding process. In 1962, Gallager invented a low-density parity-check (LDPC) [1] code, and after MacKay added the concept of iterative processing in 1999 [2], the decoding performance was very close to the Shannon limit. Because LDPC codes have excellent error correction performance, they are widely used in wireless communication systems, including the IEEE 802.11ad/ay standard adopted by Wireless Gigabit (WiGig), the IEEE 802.15.3c standard adopted by Wireless HD (WiHD), and the IEEE 802.11ax standard adopted by Wi-Fi. Furthermore, LDPC codes can be considered to improve the quality of transmission of critical applications using the 2.4 GHz-based Zigbee/Bluetooth communications [3,4]

and artificial-intelligence-assistant wireless sensor networks [5–7] because of their excellent error correction performance and efficient hardware acceleration. Additionally, the LDPC code was proposed in the 15th edition of the 5G new-radio (NR) specification published in 2018. Recently, several studies proposed hardware-efficient LDPC encoders [8,9] and decoders [10,11] for high-throughput 5G NR communication systems.

In recent years, many communication standards that adopted the LDPC code have been introduced in HD video wireless transmission, as shown in Table 1. It can be seen that, compared with the crowded 2.4 GHz band, the standard applied on the 60 GHz frequency band has the advantage of a high transmission rate. While WiHD uses the IEEE 802.15.3c communication standard and WiGig uses IEEE 802.11ad and its upgrade standard IEEE 802.11ay, the communication specifications of the three standards are different. If hardware sets are designed independently for each communication standard, the associated costs are extremely high. Thus, the previous study in [12] has been proposed a key reconfigurable processing unit of LDPC decoding for the IEEE 802.15.3c and IEEE 802.11ad standards.

**Table 1.** Current wireless transmission technology for 60GHz wireless local area networks.

| Application                   | WiHD                               | WiGig                          |                                |
|-------------------------------|------------------------------------|--------------------------------|--------------------------------|
| Transmission standard         | IEEE 802.15.3c                     | IEEE 802.11ad                  | IEEE 802.11ay                  |
| Working frequency             | 60 GHz                             | 60 GHz                         | 60 GHz                         |
| FEC code                      | LDPC codes<br>(Code rate: 1/2–7/8) | LDPC codes<br>(Rate 1/2–13/16) | LDPC codes<br>(Rate 1/2–13/16) |
| Theoretical transmission rate | 10~28 Gbps                         | 7~28 Gbps                      | 20~100 Gbps                    |

In this study, we have further designed and implemented a complete LDPC decoder based on the IEEE 802.11ad standard, extended to support the IEEE 802.15.3c standard and upwardly compatible with the IEEE 802.11ay standard, with low hardware cost, low power consumption, and high throughput. To our best knowledge, this study presents the first reconfigurable multimode LDPC decoder architecture that flexibly supports 12 LDPC matrices of the IEEE 802.15.3c, IEEE 802.11ad, and IEEE 802.11ay standards for HD video wireless transmission and provides sufficient details through detailed architecture design and the prototyping chip implementation. To support different standards, block-layer divisions of the matrices in the different standards are initially proposed to achieve reconfigurability and good hardware sharing for the reconfigurable LDPC decoding. In order to match the different row weights of different LDPC matrices, a switchable 8/16/32 hardware-shared structure is subsequently proposed for the key computational units, memories, and switching network and employed in the reconfigurable LDPC decoder architecture. The designed switching network flexibly switches between different inputs and achieves low hardware complexity. Compared with the traditional switching network, the designed switching network only requires 0.08% look-up-table bits to reconfigure the switches and support the multiple standards. The reconfigurable multimode LDPC decoder has been implemented using the TSMC 40 nm CMOS process in a core size of 1.72 mm<sup>2</sup> with the power consumption of 57.1 mW and throughput of 5.24 Gbps at the maximum operating frequency of 117 MHz in the IEEE 802.11ad standard. Additionally, the throughput of 3.9 Gbps and power consumption of 57.1 mW are achieved at the same operating frequency in the IEEE 802.15.3c standard. Compared with the LDPC decoders that support the individual standard, the reconfigurable multimode LDPC decoder implementation achieves approaching area efficiency and energy efficiency to alternatively support the IEEE 802.15.3c, IEEE 802.11ad, and IEEE 802.11ay standards.

The rest of this study is organised as follows. In Section 2, the characteristics and decoding of LDPC code are introduced. In Section 3, LDPC decoding is evaluated using the matrices of three standards for the 60 GHz wireless local area networks. In addition, for reconfigurability, the matrixes are divided into block layers. Section 4 describes the proposed decoder architecture in detail, including the computational units, switching network, and memory. Section 5 presents the VLSI implementation results of the proposed LDPC decoder and compares them with other related works. Finally, Section 6 concludes the study.

## 2. Fundamentals of LDPC Code and Decoding

The LDPC code is a type of linear block code composed of a sparse matrix. The sparse matrix is a parity-check matrix  $H$  composed of mostly 0's and a lesser number of 1's. There are  $N$  columns and  $M$  rows in the  $H$  matrix, and the code rate is defined as  $R = (N - M)/N$ . In the  $H$  matrix, each row represents a check node (CN), and the number of 1's in each row is called the row weight ( $w_r$ ); each column represents a variable node (VN), and the number of 1's in each column is called the column weight ( $w_c$ ). The 1 in the  $H$  matrix also represents the exchange of data between the CN and VN, as shown in Figure 1.

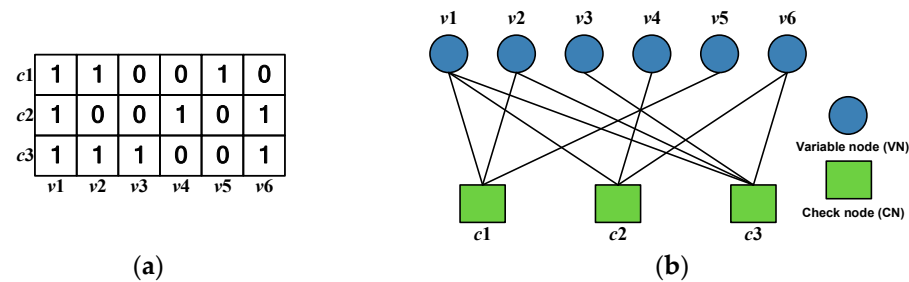


Figure 1. Example of (a)  $H$  matrix mapping to the (b) Tanner graph.

The quasi-cyclic (QC) LDPC code [13] is a common method for hardware implementation of LDPC decoding because it achieves different parallelisms in decoding with greater ease and enables easier memory access owing to its regularity. Figure 2 shows the QC-LDPC  $H$  matrix with  $R = 13/16$  in the IEEE 802.11ad standard. Each block is a submatrix with an expanding factor  $z$ . The blank block is a  $z \times z$  zero matrix. The number represents the number of shifts in the  $z \times z$  identity matrix to the right. The entire matrix can be expressed as  $m \times n$  where  $m = M \times z$  and  $n = N \times z$ .

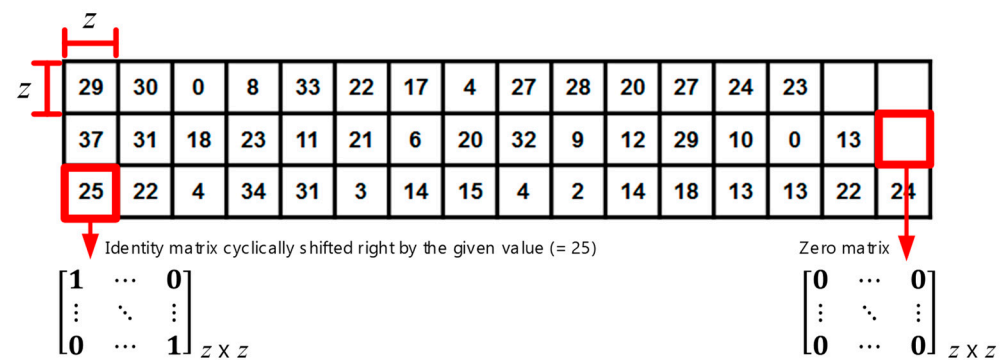


Figure 2. Example of QC-LDPC matrix.

The recent soft and hard decoding algorithms of LDPC codes have been significantly reviewed and summarized in [14]. The original soft decoding algorithm is the sum-product algorithm (SPA) [2], which has excellent error correction performance; however, the hardware implementation complexity is high. The normalised min-sum algorithm (NMSA) [15] instead of the SPA was widely used in chip implementations because of its low hardware complexity and good error-correction capabilities [16]. In terms of decoding, the iterative layer decoding schedule [17] was utilised, which includes two operations, CN and VN, and the decoding process, as shown in Figure 3. After receiving the channel information, the decoder starts iterative decoding. In the NMSA, we initially define  $y_j$  as the received channel information,  $L_{init,j}$  as initial log-likelihood ratio (LLR) message,  $Q_{i,j}^k$  as prior message,  $R_{i,j}^k$  as extrinsic message, and  $L_j$  as posterior message, where  $i$  is the index of the row of  $H$ ,  $j$  is the index of the column of  $H$ , and  $k$  is the index of the decoding iteration. The NMSA includes four steps, and the equations are described as follows.

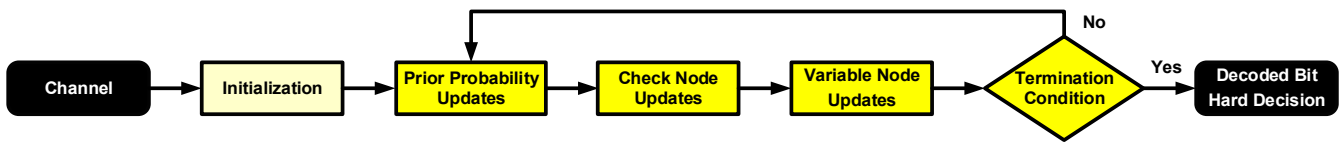


Figure 3. Flowchart of iterative LDPC decoding.

1. Initialization: The decoder receives each  $j$ th channel message  $y_j$  to initialise  $L_{init, j}$ .

$$L_{init, j} = \log \left( \frac{P(x_j = 0 | y_j)}{P(x_j = 1 | y_j)} \right). \quad (1)$$

2. Prior message updates: If  $k = 1$ ,  $L_j$  is updated as  $L_{init, j}$  and  $R_{i,j}^0$  is set to zero.

$$Q_{i,j}^k = L_j - R_{i,j}^{k-1}. \quad (2)$$

3. CN (extrinsic message) updates:

$$R_{i,j}^k = \alpha \cdot \left( \prod_{j' \in N(c) \setminus j} \text{sign}(Q_{i,j'}^k) \right) \left( \min_{j' \in N(c) \setminus j} |Q_{i,j'}^k| \right), \quad (3)$$

where  $\alpha$  is a normalisation factor.

4. VN (posterior message) updates:

$$L_j = Q_{i,j}^{k-1} + \sum_{i \in M(v)} R_{i,j}^k. \quad (4)$$

The steps 2–4 iteratively continues until the maximum number of iterations is reached. When the iteration terminates, a hard decision is made by

$$h_j = \begin{cases} 0, & \text{if } L_j \geq 0 \\ 1, & \text{if } L_j < 0 \end{cases}. \quad (5)$$

For reference, the study in [18] extends single-decoder decoding to parallel decoding with multiple sub-decoders and improves decoding performance of an LDPC code.

To reduce the hardware complexity of independently designing a set of decoders for different standards, this study proposes that it can be used in IEEE 802.11ad, IEEE 802.15.3c, and IEEE 802.11ay multimode LDPC decoders. Instead of using the NMSA, this study used the normalised probability min-sum algorithm (NPMSA) [19], which has low hardware complexity. In general, Equation (3) is the critical step with the highest computational complexity. To further simplify the computational complexity, the NPNSA was used to simplify the comparator in the sorter. The original comparator compares the two input data (IN\_1 and IN\_2) and outputs the minimum value (Min) and the second minimum value (2nd Min), as shown in Figure 4a. However, the simplified comparator discards the information of the second minimum value and outputs only that of the first minimum value, as shown in Figure 4b. According to this method, the second minimum value obtained was probably correct (Prob. Min), as shown in Figure 5. Dividing the input of the sorter into  $G$  groups and using  $G$  to 2 comparators in the last stage of the comparators slightly reduces the performance, but it can significantly reduce the hardware complexity of the operation. For reference, several alternative methods [11,20,21] were proposed to reduce the gap between the accurate second minimum and probabilistic second minimum and recover the decoding capability.



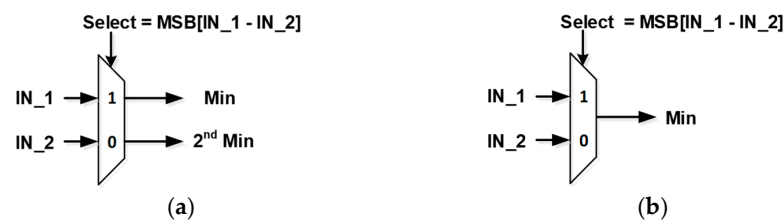


Figure 4. (a) 2-to-2 comparator and (b) 2-to-1 comparator.

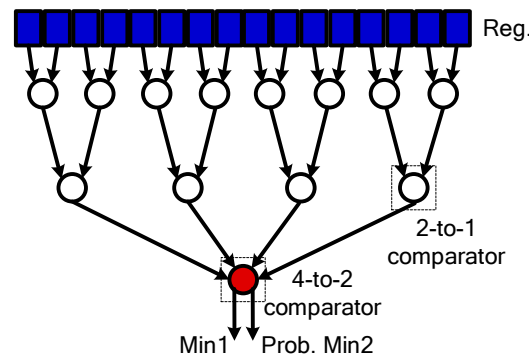


Figure 5. The architecture of the NPMSA minimum sorter.

### 3. Proposed LDPC Decoding for the Multi-Standard 60 GHz Wireless Local Area Networks

To design a set of hardware-sharing decoders, it is necessary to understand the matrix parameters in all standards and to identify the parts that can be shared in different standards.

#### 3.1. Standard Parameters and Matrix Configuration

The QC-LDPC matrix used by IEEE 802.11ad has  $R = 1/2, 5/8, 3/4$ , and  $13/16$ , as shown in Figure 6a,b.  $M$  changes according to distinct  $R$ , that is, 8, 6, 4, and 3.  $N$  is fixed at 16 and  $z$  is 42. Therefore,  $n$  is  $16 \times 42 = 672$ . The QC-LDPC matrix used in the IEEE 802.15.3c standard also has  $R = 1/2, 5/8, 3/4$ , and  $7/8$ , as shown in Figure 7a,b.  $N$  is fixed at 32, and  $z$  is 21. It can be observed that  $n$  is  $32 \times 21 = 672$ , as for IEEE 802.11ad.

|    |    |   |    |    |    |    |    |   |    |    |    |    |    |    |    |
|----|----|---|----|----|----|----|----|---|----|----|----|----|----|----|----|
| 40 | 38 |   | 13 |    | 5  |    | 18 |   |    |    |    |    |    |    |    |
|    | 36 |   | 31 |    | 7  |    | 34 |   | 10 | 14 |    |    |    |    |    |
| 34 | 35 |   | 27 |    |    |    | 30 | 2 | 1  |    |    |    |    |    |    |
|    | 27 |   | 18 |    | 12 | 20 |    |   | 15 | 6  |    |    |    |    |    |
| 35 | 41 |   | 40 |    | 39 |    | 28 |   |    | 3  | 28 |    |    |    |    |
|    | 31 |   | 23 |    | 21 |    | 20 |   | 12 |    |    | 0  | 13 |    |    |
| 29 |    | 0 |    |    | 22 |    | 4  |   | 28 |    | 27 |    | 23 |    |    |
|    | 22 |   | 24 | 31 |    | 14 |    | 4 |    |    |    | 13 |    | 22 | 24 |

(a)

|    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 20 | 36 | 34 | 31 | 20 | 7  | 41 | 34 |    | 10 | 41 |    |    |    |    |    |
| 30 | 27 |    | 18 |    | 12 | 20 | 14 | 2  | 25 | 15 | 6  |    |    |    |    |
| 35 |    | 41 |    | 40 |    | 39 |    | 28 |    |    | 3  | 28 |    |    |    |
|    | 31 |    | 23 |    | 21 |    | 20 |    | 9  | 12 |    |    | 0  | 13 |    |
| 29 |    | 0  |    |    | 22 |    | 4  |    | 28 |    | 27 | 24 | 23 |    |    |
|    | 22 |    | 34 | 31 |    | 14 |    | 4  |    |    |    |    |    | 22 | 24 |

(b)

|    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 35 | 19 | 41 | 22 | 40 | 41 | 39 | 6  | 28 | 18 | 17 | 3  | 28 |    |    |    |
| 29 | 30 | 0  | 8  | 33 | 22 | 17 | 4  | 27 | 28 | 20 | 27 | 24 | 23 |    |    |
| 37 | 31 | 18 | 23 | 11 | 21 | 6  | 20 | 32 | 9  | 12 | 29 |    | 0  | 13 |    |
| 25 | 22 | 4  | 34 | 31 | 3  | 14 | 15 | 4  |    | 14 | 18 | 13 | 13 | 22 | 24 |

(c)

|    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 29 | 30 | 0  | 8  | 33 | 22 | 17 | 4  | 27 | 28 | 20 | 27 | 24 | 23 |    |    |
| 37 | 31 | 18 | 23 | 11 | 21 | 6  | 20 | 32 | 9  | 12 | 29 | 10 | 0  | 13 |    |
| 25 | 22 | 4  | 34 | 31 | 3  | 14 | 15 | 4  | 2  | 14 | 18 | 13 | 13 | 22 | 24 |

(d)

Figure 6. IEEE 802.11ad parity check matrix with  $R =$  (a)  $1/2$ , (b)  $5/8$ , (c)  $3/4$ , and (d)  $13/16$ .

|    |    |    |    |    |    |    |   |    |   |    |    |    |    |    |    |   |    |  |    |    |    |   |    |    |    |    |
|----|----|----|----|----|----|----|---|----|---|----|----|----|----|----|----|---|----|--|----|----|----|---|----|----|----|----|
| 0  |    |    | 5  | 18 |    |    |   |    | 3 | 10 |    |    |    |    | 5  |   |    |  |    |    |    | 5 | 7  |    |    |    |
|    |    | 6  |    | 7  |    | 16 |   |    |   | 6  |    |    | 0  | 7  |    |   |    |  |    |    | 10 |   | 19 |    |    | 19 |
|    | 18 |    |    |    |    |    | 0 | 10 |   |    |    | 16 |    | 9  |    |   |    |  |    |    | 4  |   |    |    | 17 |    |
| 5  |    |    |    |    | 18 |    |   |    | 3 | 10 |    |    | 5  |    |    |   |    |  |    |    |    |   |    |    | 7  |    |
|    | 0  |    |    |    |    | 16 | 6 |    |   | 0  |    |    |    | 7  |    |   |    |  |    |    |    |   |    | 19 |    |    |
|    |    | 6  |    | 7  |    |    |   | 2  |   |    |    |    | 9  | 20 |    |   |    |  |    |    |    |   |    | 10 |    |    |
|    |    | 18 |    | 0  |    |    |   | 10 |   |    |    | 16 |    |    | 9  |   |    |  |    |    |    |   |    |    | 17 |    |
|    | 5  |    |    |    |    | 18 | 3 |    |   |    |    | 10 |    | 5  |    | 4 |    |  |    |    | 5  |   |    |    | 7  |    |
|    |    | 0  |    | 16 |    |    |   | 6  |   |    | 0  |    |    |    | 7  | 4 |    |  |    |    |    |   | 10 | 19 |    |    |
| 6  |    |    |    |    | 7  |    |   |    | 2 | 9  |    |    |    | 20 |    |   | 4  |  | 19 |    |    |   |    |    | 10 |    |
|    |    |    | 18 | 0  |    |    |   | 10 |   |    |    | 16 | 9  |    |    |   | 12 |  | 4  | 17 |    |   |    |    |    |    |
|    |    | 5  | 18 |    |    |    |   | 3  |   |    |    |    | 10 | 5  |    |   |    |  | 5  |    |    |   |    |    |    |    |
|    |    |    | 0  |    | 16 |    |   |    | 6 |    |    | 0  |    | 7  |    |   |    |  | 10 |    |    |   |    |    |    |    |
|    | 6  |    |    |    |    |    | 7 | 2  |   |    |    | 9  |    |    | 20 |   |    |  | 19 |    |    |   |    |    |    |    |
| 18 |    |    |    |    | 0  |    |   |    |   | 10 | 16 |    |    | 9  |    |   |    |  |    |    |    | 4 |    |    |    |    |

(a)

|    |    |    |    |    |    |    |    |    |    |   |    |    |    |    |    |    |    |   |    |    |   |   |    |    |    |   |   |    |    |    |
|----|----|----|----|----|----|----|----|----|----|---|----|----|----|----|----|----|----|---|----|----|---|---|----|----|----|---|---|----|----|----|
| 0  |    |    | 5  |    | 18 | 16 |    |    |    | 3 | 6  | 10 |    |    | 0  | 7  |    | 5 |    |    | 4 | 4 |    | 10 |    | 5 |   |    |    |    |
|    |    | 6  |    | 7  |    |    |    |    | 2  |   |    |    |    | 9  |    | 20 |    |   | 4  |    |   |   |    |    | 19 |   |   |    |    |    |
|    | 18 |    |    |    |    |    | 0  | 10 |    |   |    | 16 |    |    |    | 9  |    |   | 12 |    |   | 4 |    |    |    |   |   |    | 17 |    |
| 5  | 0  |    |    |    |    | 18 | 16 | 6  |    |   | 3  | 0  | 10 |    | 5  |    | 7  |   | 4  |    |   |   |    |    |    |   |   |    |    |    |
|    |    |    | 6  |    | 7  |    |    |    |    | 2 |    |    |    |    | 9  |    | 20 |   |    | 4  |   |   |    |    |    |   |   |    |    |    |
|    |    | 18 |    | 0  |    |    |    |    | 10 |   |    |    | 16 |    |    |    | 9  |   |    | 12 |   |   |    |    |    |   |   |    |    |    |
|    | 5  | 0  |    | 16 |    |    | 18 | 3  | 6  |   |    |    | 0  | 10 |    | 5  |    | 7 | 4  | 4  |   |   |    | 5  |    |   |   |    |    |    |
| 6  |    |    |    |    | 7  |    |    |    |    | 2 | 9  |    |    |    |    | 20 |    |   |    | 4  |   |   | 19 |    |    |   |   |    |    |    |
|    |    |    | 18 |    | 0  |    |    |    | 10 |   |    |    | 16 | 9  |    |    |    |   |    | 12 |   |   |    |    |    |   |   |    |    |    |
|    |    | 5  | 0  | 18 | 16 |    |    |    | 3  | 6 |    |    |    | 0  | 10 | 7  | 5  |   |    | 4  | 4 |   | 10 |    | 5  |   | 7 |    |    |    |
|    | 6  |    |    |    |    |    | 7  | 2  |    |   |    |    | 9  |    |    |    | 20 |   |    |    | 4 |   | 19 |    |    |   |   |    |    | 10 |
| 18 |    |    |    |    |    | 0  |    |    |    |   | 10 | 16 |    |    |    | 9  |    |   | 12 |    |   |   |    |    |    | 4 |   | 17 |    |    |

(b)

|    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |   |    |    |   |    |   |    |    |    |    |    |    |    |    |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|---|----|----|---|----|---|----|----|----|----|----|----|----|----|
| 0  |    |    | 5  |    | 18 | 16 |    |    |    | 3  | 6  | 10 |    |    | 0  | 7  |   | 5  |    |   | 4  | 4 |    | 10 |    | 5  |    |    |    |    |
|    | 18 | 6  |    | 7  |    |    | 0  | 10 | 2  |    |    |    | 16 | 9  |    | 20 |   | 9  |    | 4 | 12 |   | 4  | 19 |    |    |    |    |    |    |
| 5  | 0  |    |    |    | 18 | 16 | 6  |    |    | 3  | 0  | 10 |    |    | 5  |    | 7 |    | 4  |   |    | 4 | 5  |    | 10 |    | 19 |    |    |    |
|    |    | 18 | 6  | 0  | 7  |    |    |    | 10 | 2  |    |    | 16 | 9  |    | 20 |   | 9  |    | 4 | 12 |   |    | 4  |    | 19 |    | 10 |    |    |
|    | 5  | 0  |    | 16 |    |    | 18 | 3  | 6  |    |    |    | 0  | 10 |    | 5  |   | 7  | 4  | 4 |    |   |    | 5  |    |    |    |    |    |    |
| 6  |    |    | 18 |    | 0  | 7  |    |    |    | 10 | 2  | 9  |    |    | 16 | 9  |   | 20 |    |   |    | 4 | 12 | 19 |    |    |    |    |    |    |
|    |    | 5  | 0  | 18 | 16 |    |    |    | 3  | 6  |    |    |    | 0  | 10 | 7  | 5 |    |    | 4 | 4  |   | 10 |    | 5  |    | 7  |    | 19 |    |
| 18 | 6  |    |    |    |    | 0  | 7  | 2  |    |    | 10 | 16 | 9  |    |    | 9  |   | 20 | 12 |   |    | 4 |    | 19 |    | 4  |    | 17 |    | 10 |

(c)

|    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 0  | 18 | 6  | 5  | 7  | 18 | 16 | 0  | 10 | 2  | 3  | 6  | 10 | 16 | 9  | 0  | 20 | 7  | 9  | 5  | 4  | 12 | 4  | 4  | 4  | 10 | 19 | 5  | 10 |    |    |    |
| 5  | 0  | 18 | 6  | 0  | 7  | 18 | 16 | 6  | 10 | 2  | 3  | 0  | 10 | 16 | 9  | 5  | 20 | 7  | 9  | 4  | 4  | 12 | 4  | 5  | 4  | 10 | 19 | 19 | 10 |    |    |
| 6  | 5  | 0  | 18 | 16 | 0  | 7  | 18 | 3  | 6  | 10 | 2  | 9  | 0  | 10 | 16 | 9  | 5  | 20 | 7  | 4  | 4  | 4  | 12 | 19 | 5  | 4  | 10 | 17 | 19 | 10 |    |
| 18 | 6  | 5  | 0  | 18 | 16 | 0  | 7  | 2  | 3  | 6  | 10 | 16 | 9  | 0  | 10 | 7  | 9  | 5  | 20 | 12 | 4  | 4  | 4  | 10 | 19 | 5  | 4  | 7  | 17 | 19 | 10 |

(d)

**Figure 7.** IEEE 802.15.3c parity check matrix with  $R =$  (a)  $1/2$ , (b)  $5/8$ , (c)  $3/4$ , and (d)  $7/8$ .

IEEE 802.11ay is the upgraded standard of IEEE 802.11ad, and its QC-LDPC matrix is made up of the IEEE 802.11ad matrix in-place 2nd lifting, as shown in Figure 8. Taking  $R = 13/16$  as an example, the first submatrix with a value of 29 is expanded into four submatrices after the 2nd lifting operation. The 2nd lifting matrix only has two values: 0 and 1. Zero indicates that the value of the original matrix is expanded in the format of the identity matrix, and 1 indicates that the value of the original matrix is expanded in the format of the identity matrix and shifted by one unit to the right. The IEEE 802.11ay matrix and the 2nd lifting matrix are shown in Figures 9 and 10. Finally, we unified the three standard parameters in Table 2.

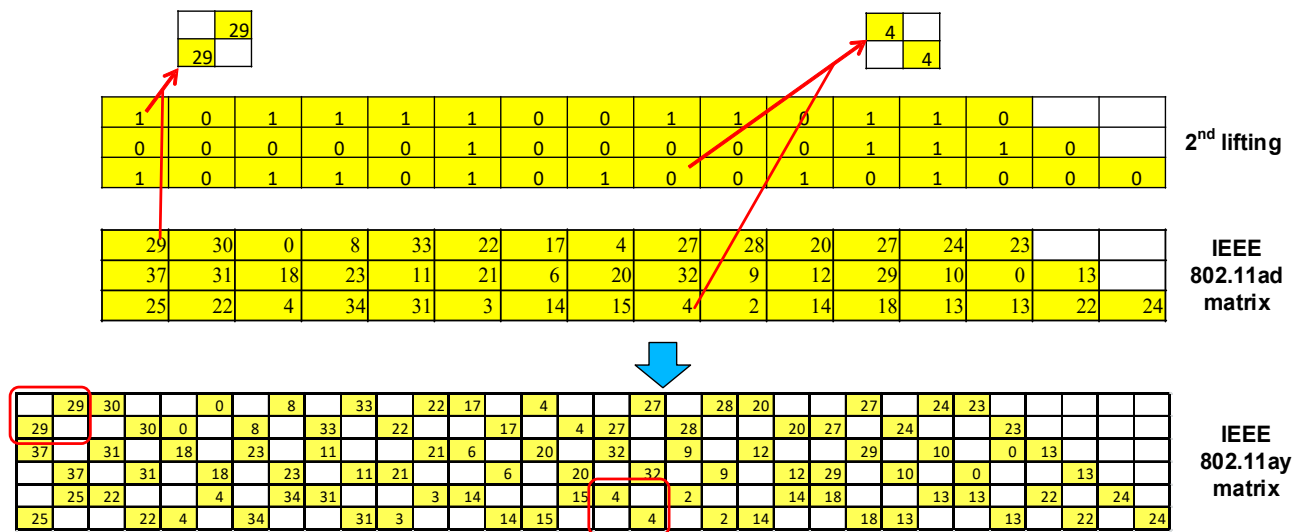


Figure 8. Generation of parity check matrix at IEEE 802.11ay  $R = 13/16$ .

|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 |   | 1 |   | 0 |   | 1 |   | 0 |   |   |   |   |   |   |   |
|   | 0 |   | 1 |   | 0 |   | 1 |   | 1 | 0 |   |   |   |   |   |
| 0 |   | 0 |   | 1 |   |   | 1 | 0 | 0 |   |   |   |   |   |   |
|   | 1 |   | 1 |   | 1 | 0 |   |   |   | 0 | 0 |   |   |   |   |
| 0 |   | 1 |   | 1 |   | 0 |   | 0 |   |   | 1 | 0 |   |   |   |
|   | 0 |   | 0 |   | 1 |   | 0 |   |   | 0 |   |   | 1 | 0 |   |
| 1 |   | 1 |   |   | 1 |   | 0 |   | 1 |   | 1 |   | 0 |   |   |
|   | 0 |   | 1 | 0 |   | 0 |   | 0 |   |   |   | 1 |   | 0 | 0 |

(a)

|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 |   | 1 | 0 |   |   |   |   |   |
| 0 | 1 |   | 1 |   | 1 | 0 | 0 | 1 | 1 | 0 | 0 |   |   |   |   |
| 0 |   | 1 |   | 1 |   | 0 |   | 0 |   |   | 1 | 0 |   |   |   |
|   | 0 |   | 0 |   | 1 |   | 0 |   | 0 | 0 |   |   | 1 | 0 |   |
| 1 |   | 1 |   |   | 1 |   | 0 |   | 1 |   | 1 | 1 | 0 |   |   |
|   | 0 |   | 1 | 0 |   | 0 |   | 0 |   |   |   |   |   | 0 | 0 |

(b)

|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 0 |   |   |   |
| 1 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 0 |   |   |
| 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 |   | 1 | 0 |   |
| 1 | 0 | 1 | 1 | 0 | 1 | 0 | 1 | 0 |   | 1 | 0 | 1 | 0 | 0 | 0 |

(c)

|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 0 |   |   |
| 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 |   |
| 1 | 0 | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 |

(d)

Figure 9. IEEE 802.11ay 2nd lifting matrix with  $R =$  (a)  $1/2$ , (b)  $5/8$ , (c)  $3/4$ , and (d)  $13/16$ .

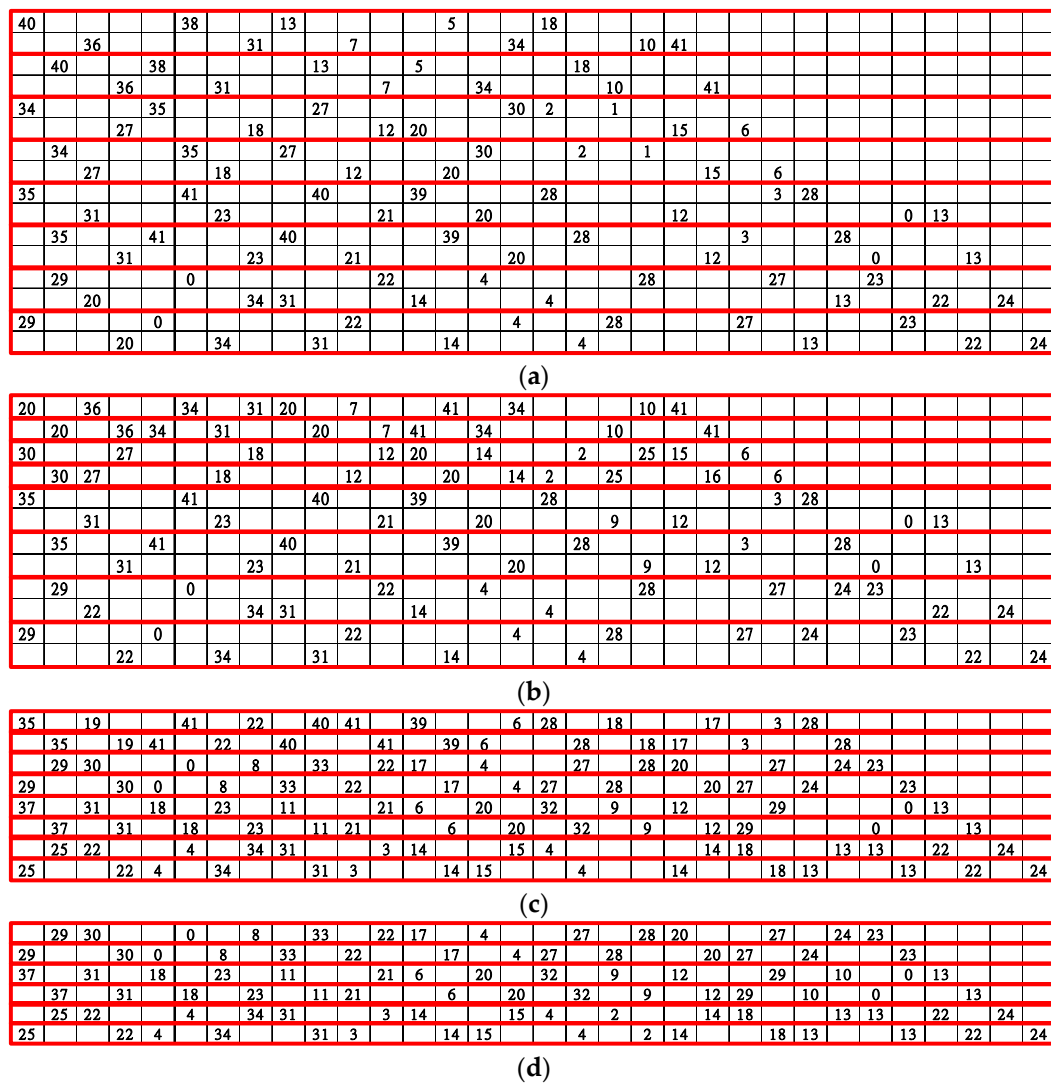


Figure 10. IEEE 802.11ay parity check matrix with  $R =$  (a) 1/2, (b) 5/8, (c) 3/4, and (d) 13/16.

Table 2. Parameters of three standards for 60 GHz wireless local area networks.

| Standards        | IEEE 802.11ay |      |      |       | IEEE 802.11ad |     |     |       | IEEE 802.15.3c |     |     |     |
|------------------|---------------|------|------|-------|---------------|-----|-----|-------|----------------|-----|-----|-----|
| Rate             | 1/2           | 5/8  | 3/4  | 13/16 | 1/2           | 5/8 | 3/4 | 13/16 | 1/2            | 5/8 | 3/4 | 7/8 |
| Row              | 16            | 12   | 8    | 6     | 8             | 6   | 4   | 3     | 16             | 12  | 8   | 4   |
| Column           | 32            | 32   | 32   | 32    | 16            | 16  | 16  | 16    | 32             | 32  | 32  | 32  |
| Expansion factor | 42            | 42   | 42   | 42    | 42            | 42  | 42  | 42    | 21             | 21  | 21  | 21  |
| Code length      | 1344          | 1344 | 1344 | 1344  | 672           | 672 | 672 | 672   | 672            | 672 | 672 | 672 |
| Row weight       | 8             | 10   | 15   | 16    | 8             | 10  | 15  | 16    | 8              | 15  | 16  | 32  |

### 3.2. Proposed Block Layer Decoding for the IEEE 802.11ad, IEEE 802.11ay, and IEEE 802.15.3c Standards

Before deciding on the hardware architecture, we must consider the characteristics of the matrix to determine the hardware parallelism and the amount of computation required. The transmission of decoding information in layer decoding is closely related to the row weights. As an example, the IEEE 802.11ad  $R = 1/2$  matrix is illustrated in Figure 11a. We observe that the row weights are staggered between layers 1 and 2, which implies that the data are not transferred between the two layers for calculation. Therefore, to improve the decoding efficiency, we can decode the two layers without data dependency together, and we refer to this as a block layer, as shown in Figure 11b.





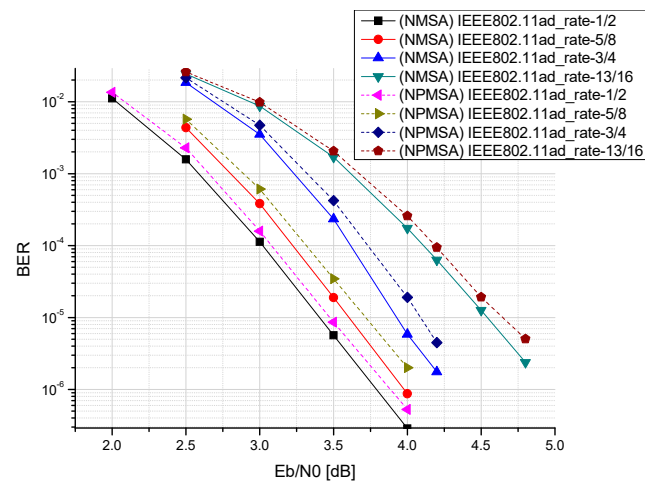


Figure 12. Comparison of decoding performance of different algorithms in IEEE 802.11ad.

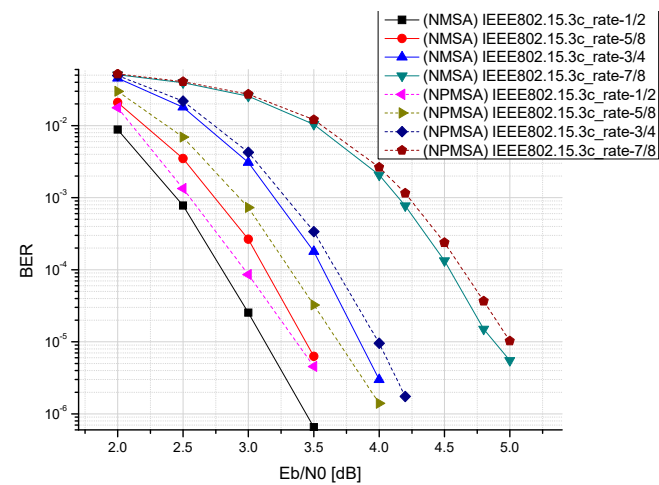


Figure 13. Comparison of decoding performance of different algorithms in IEEE 802.15.3c.

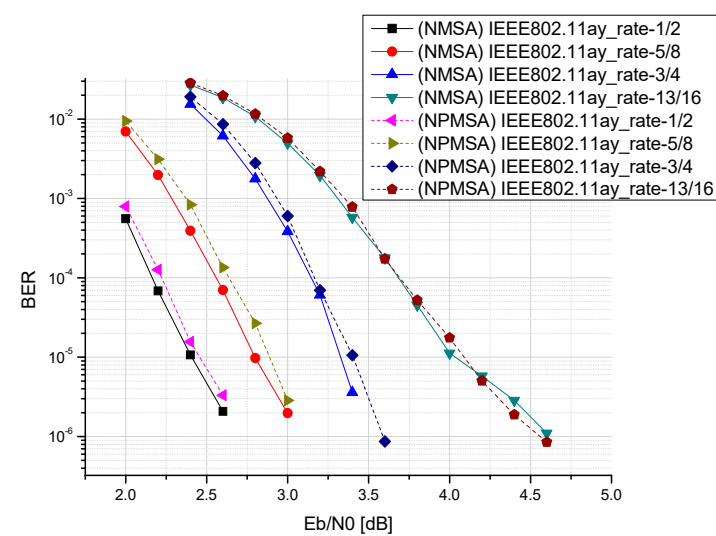


Figure 14. Comparison of decoding performance of different algorithms in IEEE 802.11ay.

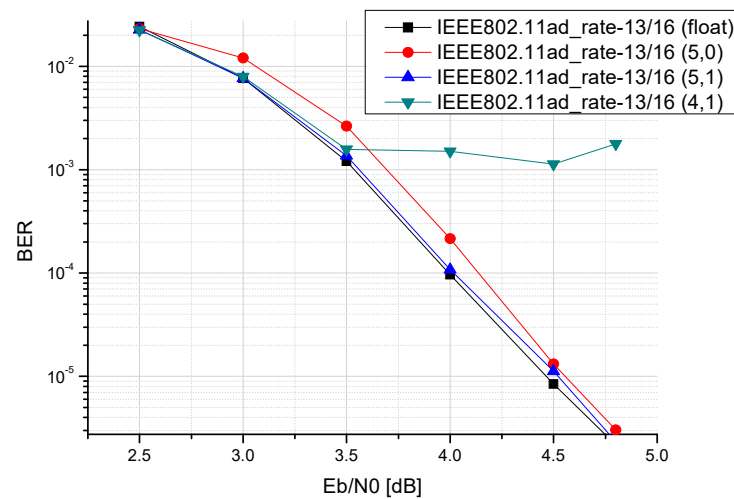


Figure 15. Fixed-point simulation in IEEE 802.11ad  $R = 13/16$ .

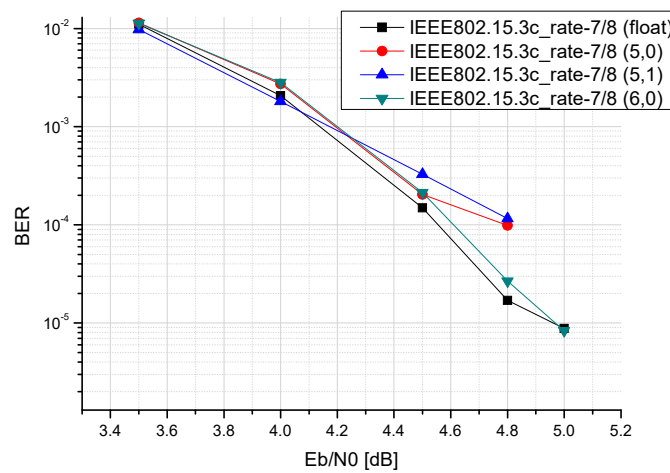


Figure 16. Fixed-point simulation in IEEE 802.15.3c  $R = 7/8$ .

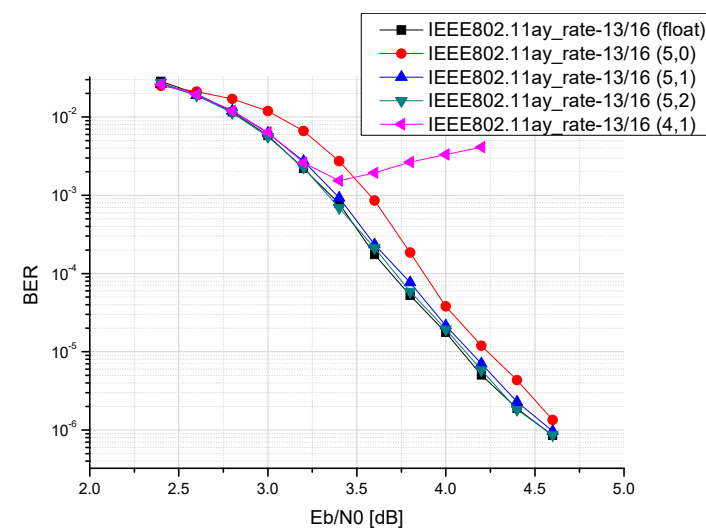


Figure 17. Fixed-point simulation in IEEE 802.11ay  $R = 13/16$ .

#### 4. Architecture Design of Proposed Reconfigurable Multimode LDPC Decoder

This section introduces the architecture of a multi-mode LDPC decoder that supports the IEEE 802.11 ad, IEEE 802.15.3 c, and IEEE 802.11 ay standards. This can be divided into

three parts. The first part consists of the memory for the calculation result, which includes posterior memory and extrinsic memory. The second part comprises an information switch network with different matrices for different standards. The third part is the computing kernel that contains the prior message processing unit (PMU) for calculating the prior messages, the CN processing unit (CNU) for calculating the extrinsic messages, and the VN processing unit (VNU) for calculating the posterior messages. The architecture is shown in Figure 18. The entire decoder hardware uses seven quantised bits for data transmission, and the arithmetic unit is performed at 21 parallelisms. For more details of the entire LDPC decoder, readers can refer to [22,23].

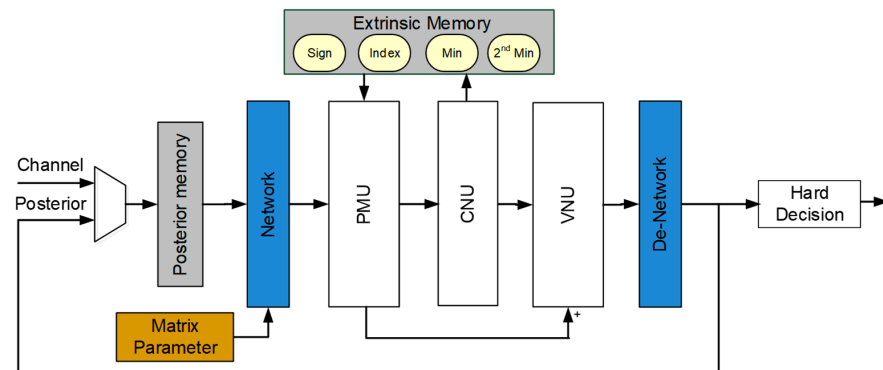


Figure 18. Reconfigurable multi-standard decoder architecture.

#### 4.1. PMU

The PMU receives the prior messages and extrinsic messages of the previous iteration and updates the prior messages. For the first iteration, as there is no information from a previous iteration, the extrinsic messages are initialised to zero, and the information is passed to the CNU for the calculation. In subsequent iterations, the input of the extrinsic messages selects different split blocks according to different matrices. The PMU architecture is illustrated in Figure 19.

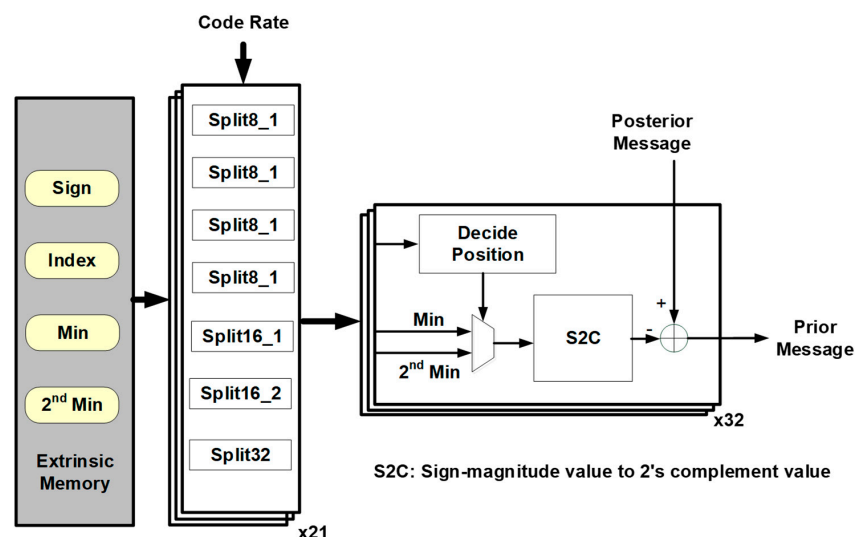


Figure 19. The architecture of the PMU.

#### 4.2. CNU

Figure 20 shows the architecture of the CNU. After receiving the prior message, the sign and magnitude of the message were separated. Because the value of the minimum searcher in CNU must be an absolute value, in terms of signs, exclusive OR logic operations are performed on all signs.

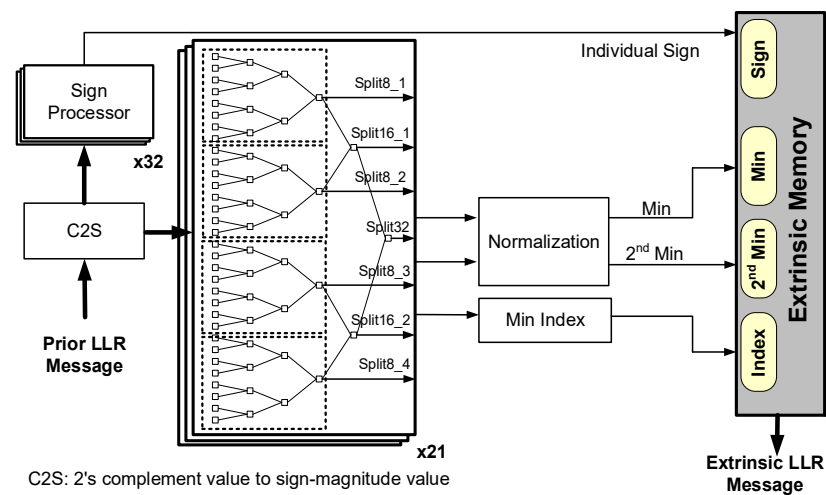


Figure 20. Reconfigurable CNU architecture.

In the sorter, the number of inputs is mainly determined according to the row weight in layered decoding, and a set of sorters can perform a row operation. Thus, the expansion factor  $z$  represents the maximum parallelism of the hardware. However, in a multimode decoder, we can regard the defined block layer as a layer operation, and a block layer operation requires 21 sets of the 32-input sorter to be realised. We refer to the reconfigurable architecture of [24], as shown in Figure 21, and apply it to our multimode decoder. This reconfigurable sorter was originally used in the IEEE 802.15.3c standard; however, we extended it to the IEEE 802.11ad and IEEE 802.11ay standards. Specific arrangements are made such that the block layer under different standards and code rates cannot have redundant idle hardware during the calculation process.

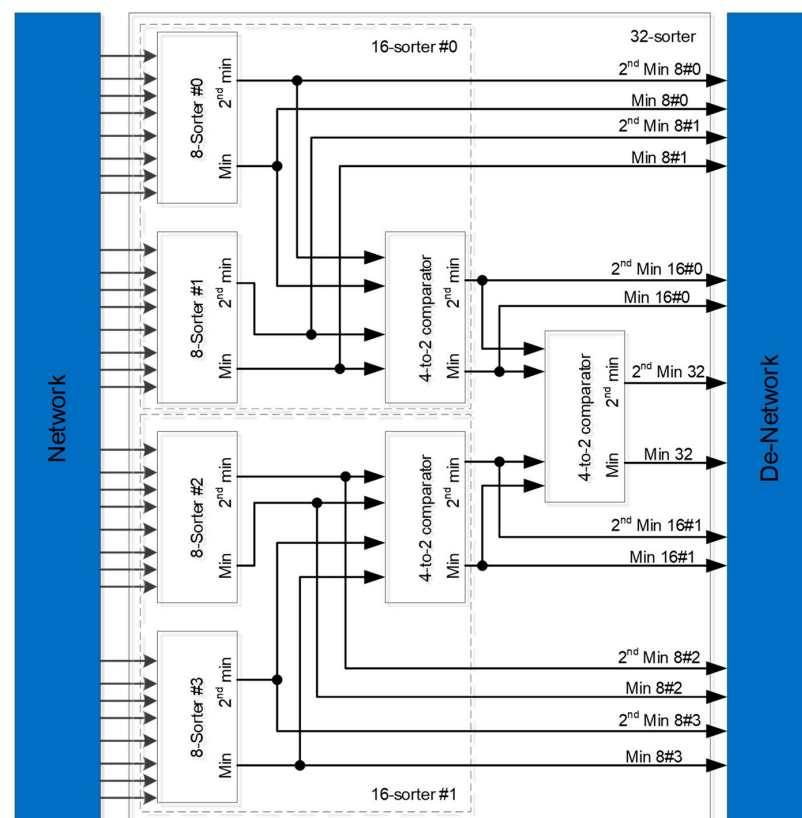
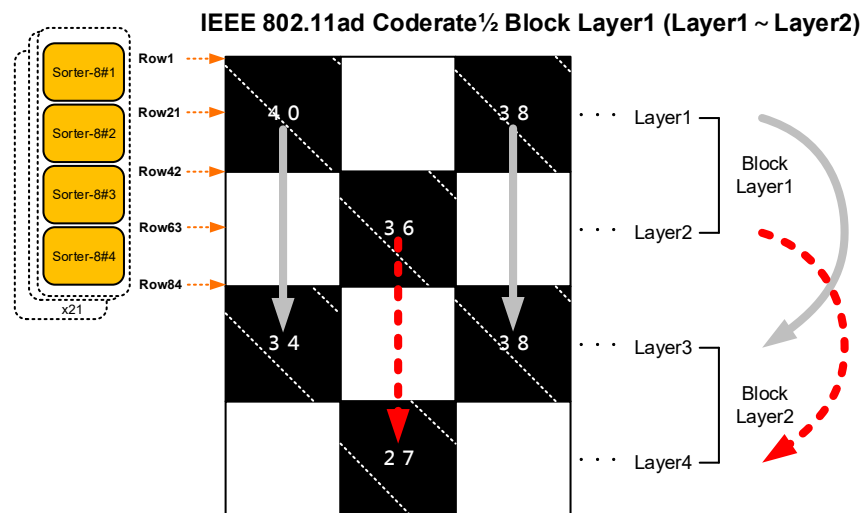


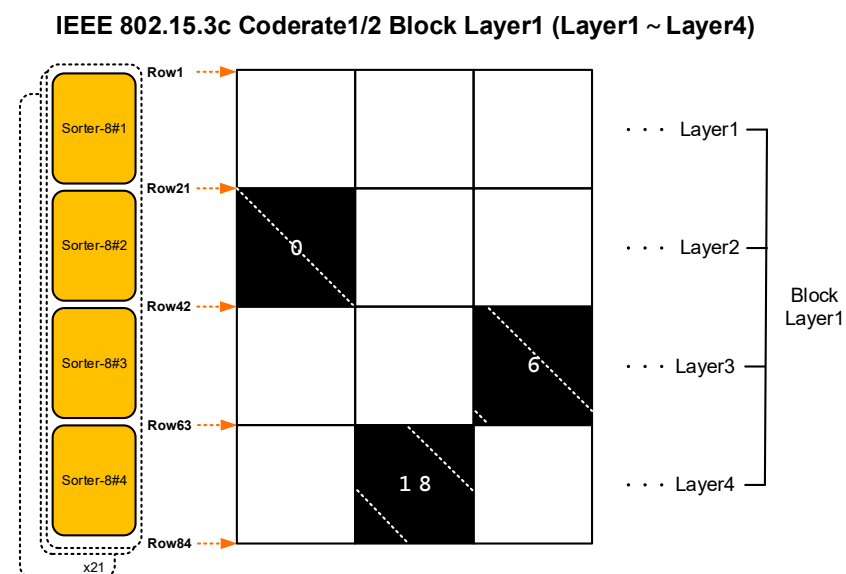
Figure 21. Reconfigurable 8/16/32 input sorter architecture.

Figure 22 shows the IEEE 802.11ad  $R = 1/2$  arrangement. Each sorter-8 represents a minimum value finder (MVF) with eight inputs, and we use 21 sets of parallel hardware for simultaneous operation. We know that the maximum row weight of the IEEE 802.11ad  $R = 1/2$  is 8; therefore, each sorter-8 can calculate one row, and 21 parallelisms can calculate rows 1 to 21. Therefore, sorter-8#1 and sorter-8#2 can only calculate a layer with an expansion factor of 42, and sorter-8#1 to sorter-8#4 can only perform block-layer calculations.



**Figure 22.** Reconfigurable sorter configuration in IEEE 802.11ad  $R = 1/2$ .

The expansion factor of IEEE 802.15.3c is 21, which is half that of IEEE 802.11ad, but the number of layers contained in one block layer is twice that of IEEE 802.11ad; therefore, the same hardware can be used for calculation. Taking the IEEE 802.15.3c  $R = 1/2$  as an example, as shown in Figure 23,  $R = 1/2$  uses four sets of sorter-8 for calculation. In the case of a parallelism of 21, sorter-8#1 can handle operations from rows 1 to 21 in one layer, whereas sorter-8#1 to sorter-8#4 can only operate on one block layer.



**Figure 23.** Reconfigurable sorter configuration in IEEE 802.15.3c  $R = 1/2$ .

Because IEEE 802.11ay is an extension of IEEE 802.11ad, the arrangement of the IEEE 802.11ay sorter is the same as that of IEEE 802.11ad. The difference is that each block layer is doubled, so the number of calculations required is doubled. Regardless of the standard,



the reconfigurable 32-input sorter can support block-level operation. There are a total of 4 schemes that will be used, as shown in Figure 24.

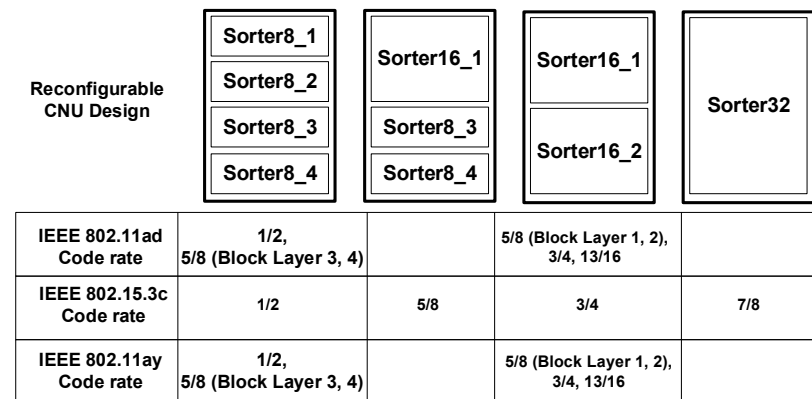


Figure 24. The configuration of the sorter under different code rates for each standard.

#### 4.3. VNU

The VNU is similar to the PMU. It contains 32 sets of parallel-computing processors. The difference is that the prior messages are obtained from the PMU and the extrinsic messages are obtained from the CNU for calculation. The final calculated posterior message is stored in the posterior memory for the next iteration operation, as shown in Figure 25.

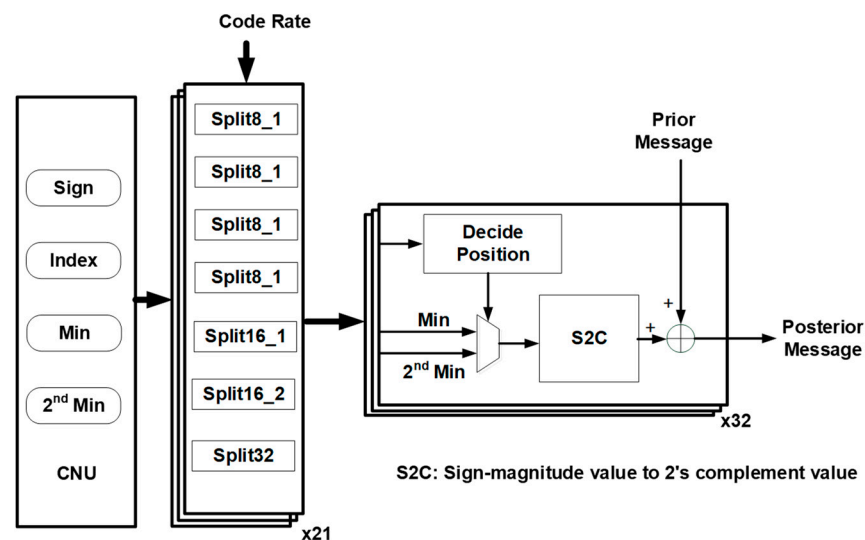


Figure 25. The architecture of the VNU.

#### 4.4. Switching Network

The design of the switching network in the reconfigurable multimode decoder architecture is also a topic that is often discussed. A multimode switching network requires different input and output sizes in different standards between the memory and processing units, and the control signal in the reconfigurable design will also be very complicated. Therefore, the designed switching network architecture is based on the architecture in [25]. Compared to the traditional Benes network [26], this architecture has the following advantages:

1. The number of inputs may not be a power of 2.
2. The number of bits required for the look-up table is very small.
3. The hardware usage rate of the proposed multi-mode architecture is extremely high.

This switching network is based on the expansion of  $2 \times 2$ ,  $3 \times 3$ , or  $5 \times 5$  switching networks, so the maximum input size  $P_M$  may not be a power of 2, where  $P_M = \beta 2^i$ ,

$\beta \in \{2, 3, 5\}$ , and  $i \in \{1, 2, 3, \dots\}$ . When the number of inputs required is 42, a traditional Benes network will need to use a network with  $2^6 = 64$  inputs, and the set of hardware will use  $S_M/2 \times (2 \log_2 S_M - 1) = 352$ ,  $2 \times 2$  switches, where  $S_M$  is the Benes network input size. However, using the network architecture proposed in [25] requires the use of a  $3 \times 2^4 = 48$  input network; the set of hardware will use  $3 \times 2^i + 3 \times 2^i \log_2 2^i = 240$ ,  $2 \times 2$  switches, and the number of  $2 \times 2$  switches used will be reduced by 112.

In this study, we employ the similar notations and illustration revealed in [25] to demonstrate the reconfigurable switching network. Figure 26a illustrates the example of six-input switching network architecture for  $(p, c, P_M) = (5, 3, 6)$  used in the reconfigurable decoder architecture, where  $p$  is the size of the submatrix and  $c$  is the shifting value. There are three stages, F1, FL, and L1. F1 stage has three switches with the control signal  $f_{1,j}$ . L1 stage also has three switches with the control signal  $l_{1,j}$ . FL stage has six switches with the control signal  $fl_j$ . Figure 26b shows the values of control signals for this six-input switching network. When the status of the switch is “CROSS”, the value of control signal is “1”. When the status of the switch is “BAR”, the value of control signal is “0”. It is noteworthy that the large switching network architecture can be split into two small switching network architectures. As shown in Figure 26,  $(5, 3, 6)$ , switching network is split into  $(2, 1, 3)$  and  $(3, 2, 3)$  switching network architectures.

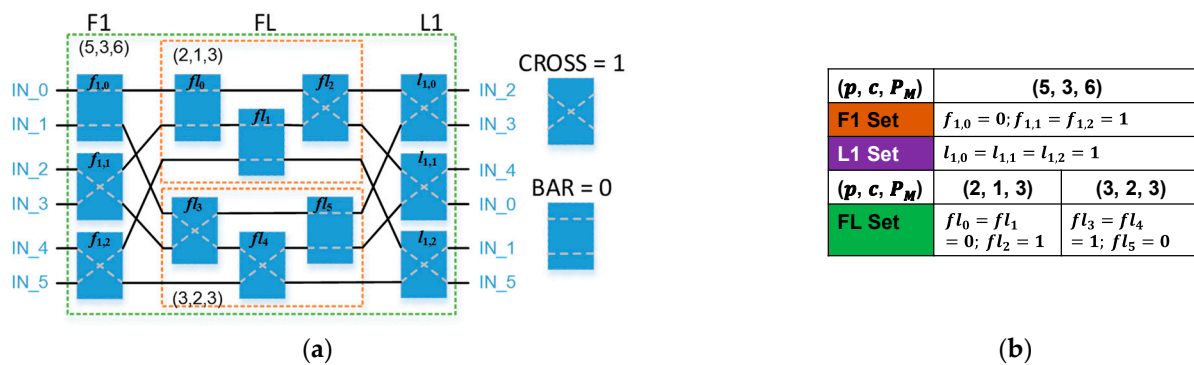


Figure 26. (a) Six-input switching network architecture and (b) its control signal table for  $(p, c, P_M) = (5, 3, 6)$ .

As  $P_M$  increases, the switching network architecture becomes complicated. Practically, the control signal in the switching network can be realised using a lookup table. The method for determining the control signal is shown in Figure 27. Block (A) is used to determine the control signal of switches in the F stages, and Block (B) is used to determine the control signal of switches in the L stages. Finally, the control signal of switches in the FL layers is determined in Block (C). When  $P_M$  is large, the control signal of each switch can be feasibly determined using the above process illustrated in Figure 27. For the more details of the control signal generation of the switch, readers can refer to [25]. Taking the  $24 \times 24$  shifting network as an example, the control signal generated by a shifting value of 14 is shown in Figure 28. In F1–F3 and L1–L3 stages, each stage has twelve switches. FL stage has 24 switches. The control signal of each switch (i.e.,  $f_{i,j}$ ,  $l_{i,j}$ , and  $fl_j$ ) was determined by the method shown Figure 27.

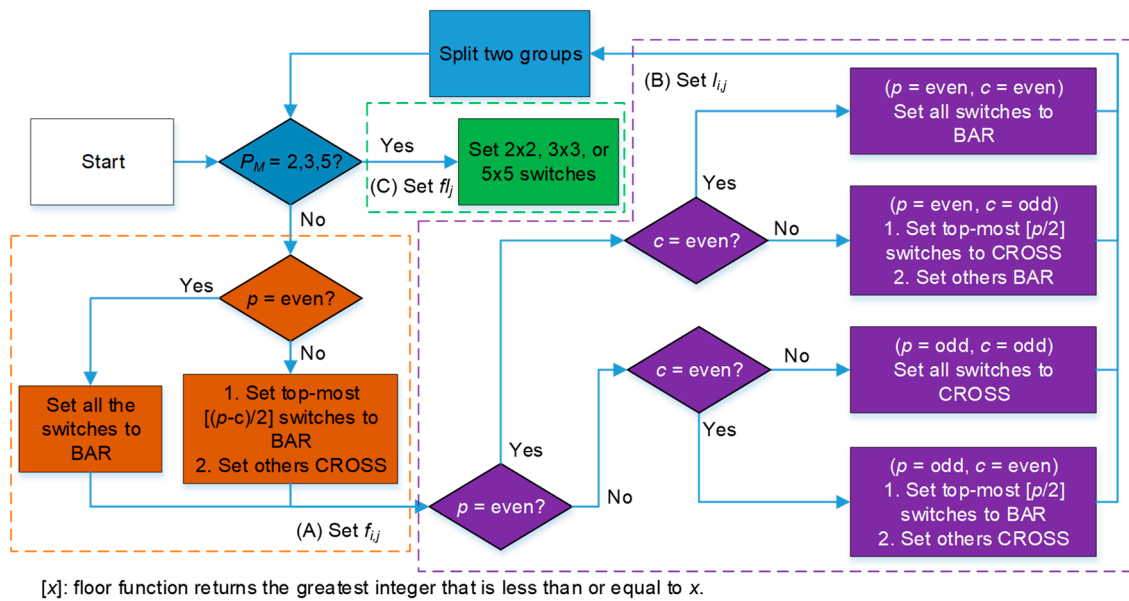


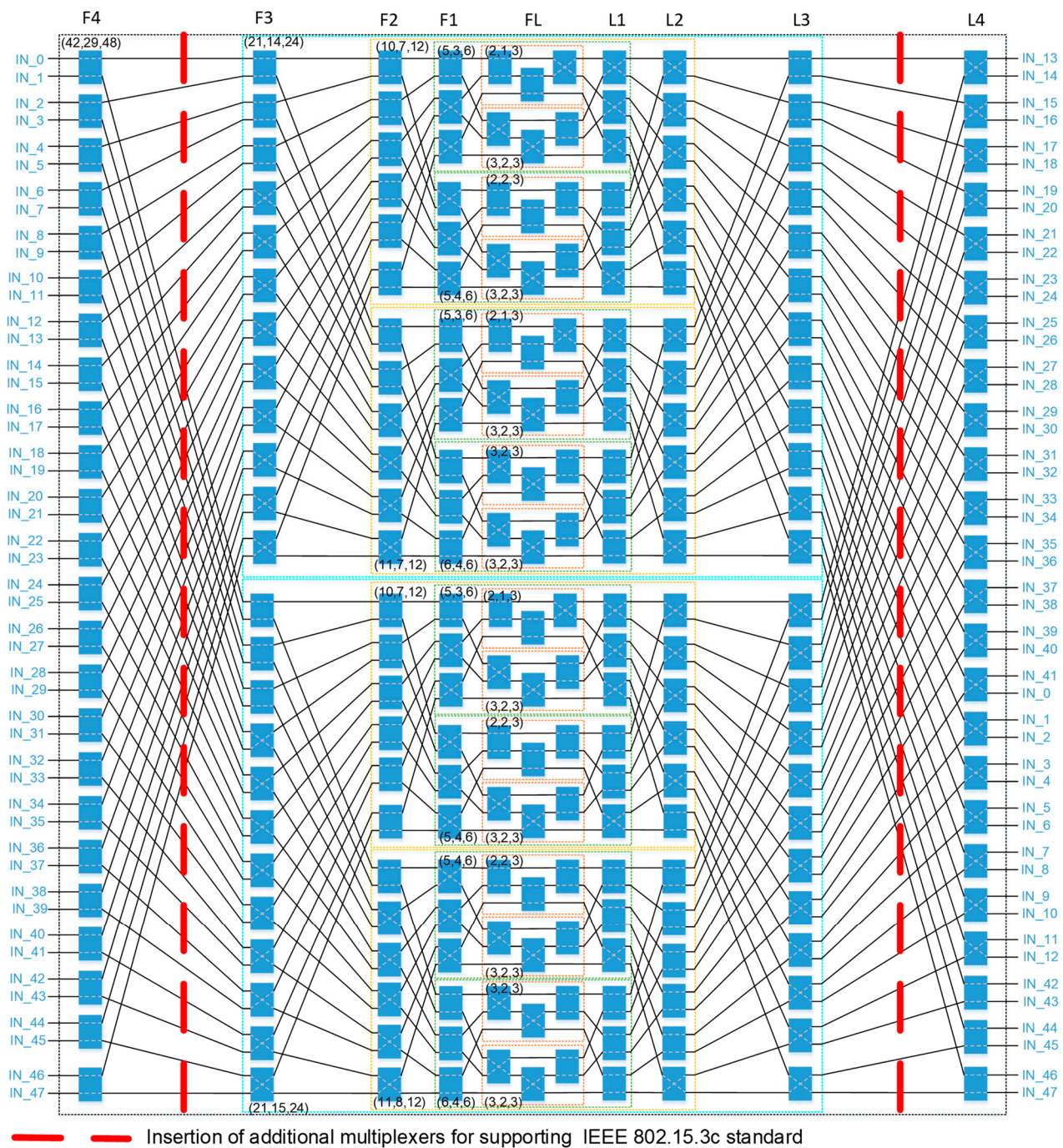
Figure 27. Flow chart of control signal generation of the switch in the switching network in the reconfigurable multimode LDPC decoder architecture.

| $(p, c, P_M)$ | (21, 14, 24)   |                             |                                      |                                   |  |                                      |                                      |                                    |
|---------------|--|-----------------------------|--------------------------------------|-----------------------------------|--|--------------------------------------|--------------------------------------|------------------------------------|
| <b>F3 Set</b> | $f_{3,0} = f_{3,1} = f_{3,2} = 0; f_{3,3} = f_{3,4} = f_{3,5} = f_{3,6} = f_{3,7} = f_{3,8} = f_{3,9} = f_{3,10} = f_{3,11} = 1$ |                             |                                      |                                   |  |                                      |                                      |                                    |
| <b>L3 Set</b> | $l_{3,0} = l_{3,1} = l_{3,2} = l_{3,3} = l_{3,4} = l_{3,5} = l_{3,6} = l_{3,7} = l_{3,8} = l_{3,9} = 0; l_{3,10} = l_{3,11} = 1$ |                             |                                      |                                   |  |                                      |                                      |                                    |
| $(p, c, P_M)$ | (10, 7, 12)  |                             |                                      |                                   | (11, 7, 12)  |                                      |                                      |                                    |
| <b>F2 Set</b> | $f_{2,0} = f_{2,1} = f_{2,2} = f_{2,3} = f_{2,4} = f_{2,5} = 0$  |                             |                                      |                                   | $f_{2,6} = f_{2,7} = 0; f_{2,8} = f_{2,9} = f_{2,10} = f_{2,11} = 1$ |                                      |                                      |                                    |
| <b>L2 Set</b> | $l_{2,0} = l_{2,1} = l_{2,2} = l_{2,3} = l_{2,4} = 1; l_{2,5} = 0$   |                             |                                      |                                   | $l_{2,6} = l_{2,7} = l_{2,8} = l_{2,9} = l_{2,10} = l_{2,11} = 1$    |                                      |                                      |                                    |
| $(p, c, P_M)$ | (5, 3, 6)  |                             | (5, 4, 6)                            |                                   | (5, 3, 6)  |                                      | (6, 4, 6)                            |                                    |
| <b>F1 Set</b> | $f_{1,0} = 0; f_{1,1} = f_{1,2} = 1$   |                             | $f_{1,3} = f_{1,4} = f_{1,5} = 1$    |                                   | $f_{1,6} = 0; f_{1,7} = f_{1,8} = 1$                                 |                                      | $f_{1,9} = f_{1,10} = f_{1,11} = 0$  |                                    |
| <b>L1 Set</b> | $l_{1,0} = l_{1,1} = l_{1,2} = 1$  |                             | $l_{1,3} = l_{1,4} = 0; l_{1,5} = 1$ |                                   | $l_{1,6} = l_{1,7} = l_{1,8} = 1$                                    |                                      | $l_{1,9} = l_{1,10} = l_{1,11} = 0$  |                                    |
| $(p, c, P_M)$ | (2, 1, 3)  | (3, 2, 3)                   | (2, 2, 3)                            | (3, 2, 3)                         | (2, 1, 3)  | (3, 2, 3)                            | (3, 2, 3)                            | (3, 2, 3)                          |
| <b>FL Set</b> | $fl_0 = fl_1 = 0; fl_2 = 1$  | $fl_3 = fl_4 = 1; fl_5 = 0$ | $fl_6 = fl_7 = fl_8 = 0$             | $fl_9 = fl_{10} = 1; fl_{11} = 0$ | $fl_{12} = fl_{13} = 0; fl_{14} = 1$                                 | $fl_{15} = fl_{16} = 1; fl_{17} = 0$ | $fl_{18} = fl_{19} = 1; fl_{20} = 0$ | $fl_{21} fl_{22} = 1; fl_{23} = 0$ |

Figure 28. Control signal table for  $(p, c, P_M) = (21, 14, 24)$ .

Figure 29 illustrates the top  $48 \times 48$  shifting network used in the reconfigurable LDPC decoder architecture, and the control signals of switches are determined for  $(p, c, P_M) = (42, 29, 48)$ . It is too complicated and trivial to illustrate the control signal table in this study. We believe readers can achieve it based on the examples demonstrated in Figures 26–28. Compared with the traditional Benes network [26], the control signals that we need is simplified and requires only 588 bits, as shown in Table 3. Applying the architecture in [25] to the designed architecture successfully reduced the hardware complexity significantly. Compared with the Benes network, the designed architecture reduces  $1792 \times 2$  switches. Finally, we used 16 sets of parallel  $48 \times 48$  shifting networks that can meet the parallel computing requirements of the IEEE 802.11ad and IEEE 802.11ay standards with an input requirement of 42 and a maximum row weight of 16. In the IEEE 802.15.3c standard, the required number of inputs is 21 and the maximum row weight is 32, which means that 32 sets of parallel hardware are required, and the number of inputs of each set must satisfy the requirement of 21 inputs. However, we observed that a  $48 \times 48$  shifting network transforms into two  $24 \times 24$  shifting networks after being split into two groups for the first time. According to this, 16 sets of  $48 \times 48$  shifting networks can meet the requirement of 32 sets of  $24 \times 24$  shifting networks for IEEE 802.15.3c standard. This only requires the additional multiplexers between the F4 to F3 and L3 to L4 transmission networks, as illustrated in Figure 29. Only adding multiplexers can complete the switching between different modes, so that hardware sharing is high.





**Figure 29.** Forty-eight-input switching network architecture for  $(p, c, P_M) = (42, 29, 48)$ .

**Table 3.** Comparison of distinct switching networks.

| Network              | Benes Network [26] | This Work      |
|----------------------|--------------------|----------------|
| Network size         | $64 \times 64$     | $48 \times 48$ |
| Look-up table (bits) | 709,984 (100%)     | 588 (0.08%)    |

#### 4.5. Memory Organization

Memory is divided into two parts: posterior memory and extrinsic memory, both of which are used to save the posterior messages and extrinsic messages required for the next iteration after the current iteration update. Considering the auto place and route (APR) congestion problem, the memory design adopts a register-based design that can be placed

more flexibly. The posterior memory part adopts a single-port design, and the extrinsic memory adopts a two-port design. The posterior memory must save the post-probability value of the code length. In the IEEE 802.11ad and IEEE 802.15.3c standards, the code length is 672, but the code length of the IEEE 802.11ay standard is 1344; therefore, we must follow the maximum demand IEEE 802.11ay standard 1344 code length multiplied by our quantisation bits 7. Thus, the required memory size is 9408 bits ( $=1344 \times 7$ ).

Four pieces of information need to be saved in the extrinsic memory: address information of the minimum value, sign, minimum value, and second minimum value. However, the amount of information that must be stored in different standards and code rates is also different. Different data-storage arrangements must be made according to the calculation results of each block layer. The storage requirements of each code rate under different standards are listed in Figures 30 and 31. Figure 30 shows the extrinsic memory capacity required by IEEE 802.11ad, and it is worth noting that the IEEE 802.11ay matrix is extended by the IEEE 802.11ad matrix, so the required extrinsic memory capacity is the same.

|           | IEEE 802.11ad/ay rate1/2        |       |     |                     | IEEE 802.11ad/ay rate5/8  |       |     |                     | IEEE 802.11ad/ay rate3/4         |       |     |                     | IEEE 802.11ad/ay rate13/16       |       |     |                     |
|-----------|---------------------------------|-------|-----|---------------------|---|-------|-----|---------------------|----------------------------------|-------|-----|---------------------|----------------------------------|-------|-----|---------------------|
| Split     | Sign                            | Index | Min | 2 <sup>nd</sup> Min | Sign  | Index | Min | 2 <sup>nd</sup> Min | Sign                             | Index | Min | 2 <sup>nd</sup> Min | Sign                             | Index | Min | 2 <sup>nd</sup> Min |
| 8_1       | 8                               | 3     | 5   | 5                   | 8   | 3     | 5   | 5                   |                                  |       |     |                     |                                  |       |     |                     |
| 8_2       | 8                               | 3     | 5   | 5                   | 8   | 3     | 5   | 5                   |                                  |       |     |                     |                                  |       |     |                     |
| 8_3       | 8                               | 3     | 5   | 5                   | 8   | 3     | 5   | 5                   |                                  |       |     |                     |                                  |       |     |                     |
| 8_4       | 8                               | 3     | 5   | 5                   | 8   | 3     | 5   | 5                   |                                  |       |     |                     |                                  |       |     |                     |
| 16_1      |                                 |       |     |                     | 16  | 4     | 5   | 5                   | 16                               | 4     | 5   | 5                   | 16                               | 4     | 5   | 5                   |
| 16_2      |                                 |       |     |                     | 16  | 4     | 5   | 5                   | 16                               | 4     | 5   | 5                   | 16                               | 4     | 5   | 5                   |
| 32        |                                 |       |     |                     |   |       |     |                     |                                  |       |     |                     |                                  |       |     |                     |
| Total bit | $(8 + 3 + 5 + 5) \times 4 = 84$ |       |     |                     | Layers 1~2:<br>$(16 + 4 + 5 + 5) \times 2 = 60$<br>Layers 3~4:<br>$(8 + 3 + 5 + 5) \times 4 = 84$ |       |     |                     | $(16 + 4 + 5 + 5) \times 2 = 60$ |       |     |                     | $(16 + 4 + 5 + 5) \times 2 = 60$ |       |     |                     |

**Figure 30.** The number of bits required by the extrinsic memory in the IEEE 802.11ad/ay standard with different code rates.

|           | IEEE 802.15.3c rate1/2          |       |     |                     | IEEE 802.15.3c rate5/8                             |       |     |                     | IEEE 802.15.3c rate3/4           |       |     |                     | IEEE 802.15.3c rate7/8  |       |     |                     |
|-----------|---------------------------------|-------|-----|---------------------|--|-------|-----|---------------------|----------------------------------|-------|-----|---------------------|-------------------------|-------|-----|---------------------|
| Split     | Sign                            | Index | Min | 2 <sup>nd</sup> Min | Sign   | Index | Min | 2 <sup>nd</sup> Min | Sign                             | Index | Min | 2 <sup>nd</sup> Min | Sign                    | Index | Min | 2 <sup>nd</sup> Min |
| 8_1       | 8                               | 3     | 5   | 5                   |  |       |     |                     |                                  |       |     |                     |                         |       |     |                     |
| 8_2       | 8                               | 3     | 5   | 5                   |  |       |     |                     |                                  |       |     |                     |                         |       |     |                     |
| 8_3       | 8                               | 3     | 5   | 5                   | 8  | 3     | 5   | 5                   |                                  |       |     |                     |                         |       |     |                     |
| 8_4       | 8                               | 3     | 5   | 5                   | 8  | 3     | 5   | 5                   |                                  |       |     |                     |                         |       |     |                     |
| 16_1      |                                 |       |     |                     | 16   | 4     | 5   | 5                   | 16                               | 4     | 5   | 5                   |                         |       |     |                     |
| 16_2      |                                 |       |     |                     |  |       |     |                     | 16                               | 4     | 5   | 5                   |                         |       |     |                     |
| 32        |                                 |       |     |                     |  |       |     |                     |                                  |       |     |                     | 32                      | 5     | 5   | 5                   |
| Total bit | $(8 + 3 + 5 + 5) \times 4 = 84$ |       |     |                     | $(8 + 3 + 5 + 5) \times 2 + (16 + 4 + 5 + 5) = 72$ |       |     |                     | $(16 + 4 + 5 + 5) \times 2 = 60$ |       |     |                     | $(32 + 5 + 5 + 5) = 47$ |       |     |                     |

**Figure 31.** The number of bits required by the extrinsic memory in the IEEE 802.15.3c standard with different code rates.

Finally, the extrinsic memory structure, as shown in Figure 32, was divided into two parts: Memory\_1 and Memory\_2. Memory\_2 is used only in IEEE 802.11ay. Each memory is divided into 21 memory banks to store 21 pieces of parallel hardware information, and each memory bank has four memory cells to store four block-level information. The memory cell size is 84 bits, and the data will be stored in a total of four cases of different sizes. The total extrinsic memory size is 14,112 bits ( $=2 \times 21 \times 4 \times 84$ ).



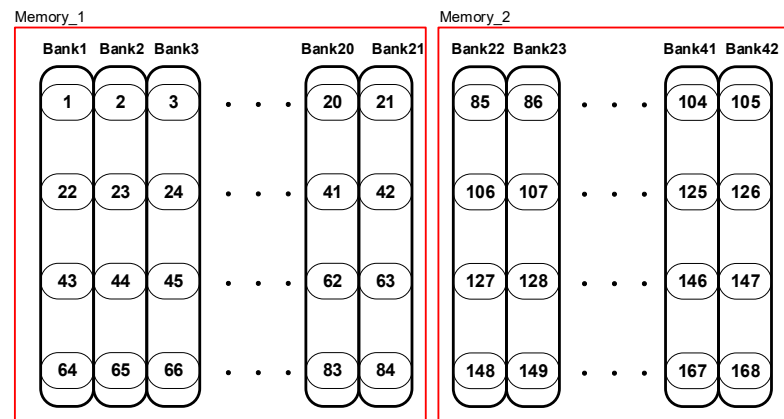


Figure 32. Extrinsic memory in reconfigurable multi-standard decoder.

### 5. VLSI Implementation of Proposed Reconfigurable Multimode LDPC Decoder

Figure 33 reveals the block-level chip implementation results of the proposed reconfigurable LDPC decoder. The chip was implemented using a TSMC 40 nm CMOS process with an operating voltage of 0.9 V; operating frequency of 117 MHz; and core area  $1.312 \text{ mm} \times 1.312 \text{ mm}$ , that is,  $1.72 \text{ mm}^2$ . The throughput is described as follows:

$$\text{Throughput } (Tp) = \frac{\text{Codeword} \times \text{Frequency}}{\left[ \frac{z}{\text{parallelism} \times S_p} \times \text{Block layer} \right] \times \text{Iteration}}, \quad (6)$$

where  $z$  is the expending factor and  $S_p$  is the standard parameter ( $=1$  for IEEE 802.15.3c;  $=2$  for IEEE 802.11ad/ay).

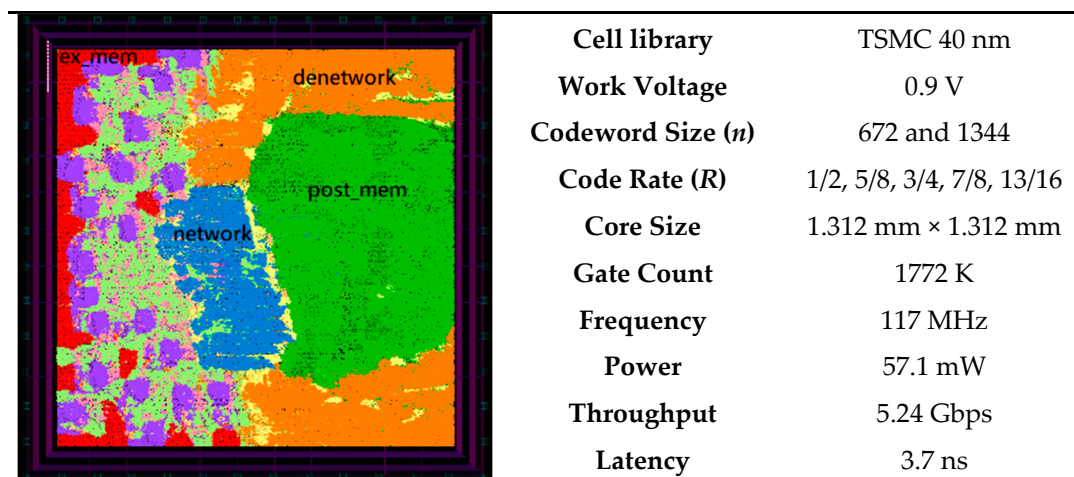


Figure 33. Chip summary and layout of the proposed reconfigurable multi-standard LDPC decoder.

Currently, there are no other studies discussing the integration of LDPC decoders for 60 GHz wireless transmission, and there are no related studies on the implementation of the IEEE 802.11ay standard on the chip. Therefore, the results can only be compared with the single standard studies of IEEE 802.11ad or IEEE 802.15.3c. For a fair systematic comparison with other studies, normalised metrics [11,27] are utilised and listed as follows:

$$\text{Normalized Area Efficiency (NAE)} = \frac{Tp \times \text{Normalized Area factor} \left( = \left( \frac{S}{40} \right)^2 \right)}{\text{Area} \times \text{Frequency}}, \quad (7)$$

$$\text{Normalized Energy Efficiency (NEE)} = \frac{\text{Power} \times \text{Normalized energy factor} \left( = (40/S) \times (0.9/U)^2 \right)}{T_p \times \text{Iteration}}, \quad (8)$$

where  $S$  is the scaled technology and  $U$  is the scaled supply voltage. Table 4 shows a comparison with other IEEE 802.11ad studies. In terms of the NEE, the hardware architecture we proposed is superior to that of other studies. In the NAE, the performance is particularly outstanding because [28] only operates at one rate. Table 5 shows a comparison with other IEEE 802.15.3c-related studies. In comparison with IEEE 802.15.3c, the proposed hardware architectures are slightly inferior. This is because a large part of the proposed hardware architecture complies with the hardware added by IEEE 802.11ad and IEEE 802.11ay; therefore, the values cannot be compared with a single standard.

**Table 4.** Chip comparisons among different LDPC decoders for IEEE 802.11ad standard.

| Decoders                    | This Study                         | [28]          | [29]                    | [30]                    | [31]                    |
|-----------------------------|------------------------------------|---------------|-------------------------|-------------------------|-------------------------|
| Implementation              | Post-layout                        | Measurement   | Measurement             | Measurement             | Measurement             |
| Technology                  | 40 nm                              | 40 nm         | 40 nm                   | 28 nm                   | 28 nm                   |
| Voltage                     | 0.9 V                              | 0.9 V         | 1.1 V                   | 0.9 V                   | 0.9 V                   |
| Standard                    | IEEE 802.11ad/ay<br>IEEE 802.15.3c | IEEE 802.11ad | IEEE 802.11ad           | IEEE 802.11ad           | IEEE 802.11ad           |
| Code rate ( $R$ )           | 1/2, 5/8, 3/4<br>7/8, 13/16        | 13/16         | 1/2, 5/8, 3/4,<br>13/16 | 1/2, 5/8, 3/4,<br>13/16 | 1/2, 5/8, 3/4,<br>13/16 |
| Algorithm                   | NPMSA                              | MSA           | MSA                     | MSA                     | MSA                     |
| Quantization (bits)         | 7                                  | 5             | 5                       | 5                       | 4                       |
| Iteration                   | 5                                  | 5             | 7                       | 10                      | 4                       |
| Parallelism                 | 21                                 | 42            | 16                      | 168                     | 42                      |
| Frequency (MHz)             | 117                                | 500           | 220                     | 202                     | 470                     |
| $T_p$ (Gbps)                | 5.24                               | 5.6           | 6.16                    | 6.78                    | 18.4                    |
| Area (mm <sup>2</sup> )     | 1.72                               | 0.16          | 0.8                     | 1.99                    | 0.78                    |
| Power (mW)                  | 57.1                               | 99            | 203                     | 104                     | 166                     |
| NAE (bits/mm <sup>2</sup> ) | 0.026                              | 0.07          | 0.035                   | 0.0082                  | 0.024                   |
| NEE (pJ/bit)                | 2.179                              | 3.53          | 3.15                    | 2.19                    | 3.22                    |

**Table 5.** Chip comparisons among different LDPC decoders for IEEE 802.15.3c standard.

| Decoders                    | This Study                         | [32]                   | [33]                   | [34]                   | [24]                   |
|-----------------------------|------------------------------------|------------------------|------------------------|------------------------|------------------------|
| Implementation              | Post-layout                        | Measurement            | Measurement            | Measurement            | Measurement            |
| Technology                  | 40 nm                              | 90 nm                  | 65 nm                  | 90 nm                  | 65 nm                  |
| Voltage                     | 0.9 V                              | 1.2 V                  | 1.2 V                  | 1.05 V                 | 1.0 V                  |
| Standard                    | IEEE 802.11ad/ay<br>IEEE 802.15.3c | IEEE 802.15.3c         | IEEE 802.15.3c         | IEEE 802.15.3c         | IEEE 802.15.3c         |
| Code rate                   | 1/2, 5/8, 3/4<br>7/8, 13/16        | 1/2, 5/8,<br>3/4, 7/8, | 1/2, 5/8,<br>3/4, 7/8, | 1/2, 5/8,<br>3/4, 7/8, | 1/2, 5/8,<br>3/4, 7/8, |
| Algorithm                   | NPMSA                              | NMSA                   | NMSA                   | NMSA                   | NMSA                   |
| Quantization (bits)         | 7                                  | 10                     | 6                      | 6                      | 6                      |
| Iteration                   | 5                                  | 10                     | 10                     | 5                      | 5                      |
| Parallelism                 | 21                                 | 21                     | 21                     | 21                     | 21                     |
| Frequency (MHz)             | 117                                | 768                    | 400                    | 157                    | 197                    |
| $T_p$ (Gbps)                | 3.90                               | 7.92                   | 6.72                   | 5.28                   | 5.79                   |
| Area (mm <sup>2</sup> )     | 1.72                               | 2.67                   | 1.3                    | 2.25                   | 1.56                   |
| Power (mW)                  | 57.1                               | 437.2                  | 537.6                  | 182                    | 361                    |
| NAE (bits/mm <sup>2</sup> ) | 0.019                              | 0.019                  | 0.034                  | 0.075                  | 0.049                  |
| NEE (pJ/bits)               | 2.92                               | 1.98                   | 2.76                   | 2.25                   | 6.21                   |

## 6. Conclusions

In this study, a reconfigurable LDPC decoder was proposed to support the application of three standards of 60 GHz wireless transmission: IEEE 802.11ad, IEEE 802.15.3c, and IEEE 802.11ay. To support different standards, we divide the matrix in different standards into block layers for decoding to ensure good hardware sharing and use reconfigurable hardware architecture in the CNU and switch network to save a lot of hardware consumption. Finally, the multi-mode reconfigurable LDPC decoder applied to 60 GHz wireless transmission is realised using the TSMC 40 nm CMOS process, using 21 parallelisms, two pipeline stages, an operating frequency of 117 MHz, and a core area of  $1.312\text{ mm} \times 1.312\text{ mm}$ ; the power consumption is only 57.1 mW. The throughput is up to 5.24 Gbps in the IEEE 802.11ad and IEEE 802.11ay modes, and the throughput is 3.9 Gbps in the IEEE 802.15.3c mode.

**Author Contributions:** Conceptualization, C.-H.L. and T.-S.C.; methodology, C.-H.L., T.-S.C. and H.-H.S.; software, H.-H.S.; validation, C.-H.L., H.-H.S. and C.-K.L.; formal analysis, C.-H.L. and H.-H.S.; investigation, C.-H.L., T.-S.C. and H.-H.S.; resources, T.-S.C. and H.-H.S.; data curation, T.-S.C. and H.-H.S.; writing—original draft preparation, C.-H.L. and H.-H.S.; writing—review and editing, C.-H.L. and C.-K.L.; visualization, C.-H.L. and C.-K.L.; supervision, C.-H.L. and C.-K.L.; project administration, C.-H.L.; funding acquisition, C.-H.L. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was supported in part by Ministry of Science and Technology, Taiwan, under grants MOST 108-2221-E-155-046 and MOST 110-2221-E-155-013.

**Acknowledgments:** The authors would like to thank Taiwan Semiconductor Research Institute (TSRI) for the technical support.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Gallager, R. Low-Density Parity-Check Codes. *IEEE Trans. Inform. Theory* **1962**, *8*, 21–28. [\[CrossRef\]](#)
2. MacKay, D.J.C. Good Error-Correcting Codes Based on Very Sparse Matrices. *IEEE Trans. Inform. Theory* **1999**, *45*, 399–431. [\[CrossRef\]](#)
3. Andriyanov, N.; Dement'ev, V. Topology, Protocols and Databases in Bluetooth 4.0 Sensor Networks. In Proceedings of the 2018 Moscow Workshop on Electronic and Networking Technologies (MWENT), Moscow, Russia, 14–16 March 2018; pp. 1–7.
4. Singh, R.; Baz, M.; Narayana, C.L.; Rashid, M.; Gehlot, A.; Akram, S.V.; Alshamrani, S.S.; Prashar, D.; AlGhamdi, A.S. Zigbee and Long-Range Architecture Based Monitoring System for Oil Pipeline Monitoring with the Internet of Things. *Sustainability* **2021**, *13*, 10226. [\[CrossRef\]](#)
5. Safaldin, M.; Otair, M.; Abualigah, L. Improved Binary Gray Wolf Optimizer and SVM for Intrusion Detection System in Wireless Sensor Networks. *J. Ambient. Intell. Humaniz. Comput.* **2021**, *12*, 1559–1576. [\[CrossRef\]](#)
6. Khasawneh, A.; Kaiwartya, O.; Abualigah, L.; Lloret, J. Green Computing in Underwater Wireless Sensor Networks Pressure Centric Energy Modeling. *IEEE Syst. J.* **2020**, *14*, 4735–4745. [\[CrossRef\]](#)
7. Otair, M.; Ibrahim, O.T.; Abualigah, L.; Altalhi, M.; Sumari, P. An Enhanced Grey Wolf Optimizer based Particle Swarm Optimizer for Intrusion Detection System in Wireless Sensor Networks. *Wirel. Netw.* **2022**, *28*, 721–744. [\[CrossRef\]](#)
8. Thi Bao Nguyen, T.; Nguyen Tan, T.; Lee, H. Efficient QC-LDPC Encoder for 5G New Radio. *Electronics* **2019**, *8*, 668. [\[CrossRef\]](#)
9. Petrović, V.L.; El Mezeni, D.M.; Radošević, A. Flexible 5G New Radio LDPC Encoder Optimized for High Hardware Usage Efficiency. *Electronics* **2021**, *10*, 1106. [\[CrossRef\]](#)
10. Thi Bao Nguyen, T.; Nguyen Tan, T.; Lee, H. Low-Complexity High-Throughput QC-LDPC Decoder for 5G New Radio Wireless Communication. *Electronics* **2021**, *10*, 516. [\[CrossRef\]](#)
11. Lin, C.-H.; Wang, C.-X.; Lu, C.-K. LDPC Decoder Design Using Compensation Scheme of Group Comparison for 5G Communication Systems. *Electronics* **2021**, *10*, 2010. [\[CrossRef\]](#)
12. Su, H.-H.; Chen, T.-S.; Lin, C.-H. Reconfigurable Check Node Unit Design of Dual-Standard LDPC Decoding for 60GHz Wireless Local Area Network. In Proceedings of the 2019 IEEE International Conference on Consumer Electronics-Taiwan (ICCE-TW), Yilan, Taiwan, 20–22 May 2019; pp. 1–2.
13. Richardson, T.J.; Urbanke, R.L. Efficient Encoding of Low-Density Parity-Check Codes. *IEEE Trans. Inform. Theory* **2001**, *47*, 638–656. [\[CrossRef\]](#)
14. Roberts, M.K.; Anguraj, P. A Comparative Review of Recent Advances in Decoding Algorithms for Low-Density Parity-Check (LDPC) Codes and Their Applications. *Arch. Comput. Methods Eng.* **2021**, *28*, 2225–2251. [\[CrossRef\]](#)

15. Jinghu Chen, J.; Fossorier, M.P.C. Near Optimum Universal Belief Propagation Based Decoding of Low-Density Parity Check Codes. *IEEE Trans. Commun.* **2002**, *50*, 406–414. [[CrossRef](#)]
16. Zhao, J.; Zarkeshvari, F.; Banihashemi, A. On Implementation of Min-Sum Algorithm and Its Modifications for Decoding Low-Density Parity-Check (LDPC) Codes. *IEEE Trans. Commun.* **2005**, *53*, 549–554. [[CrossRef](#)]
17. Mansour, M.M.; Shanbhag, N.R. High-throughput LDPC Decoders. *IEEE Trans. VLSI Syst.* **2003**, *11*, 976–999. [[CrossRef](#)]
18. Zhang, Z.; Zhou, L.; Zhou, Z.H. Design of A Parallel Decoding Method for LDPC Code Generated via Primitive Polynomial. *Electronics* **2021**, *10*, 425. [[CrossRef](#)]
19. Cheng, C.-C.; Yang, J.-D.; Lee, H.-C.; Yang, C.-H.; Ueng, Y.-L. A Fully Parallel LDPC Decoder Architecture Using Probabilistic Min-Sum Algorithm for High-Throughput Applications. *IEEE Trans. Circuits Syst.* **2014**, *61*, 2738–2746. [[CrossRef](#)]
20. Tsatsaragkos, I.; Paliouras, V. Approximate Algorithms for Identifying Minima on Min-Sum LDPC Decoders and Their Hardware Implementation. *IEEE Trans. Circuits Syst.* **2015**, *62*, 766–770. [[CrossRef](#)]
21. Català-Pérez, J.M.; Lacruz, J.O.; García-Herrero, F.; Valls, J.; Declercq, D. Second Minimum Approximation for Min-Sum Decoders Suitable for High-Rate LDPC Codes. *Circuits Syst. Signal Process.* **2019**, *38*, 5068–5080. [[CrossRef](#)]
22. Lin, C.-H.; Wu, Y.-S.; Song, C.-P. Energy-Efficient LDPC Codec Design Using Cost-Effective Early Termination Scheme. *IET Comput. Digit. Tech.* **2018**, *13*, 118–125. [[CrossRef](#)]
23. Xiang, B.; Shen, R.; Pan, A.; Bao, D.; Zeng, X. An Area-Efficient and Low-Power Multirate Decoder for Quasi-Cyclic Low-Density Parity-Check Codes. *IEEE Trans. VLSI Syst.* **2010**, *10*, 1447–1460. [[CrossRef](#)]
24. Yen, S.W.; Hung, S.Y.; Chen, C.H.; Chang, H.C.; Jou, S.J.; Lee, C.Y. A 5.79-Gigabits per Second Energy-Efficient Multirate LDPC Codec Chip for IEEE 802.15.3c Applications. *IEEE J. Solid-State Circuits* **2012**, *47*, 2246–2257. [[CrossRef](#)]
25. Oh, D.; Parhi, K.K. Low-Complexity Switch Network for Reconfigurable LDPC Decoders. *IEEE Trans. VLSI Syst.* **2010**, *18*, 85–94. [[CrossRef](#)]
26. Beneš, V.E. Optimal Rearrangeable Multistage Connecting Networks. *Bell Syst. Tech. J.* **1964**, *43*, 1641–1656. [[CrossRef](#)]
27. Lin, C.-H.; Hsieh, C.-W. Low-Routing-Complexity Convolutional/Turbo Decoder Design for Iterative Detection and Decoding Receivers. *IEEE Trans. Circuits Syst. I Regul. Pap.* **2019**, *66*, 4476–4489. [[CrossRef](#)]
28. Li, M.; Naessens, F.; Debacker, P.; Raghavan, P.; Desset, C.; Li, M.; Dejonghe, A.; Van der Perre, L. An Area and Energy Efficient Half-Row-Paralleled Layer LDPC Decoder for the 802.11Ad Standard. In Proceedings of the 2013 IEEE Workshop on Signal Processing Systems (SiPS), Taipei, Taiwan, 16–18 October 2013; pp. 112–117.
29. Motozuka, H.; Yosoku, N.; Sakamoto, T.; Tsukizawa, T.; Shirakata, N.; Takinami, K. A 6.16 Gigabits per Second 4.7 pJ/Bit/Iteration LDPC Decoder for IEEE 802.11 ad Standard in 40 nm LP-CMOS. In Proceedings of the 2015 IEEE Global Conference on Signal and Information Processing (GlobalSIP), Orlando, FL, USA, 14–16 December 2015; pp. 1289–1292.
30. Milicevic, M.; Gulak, P.G. A Multi-Gb/s Frame-Interleaved LDPC Decoder with Path-Unrolled Message Passing in 28-nm CMOS. *IEEE Trans. VLSI Syst.* **2018**, *26*, 1908–1921. [[CrossRef](#)]
31. Li, M.; Weijers, J.W.; Derudder, V.; Vos, I.; Rykunov, M.; Dupont, S.; Debacker, P.; Dewilde, A.; Huang, Y.; Van der Perre, L.; et al. An Energy Efficient 18Gigabits per Second LDPC Decoding Processor for 802.11Ad in 28 nm CMOS 2015. In Proceedings of the 2015 IEEE Asian Solid-State Circuits Conference (A-SSCC), Xiamen, China, 9–11 November 2015; pp. 1–5.
32. Lee, X.; Chen, C.; Chang, H.; Lee, C. A 7.92 Gigabits per Second 437.2 mW Stochastic LDPC Decoder Chip for IEEE 802.15.3c Applications. *IEEE Trans. Circuits Syst.* **2015**, *62*, 507–516. [[CrossRef](#)]
33. Chen, Z.; Peng, X.; Zhao, X.; Okamura, L.; Zhou, D.; Goto, S. A 6.72-Gigabits per Second 8 pJ/Bit/Iteration IEEE 802.15.3c LDPC Decoder Chip. *IEICE Trans. Fundam. Electron. Commun. Comput. Sci.* **2013**, *96*, 2623–2632. [[CrossRef](#)]
34. Li, M.-R.; Yang, C.-H.; Ueng, Y.-L. A 5.28-Gigabits per Second LDPC Decoder with Time-Domain Signal Processing for IEEE 802.15.3c Applications. *IEEE J. Solid-State Circuits* **2017**, *52*, 592–604. [[CrossRef](#)]