International Journal on Advanced Science Engineering Information Technology

Vol.15 (2025) No. 2
ISSN: 2088-5334

Design and Implementation of Four-Parallel Turbo Decoder for HomePlug Green PHY

Jeongju Jeon [a], Sunhee Kim [a,*]

[a] *Department of System Semiconductor Engineering, Sangmyung University, Cheonan-si, Chungcheongnam-do, Republic of Korea*
Corresponding author: [*]happyshkim@smu.ac.kr

***Abstract*—As electric vehicles are popularized, the importance of the safety of electric vehicle batteries has increased. Lithium-ion batteries, which are mainly used as batteries for electric vehicles, have the possibility of fire due to thermal factors, collisions, and overcharging. Recently, to prevent overcharging, battery information is exchanged between electric vehicle chargers and electric vehicle battery management systems using PLC. HomePlug Green Phy(HPGP) is a communication method for Smart Grid applications and EVs among PLCs. HPGP uses 3072-OFDM, and the number of available carriers varies depending on the region. One PHY block is generated with several to dozens of OFDM symbols. Therefore, to guarantee a constant latency of the PHY transceiver, a high-speed transceiver that can process FEC within the required time is needed. In this paper, we propose a 4-parallel turbo decoder. By analyzing channel interleaver for HPGP, four turbo decoders can be processed simultaneously. Through analysis of the turbo interleaving/deinterleaving address, the turbo decoder shares the data memory of the channel deinterleaver. It reduces the interleaving/deinterleaving address memory size, thereby reducing the overall memory size. The proposed architecture was designed with Verilog, and its functions were verified using VCS/Verdi. The Kintex UltraScale Xcku11p-ffve1517-2-e was used as the target for implementation. The clock frequency is 125 MHz. After analyzing the overall architecture of HPGP FEC, the design was made considering the timing and interface with the pre/post blocks of the Encoder and Decoder, so it will be well used in the HPGP transceiver.**

***Keywords*—HPGP; interleaver; parallel processing; power line communication; turbo code.**

Manuscript received 30 Aug. 2024; revised 28 Oct. 2024; accepted 18 Dec. 2024. Date of publication 30 Apr. 2025.

I. INTRODUCTION

Electric Vehicles (EV) have attracted attention as a countermeasure to reduce global warming and environmental pollution because they are higher energy efficiency and lower CO2 emissions compared to fossil fuel-powered vehicles [1]-[4]. According to the IEA's Global Electric Vehicle Outlook 2024, global electric vehicle sales reached approximately 14 million in 2023, accounting for 18% of total vehicle sales, and cumulative EV sales reached about 40 million units [5]. Many countries are providing tax and subsidy benefits for EVs and are strengthening carbon emission regulations for internal combustion engine vehicles, so the transition to EVs is expected to continue [6]-[9]. However, access to electric vehicles is still an issue due to the lack of charging stations. Especially, the cost and limited lifespan of batteries, as well as fires due to their instability, are challenges that need to be addressed [10]-[12].

Lithium-ion batteries are widely used as batteries for EVs. However, they sometimes rise sharply in temperature and lead to fire due to internal and external thermal factors or chemical/physical collisions [13]-[15]. In addition, when overcharging is performed, the anode material and the cathode material can come into contact with each other due to damage to the anode-cathode separator, which can cause a simultaneous release of energy and a fire [16]-[18]. Recently, EV's Battery Management Systems (BMS) that can prevent overcharging have been expanded [19]-[21]. In addition, by installing a modem in both EVs and chargers for EVs, the chargers can receive the charging state information of batteries from the BMS of EVs to prevent overcharging by the charger itself. Powerline communication (PLC) through a power supply cable is mainly used as communication between the chargers and the EVs [22]-[25].

Among PLC technologies, HomePlug Green PHY(HPGP) is a solution for smart grid applications and EVs [26]. HPGP is limited to three ROBust OFDM (ROBO)-Audio-Video (AV) modes among the physical layers of HPAV. It also supports only Quadrature Phase Shift Keying (QPSK), and rate 1/2 turbo convolutional coding [27]-[30].

HPGP performs turbo coding, channel interleaving, and ROBO interleaving for forward error correction (FEC). It uses 3072-FFT, of which 1155 carriers are used. Depending on the regulatory constraints of the region in use, it uses a minimum of 275 carriers and a maximum of 1155 carriers. Therefore, it is necessary to design FEC with a parallel structure to keep the latency of PHY constant regardless of tone masking [30]. This paper proposes a 4-parallel turbo decoder by analyzing its FEC algorithm.

Turbo coding consists of turbo interleaving and two recursive systematic convolutional (RSC) encodings. When the turbo interleaving formula is simple or has a clear pattern, the interleaving/deinterleaving address is calculated in real time [31]. On the other hand, when the formula is complex, storing the interleaver formula's address value in memory is often used [32]. This manuscript used a structure that stores the interleaving/deinterleaving address in the LUT. At this time, the interleaving/deinterleaving address and data are rearranged to enable 4-parallel rsc decoding. This shows the results of shortening the operating time, reducing the size of the interleaving/deinterleaving address LUT, and increasing efficiency.

II. Materials and Method

A. HomePlug Green PHY

Fig. 1 shows a block diagram of the transmitter and receiver for the HPGP physical layer. Though HPGP supports HomePlug 1.0.1 Frame Control (FC), HomePlug AV Frame Control, and HomePlug AV Payload, this paper designs HomePlug AV FC and HomePlug AV Payload as shown in Fig. 1. HomePlug AV FC is processed by turbo convolutional encoding, channel interleaving, and frequency diversity copying. HomePlug AV payload is processed by scrambling, turbo convolutional encoding, channel interleaving, and ROBO interleaving. HPGP OFDM modulation consists of a mapper, which supports only QPSK, 3072-point Inverse Fast Fourier Transform (IFFT), cycle prefix and preamble insertion.

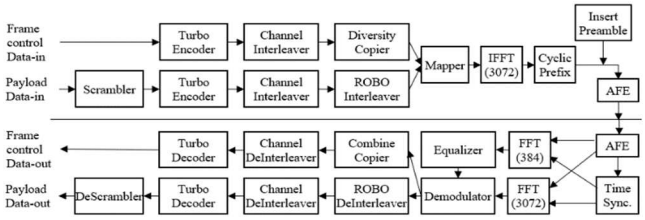

Fig. 1 HPGP transceiver

The receiver obtains data by reversing the transmitter's processing of FC and payload. FC and payload are processed with a 3072-point FFT. After the 384-point FFT, the preamble obtains information for FC and payload through an equalizer and demodulator.

Turbo coding is applied to both the FC and the payload. In both cases, the encoding/decoding algorithm is the same, except for the information length. The information of the turbo encoder is the data transmitted from the MAC layer when it is FC, and the data passed through the scrambler when it is payload. Both data are 1-bit serial data. Therefore, since the data format and timing are the same, with only the information length being different, both FC and payload are processed with a single turbo encoder and turbo decoder.

The output of the turbo encoder is transmitted to the channel interleaver. Channel interleaver divides the input data into four parts as shown in Figure 2. Then, when outputting the data, it takes one data from each of the four parts and makes the four data into one pair. Therefore, the channel interleaver outputs the data after receiving all the data from the turbo encoder.

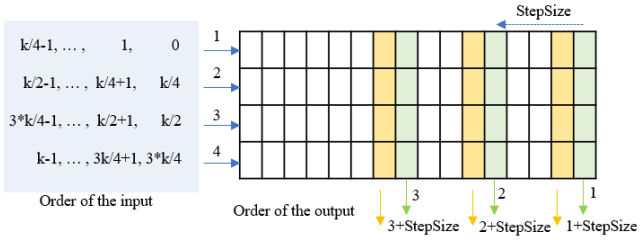

Fig. 2 The order of input and output of channel interleaver

At the receiver, the output data of the channel deinterleaver is sent to the turbo decoder. Since the channel deinterleaver operates in reverse concerning the channel interleaver, the data to the channel deinterleaver is input simultaneously from four parts based on the data to be output. In other words, when the data reception of the first part is finished, the data reception of the fourth part is also finished at the same time. Therefore, in this paper, we propose and design a 4-parallel structure that processes turbo decoding for each of the four parts simultaneously by reflecting the characteristics of the channel deinterleaver.

B. Turbo Encoding Algorithm

The turbo encoder consists of two RSC encoders and a turbo interleaver. The RSC encoder generates parity data for the information data. The turbo interleaver is a block that changes the order of information data to generate two different parity data for the information data. Turbo coding is used for FC and payload. The physical blocks (PB) size processed by turbo coding is 16 bytes for FC and 136 bytes or 520 bytes for payload. The interleaving method varies depending on the PB size.

As shown in Fig. 3, the RSC encoder generates a 1-bit parity p for a 2-bit information $\{u_1, u_2\}$. The first input is u_1, and the second input is u_2. Since the parity bit is affected by the current 1-pair input and the previous 3-pair inputs, the encoder has a total of 8 states. The two RSC encoders each generate 1-bit parity for the 2-bit information. The code rate of puncturing is 1/2, and 2 bits are generated for 2-bit information. Therefore, the puncturing block is not necessary.

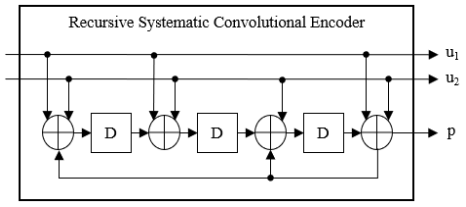

Fig. 3 Block diagram of Recursive Systematic Convolutional Encoder

A turbo encoder requires a turbo interleaver to rearrange the order of information data. The turbo interleaver interleaves the PBs in two-bit units instead of a single bit,

keeping the original bit pairs together. Equation 1 determines the output of the turbo interleaver using the parameters defined in Table 1 according to the PB size.

$$l(x) = [S(x \bmod N) - (x \operatorname{div} N) * N + L] \bmod L \quad for\ x = 0, 1, \ldots, (L-1) \tag{1}$$

where div is integer division operation, mod is modulo operation. S(x) is seed table, N is seed table length, and L is interleaver length. The interleaver length is equal to the size of dual-bit block for PB Size, 64 (=16×8/2), 544 (=136×8/2), 2080 (=520×8/2). Table 2 is the seed table when PB size is 16. The values of S(x) are determined for the remainders 0 to 7 when divided by N=8 for interleaver length 64.

TABLE I
INTERLEAVER PARAMETERS

PB Size (Octets)	N Value	M Value	Interleaver Length L
16	8	8	64
136	34	16	544
520	40	52	2080

TABLE II
INTERLEAVER SEED TABLE FOR PB-16 OCTETS

x	0	1	2	3	4	5	6	7
S(x)	54	23	61	12	35	2	40	25

Interleaved information bit pairs are swapped when interleaver's output index x is even. Two RSC encoders each process the entire data twice. In the first pass, the initial state of the encoder is set to all zeros. After passing through the entire information blocks, the initial state of the second pass is set through the given matrix operation on the final state. The parity generated in the second pass is transmitted to the channel interleaver.

C. Turbo Decoding Algorithm

Fig. 4 shows a conceptual block diagram of the turbo decoder. RSC decoders 1 and 2 correspond to RSC encoders 1 and 2. RSC decoding is based on soft output and uses the Maximum A-Posteriori (MAP) algorithm [33]. This paper uses Maximum Log-MAP to reduce the computational complexity [33], [34]. In soft output, the probability of the output data is calculated according to the Log Likelihood Ratios (LLR). The probability information calculated in RSC decoder 1 is transferred to RSC decoder 2 through the interleaver, and the probability information calculated in RSC decoder 2 is transferred to RSC decoder 1 through the deinterleaver. One decoder repeats the probability calculation using the probability information received from the other decoder, thereby increasing the accuracy of the probability of the decision of the received data.

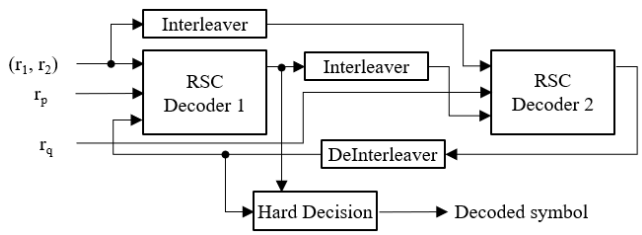

Fig. 4 Conceptual block diagram of the turbo decoder

The HPGP RSC encoder generates 1-bit parity for 2-bit information data, so probability calculations for received data are processed in a 2-bit pair. The HPGP RSC encoder has 8 states and forms a trellis diagram as shown in Fig. 5 according to the convolution relationship. In Fig. 5, s is a state. Alpha, beta, and gamma are forward recoverable variable, backward recoverable variable, and branch metric variable, respectively.

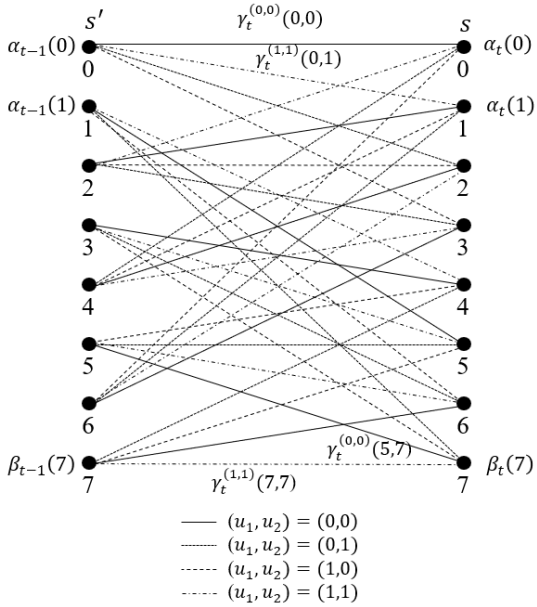

Fig. 5 Trellis diagram for turbo decoding

The gamma is computed as follows [35, 36]

$$\gamma_t^{(u_1,u_2)}(s', s) = C \cdot el_t(u_1, u_2) + \frac{Lc}{2}(u_1 \cdot r_1 + u_2 \cdot r_2 + p \cdot r_p) \tag{2}$$

where r, u, and p represent the received bit, the information bit, and parity bit, respectively, when transitioning from state s' to state s. el represents the extrinsic LLR.

The alpha and beta are computed as follows [35, 36]

$$\alpha_t(s) = \max_{s'}(\alpha_{t-1}(s') + \gamma_t^{(r_1,r_2)}(s', s)) \tag{3}$$

$$\beta_{t-1}(s') = \max_{s}(\beta_t(s) + \gamma_t^{(r_1,r_2)}(s', s)) \tag{4}$$

a-posteriori LLR and extrinsic LLR are computed as follows.

$$post_t(u_1, u_2) = \max_{s}(\alpha_t(s') + \gamma_t^{(u_1,u_2)}(s', s) + \beta_{t+1}(s')) \tag{5}$$

$$el_t(u_1, u_2) = post_t(u_1, u_2) - post_t(0,0) \tag{6}$$

D. Turbo Encoder Structure

Fig. 6 shows a block diagram of a turbo encoder. It consists of a serial-to-parallel block, a turbo interleaver, two rsc blocks, and a controller. The serial-to-parallel block converts 1-bit data input into a 2-bit pair.

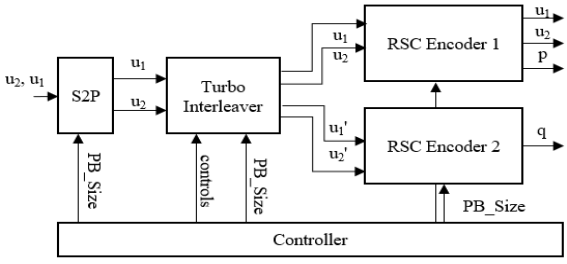

Fig. 6 Block diagram of turbo encoder

RSC encoder 1 generates a parity bit p for the information data. RSC encoder 2 generates a parity bit q for the interleaved information data. The output of the turbo encoder is a 4-pair {u1, u2, p, q}. Therefore, RSC encoder 1 waits until the input data of encoder 2 is ready, that is, until the entire PB size information for turbo interleaving is input and then generates a parity bit at the same time.

Fig. 7 shows a block diagram of turbo interleaver. Input information data is sequentially stored in memory in the form of 2-bit pairs. When the information is stored as much as the PB size, memory changes from write mode to read mode. Two read addresses are given for encoders 1 and 2, respectively. Since the data transmitted to encoder 1 is in the same order as the original data, read address 1 is designed as a counter that increases by 1 from 0. On the other hand, the data transmitted to encoder 2 is turbo interleaved data. Therefore, memory read address 2 is the value calculated by Equation (1). The interleaving address values of Equation (1) are stored in the look-up table (LUT). Since the address values of PB-16, 136, and 520 must be stored, the LUT stores (16 + 136 + 520) * 8/2 = 2688 values and the stored values are 12 bits because it is 0 to 2080 (= 520 * 8/2) - 1. When read data 2 is an even number, 2-bit data are swapped and output.

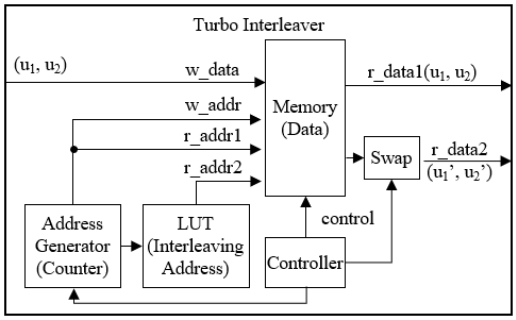

Fig. 7 Block diagram of the proposed turbo interleaver

E. Turbo Decoder Structure

Fig. 8 shows a proposed turbo decoder block diagram. There are 4 decoder blocks for 4-parallel processing and 2 memories for extrinsic LLRs. The input data {r_1, r_2, r_p, r_q } of the turbo decoder is in the memory of the channel deinterleaver. For 4-parallel processing, 4 pairs of data are stored at one address in memory.

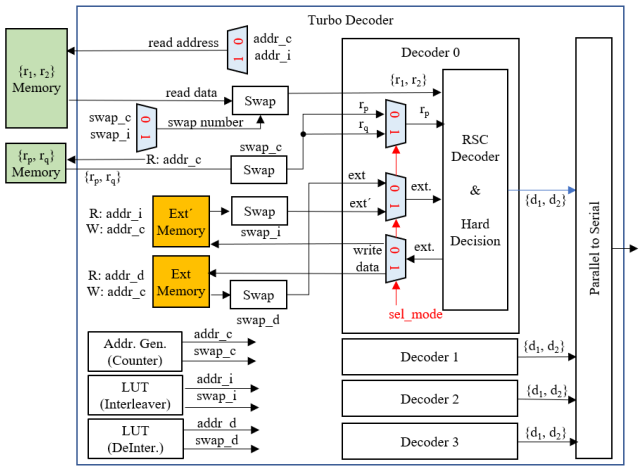

Fig. 8 Block diagram of the proposed turbo decoder

Turbo decoder performs rsc decoding 1 and 2 in turn and repeats this process 4-8 times. In this design, rsc decoder 1 and 2 were not designed separately as shown in Fig. 8, but input and output data were selected for one rsc decoder to perform both decodings. In the first decoding of a process, sel_mode is 0, and in the second decoding, sel_mode is 1.

While sel_mode = 0, the original data {r_1, r_2} and the parity bit r_p of the original data are used, so the address addr_c, which sequentially increases based on the counter is given as the read address for the information and parity data memory of the Channel deinterleaver. The read data is distributed to decoder 0, 1, 2, and 3 through the swap process. The swap algorithm will be explained in detail when explaining turbo interleaver/deinterleaver. Extrinsic LLRs stored in ext. memory are deinterleaved and then distributed to the decoders through the swap process.

The extrinsic LLRs generated by the RSC decoder during sel_mode = 0 are values to be used during the other mode, i.e., sel_mode = 1, so they are stored in ext.' memory. At this time, ext LLRs calculated by decoder 0, 1, 2, and 3 are concatenated and stored in one memory address in the ext.' memory.

While sel_mode = 1, interleaved data {r_1', r_2'}and parity bit r_q of interleaved data are used. In the information memory of the channel deinterleaver, interleaving address addr_i is given as a read address, and the read data is distributed to decoder 0, 1, 2, and 3 through the swap process according to swap_i. In the parity memory, address addr_c, which increases sequentially based on the counter, is given as a read address, and is distributed to decoder 0, 1, 2, and 3 through the swap process according to swap_c. Extrinsic LLRs stored in ext.' memory are interleaved and then distributed to the decoder through the swap process.

The extrinsic LLRs generated by the RSC decoder during sel_mode = 1 are values to be used during sel_mode = 0, so they are stored in ext. memory. At this time, the ext LLRs calculated in decoder 0, 1, 2, and 3 are concatenated and stored in one memory address in ext memory.

As in the turbo encoder, the interleaving and de-interleaving address values are stored in the LUT. In order to process turbo decoding in 4-parallels, unlike the encoder interleaver, 4 data must be accessed at a time.

Considering the logical data order stored in the channel deinterleaver, we analyzed the address values stored in the LUT of the turbo interleaver and deinterleaver. When the address values are divided into four parts according to the PB size, it was confirmed that there are some rules between the data addresses that must be transmitted to each RSC decoder. For example, Fig. 9 shows the relationship between the value of the interleaving address and the RSC decoder when the PB size is 16 bytes. The horizontal lines are the address values of the data that must be input for one RSC decoder. For RSC decoding, data at address 54, 23, and 61 must be input in order to RSC decoder 0. Similarly, data at address 38, 7, and 45 must be input in order to RSC decoder 1. The vertical columns are the address values of the data that must be read simultaneously. That is, data at addresses 54, 38, 22, and 6 must be read simultaneously and input to each RSC decoder, and then data at addresses 23, 7, 55, and 39 must be read simultaneously.

54	23	61	12	35	2	40	25	46	15	53	4	27	58	32	17
38	7	45	60	19	50	24	9	30	63	37	52	11	42	16	1
22	55	29	44	3	34	8	57	14	47	21	36	59	26	0	49
6	39	13	28	51	18	56	41	62	31	5	20	43	10	48	33

RSC decoder 0
RSC decoder 1
RSC decoder 2
RSC decoder 3

1st 2nd → 15th last

Fig. 9 Relationship between the value of the interleaving address and the RSC decoder

When the interleaving address value for PB 16 is divided by 16, the quotient and remainder of 16 (= (Interleaver length 64)/4) are organized as shown in Fig. 10. The remainders for one column are all the same and only the quotient is different. Since channel deinterleaver operates the inverse of the channel interleaver, it receives, and stores four data separated by 16 intervals at the same time. In other words, four data with the same remainder are received at the same time. Therefore, in this design, four data are stored in one memory address. At this time, the data of the first row was assigned to the MSB, and the data of the fourth row was assigned to the LSB. In our design, the soft data output is 4 bits. Since 4 pairs of $\{r_1, r_2, r_p, r_q\}$ are stored, the total word length of the memory is 64 (=4×4×4) bits. That is, when the interleaving address value is divided by 16, the remainder is the address of the data memory, and the quotient is the location of the 16-bit data to be selected from the 60-bit read data.

3_6	1_7	3_13	0_12	2_3	0_2	2_8	1_9	2_14	0_15	3_5	0_4	1_11	3_10	2_0	1_1
2_6	0_7	2_13	3_12	1_3	3_2	1_8	0_9	1_14	3_15	2_5	3_4	0_11	2_10	1_0	0_1
1_6	3_7	1_13	2_12	0_3	2_2	0_8	3_9	0_14	2_15	1_5	2_4	3_11	1_10	0_0	3_1
0_6	2_7	0_13	1_12	3_3	1_2	3_8	2_9	3_14	1_15	0_5	1_4	2_11	0_10	3_0	2_1

Fig. 10 Quotient and remainder of 16 of the interleaving address values

Since the memory addresses of the four data to be read simultaneously are the same, the number of address values stored in the LUT is reduced from 4 to 1. Instead, 8 bits indicating the bit position are added. The address value is reduced from 12 bits to 10 bits, and the overall memory size is reduced from (12×4) × (2^12) to (10+8) × (2^10) bits. The values stored in the LUT are in the form of Fig. 11. The LUT stores the address (addr_i) of the data to be read and the position of the bit in the data (swap_i). For example, when it is PB16, the first interleaving data is stored at address 6 (the remainder is 6). Then, the data bits are divided and distributed to the RSC decoder according to the quotient. In interleaving of the first data, the data to be transmitted to RSC decoder 0 is stored in the location where the quotient is 3, so it is the lowest 16 bits (bit 15-0) of the data read from address 6. The upper 16 bits (bits 31-16) are passed to RSC decoder 1, bits 47-32 to RSC decoder 2, and the most significant 63-48 bits are passed to RSC decoder 3. The second interleaved data is read from address 7, bits 15-0 are passed to RSC decoder 2, bits 32-16 are passed to RSC decoder 3, bits 47-32 are passed to RSC decoder 0, and the most significant 63-48 bits are passed to RSC decoder 1.

3210 _6	1032 _7	3210 _13	0321 _12	2103 _3	0321 _2	2103 _8	1032 _9	2103 _14	0321 _15	3210 _5	0321 _4	1032 _11	3210 _10	2103 _0	1032 _1

Fig. 11 Format of values stored in the LUT

When interleaving, data is swapped when the address of the LUT is even. On the other hand, when deinterleaving, data is swapped when the address stored in the LUT is even. The rules regarding the interleaving address and data location mentioned appear the same in the interleaver/deinterleaver of PB-16, 136, and 520. Fig. 12 shows a block diagram of the RSC decoder. It consists of cal_gamma, cal_alpha, cal_beta, cal_extP, cal_controller, and hard decision blocks. Gamma and alpha are calculated in the order of data input, but beta must be calculated backward, so memory for storing gamma and alpha values is added.

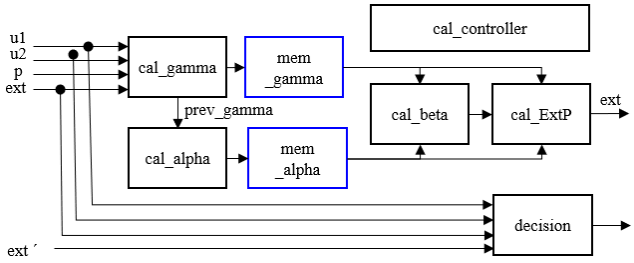

Fig. 12 Block diagram of the RSC decoder

The RSC encoder has eight states and processes in a 2-bit pair, which makes four combinations $\{00_2, 01_2, 10_2, 11_2\}$, so gamma must be calculated for a total of 32 cases each for all input data. However, as a result of analyzing the trellis diagram for HPGP RSC decoding in Fig. 5, input combinations available in even and odd states are distinguished. So only four available inputs in two states, that is, a total of eight gamma values, need to be calculated. Therefore, the computational logic and memory capacity are reduced.

Similarly, when calculating a-posteriori LLR, we need to find the maximum value among a-posteriori LLRs for (u_1, u_2) for each state. Since gamma included in the a-posteriori LLR of Equation 6 has the same value, we can find the maximum value among the values not included in gamma. The hardware size and time are reduced.

Alpha and beta are also calculated as much as the data length. Unlike gamma, alpha and beta continue to increase in value as gamma is added to the previously calculated alpha and beta as shown in Equations (3) and (4). However, alpha and beta compare the values for 8 states for one input data and leave the maximum value. In other words, relative values are meaningful, not absolute values. Therefore, for hardware implementation, alpha and beta for 8 states are calculated for each input data, and then the minimum value among them is subtracted from the 8 alpha and beta values, leaving only the minimum value that can be compared for 8 states. The values of alpha and beta are designed to not increase continuously but remain at a certain level.

When making a hard decision, the final calculated probability value is used to select the most probable one among the four input cases for each data. Since it is determined by a 2-bit pair, 1 bit is output in order through 2-bit parallel to serial.

III. Results and Discussion

The proposed turbo encoder and decoder for HPGP were designed in Verilog-HDL, simulated in VCS/Verdi, and verified in FPGA. Fig. 13 shows the simulation results of the turbo encoder in Verdi. 1-bit data is inputted continuously to the turbo encoder. 2-bit data is outputted once every two clocks through the serial-to-parallel module and stored in the

memory. After the input data is fully stored, the data is transmitted to RSC encoders 1 and 2. At this time, data in the same order as the input data is inputted to RSC encoder 1, and interleaved data is inputted to RSC encoder 2. The 2-bit data is inputted to the encoder. RSC encoding is repeated twice, and the second result, information u1, u2 and parity p, q, are outputted as the final output.

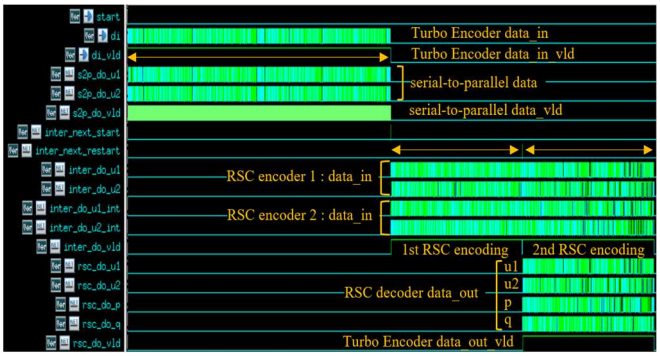

Fig. 13 Simulation results of turbo encoder in Verdi

Fig. 14 shows the simulation results of the turbo decoder in Verdi. In this result, turbo decoding is set to repeat twice (the 1st turbo decoding and the 2nd turbo decoding). There are 4 RSC decoders according to the 4-Parallel structure, and data is input to each simultaneously. One turbo decoding must do RSC decoding twice (decoding 1 and decoding 2).

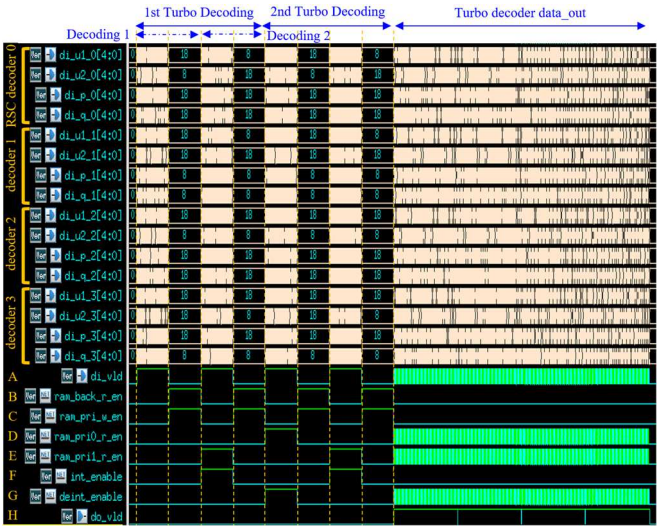

Fig. 14 Simulation results of turbo decoder in Verdi

In each decoding operation, gamma and alpha are calculated during the forward section (A) when data is input. While calculating beta during the backward section (B), the extrinsic LLR value is calculated and stored (C). While calculating gamma, the extrinsic LLR value calculated in the previous decoding must be read. Since there is no previously calculated value in the first decoding, the extrinsic value is read from the 2nd decoding during the 1st turbo decoding (D, E). During decoding 2, interleaving must be performed when reading data u_1, u_2 (F), and during decoding 1, previously stored extrinsic LLR values must be deinterleaved (G). After two turbo decoding, the final data (H) of the turbo decoder is output. Looking at the time interval, it can be seen that it is 8 times the decoding input time. This is because the RSC decoder was operated in 4-parallel, and during decoding, 2-bit was processed as one pair.

Table 3 is the summary of FPGA device utilization. The proposed turbo encoder and decoder was implemented targeting Kintex UltraScale Xcku11p-ffve1517-2-e using Vivado 19.2. The clock frequency is 125 MHz. Since the information data in turbo encoder was stored in flipflops instead of memory, CLB Registers and CLB LUTs were used instead of BlockRAM.

TABLE III
SUMMARY OF FPGA DEVICE UTILIZATION

Resource	Utilization	
	Encoder	Decoder
CLB LUTs	6709	5424
CLB Registers	4372	1014
Carry8	0	451
F7 Muxes	1311	33
Block RAM Tile	0	37.5

The designed turbo encoder and turbo decoder operate correctly for all cases of PB16, 136, and 520, and the Turbo decoder uses a 4-parallel structure to shorten the decoding time. After analyzing the entire structure of HPGP FEC, it was designed by considering the timing and interface with the pre/post blocks of the Encoder and Decoder, so it will be well used in the HPGP transceiver.

IV. CONCLUSION

In this paper, we design a turbo encoder and decoder for HPGP. HPGP uses OFDM and the number of carriers used varies from 275 carriers to a maximum of 1155 carries depending on the usage area. So high-speed parallel processing is required to maintain a certain latency. We analyzed the FEC algorithm of HPGP and proposed a 4-parallel turbo decoder structure. By analyzing the address values of turbo interleaver and deinterleaver, we enabled simultaneous processing of four decoders. In addition, we reduced the area by eliminating redundant gamma and extrinsic LLR calculations in decoding through analysis of the trellis diagram of encoding. After designing the turbo encoder and decoder in Verilog HDL according to the proposed structure, we verified its function through simulation in VCS/Verdi environment. When synthesized with Kintex UltraScale Xcku11p-ffve1517-2-e target using Vivado 19.2, the clock frequency is 125 MHz. It was confirmed that the designed turbo encoder and decoder operate correctly for all cases of PB16, 136, and 520.

ACKNOWLEDGMENT

This research was funded by a 2024 research Grant from Sangmyung University(2024-A001-0123).

REFERENCES

[1] J. A. Sanguesa, V. Torres-Sanz, P. Garrido, F. J. Martinez, J. M. Marquez-Barja, "A Review on Electric Vehicles: Technologies and Challenges," *Smart Cities*, vol. 4, no. 1, pp. 372-404, 2021, doi:10.3390/smartcities4010022.

[2] S. M. Arif, T. T. Lie, B. C. Seet, S. Ayyadi, K. Jensen, "Review of Electric Vehicle Technologies, Charging Methods, Standards and Optimization Techniques," *Electronics*, vol. 10, no. 16, 1910, 2021, doi: 10.3390/electronics10161910.

[3] G. Krishna, "Understanding and identifying barriers to electric vehicle adoption through thematic analysis," *Transportation Research Interdisciplinary Perspectives*, vol. 10, 100364, June 2021, doi:10.1016/j.trip.2021.100364.

[4] S. Islam, A. Iqbal, M. Marzband, I. Khan, A. M.A.B. Al-Wahedi, "State-of-the-art vehicle-to-everything mode of operation of electric vehicles and its future perspectives," *Renewable and Sustainable Energy Reviews*, vol. 166, 112574, Sept. 2022, doi:10.1016/j.rser.2022.112574.

[5] IEA, "Global EV Outlook 2024," Internal Energy Aagency, Paris, Apr. 2024. [Online]. Available: https://www.iea.org/reports/global-ev-outlook-2024.

[6] P. Barman, L. Dutta, S. Bordoloi, A. Kalita, P. Buragohain, S. Bharali, B. Azzopardi, "Renewable energy integration with electric vehicle technology: A review of the existing smart charging approaches," *Renewable and Sustainable Energy Reviews*, vol. 183, 113518, 2023, doi: 10.1016/j.rser.2023.113518.

[7] S. LaMonaca, L. Ryan, "The state of play in electric vehicle charging services – A review of infrastructure provision, players, and policies," *Renewable and Sustainable Energy Reviews*, vol. 154, 111733, 2022, doi: 10.1016/j.rser.2021.111733.

[8] S. S. G. Acharige, M. E. Haque, M. T. Arif, N. Hosseinzadeh, K. N. Hasan and A. M. T. Oo, "Review of Electric Vehicle Charging Technologies, Standards, Architectures, and Converter Configurations," *IEEE Access*, vol. 11, pp. 41218-41255, Apr. 2023, doi: 10.1109/access.2023.3267164.

[9] S.S. Ravi, M. Aziz, "Utilization of Electric Vehicles for Vehicle-to-Grid Services: Progress and Perspectives," *Energies*, vol. 15, no. 2, pp. 589. Jan. 2022, doi: 10.3390/en15020589.

[10] M. R. Khalid, I. A. Khan, S. Hameed, M. S. J. Asghar, J.-S. Ro, "A comprehensive review on structural topologies power levels energy storage systems and standards for electric vehicle charging stations and their impacts on grid," *IEEE Access*, vol. 9, pp. 128069-128094, Sept. 2021, doi: 10.1109/access.2021.3112189.

[11] M. R. Khalid, I. A. Khan, S. Hameed, M. S. J. Asghar, J. Ro, "A Comprehensive Review on Structural Topologies, Power Levels, Energy Storage Systems, and Standards for Electric Vehicle Charging Stations and Their Impacts on Grid," *IEEE Access*, vol. 9, pp. 128069-128094, Sept. 2021, doi: 10.1109/access.2021.3112189.

[12] F. Adegbohun, A. v. Jouanne, K. Y. Lee, "Autonomous Battery Swapping System and Methodologies of Electric Vehicles," *Energies*, vol. 12, no. 4. pp. 667, Feb. 2019, doi: 10.3390/en12040667.

[13] X. Zhang, Z. Li, L. Luo, Y. Fan, Z. Du, "A review on thermal management of lithium-ion batteries for electric vehicles," *Energy*, vol. 238, part A, 121652, Jan. 2022, doi: 10.1016/j.energy.2021.121652.

[14] V.G. Choudhari, Dr A.S. Dhoble, T.M. Sathe, "A review on effect of heat generation and various thermal management systems for lithium ion battery used for electric vehicle," *Journal of Energy Storage*, vol. 32, 101729, Dec. 2020, doi: 10.1016/j.est.2020.101729.

[15] C. Roe, X. Feng, G. White, R. Li, H. Wang, X. Rui, C. Li, F. Zhang, V. Null, M. Parkes, Y. Patel, Y. Wang, H. Wang, M. Ouyang, G. Offer, B. Wu, "Immersion cooling for lithium-ion batteries – A review," *Journal of Power Sources*, vol. 525, 231094, March 2022, doi:10.1016/j.jpowsour.2022.231094.

[16] J. Hong, Z. Wang, C. Qu, Y. Zhou, T. Shan, J. Zhang, Yankai Hou, "Investigation on overcharge-caused thermal runaway of lithium-ion batteries in real-world electric vehicles," *Applied Energy*, vol. 321, 119229, Sept. 2022, doi: 10.1016/j.apenergy.2022.119229.

[17] N. Mao, T. Zhang, Z. Wang, Q. Cai, "A systematic investigation of internal physical and chemical changes of lithium-ion batteries during overcharge," *Journal of Power Sources*, vol. 518, 230767, Jan. 2022, doi: 10.1016/j.jpowsour.2021.230767.

[18] J. Jaguemont, F. Bardé, "A critical review of lithium-ion battery safety testing and standards," *Applied Thermal Engineering*, vol. 231, 121014, Aug. 2023, doi: 10.1016/j.applthermaleng.2023.121014.

[19] T. N. V. Krishna, S. V. S. V. P. D. Kumar, S. Srinivasa Rao, L. Chang, "Powering the Future: Advanced Battery Management Systems (BMS) for Electric Vehicles," *Energies*, vol. 17, 3360, July 2024, doi:10.3390/en17143360.

[20] S. Thangavel, D. Mohanraj, T. Girijaprasanna, S. Raju, C. Dhanamjayulu, S. M. Muyeen, "A Comprehensive Review on Electric Vehicle: Battery Management System, Charging Station, Traction Motors," *IEEE Access*, vol. 11, pp. 20994-21019, Feb. 2023, doi:10.1109/access.2023.3250221.

[21] A. Jose, S. Shrivastava, "Evolution of Electrical Vehicles, Battery State Estimation, and Future Research Directions: A Critical Review," *IEEE Access*, vol. 12, pp. 158627-158646, Oct. 2024, doi:10.1109/access.2024.3482698.

[22] J. Johnson, T. Berg, B. Anderson, B. Wright, "Review of Electric Vehicle Charger Cybersecurity Vulnerabilities, Potential Impacts, and Defenses," *Energies*, vol. 15, no. 11, May 2022, doi:10.3390/en15113931.

[23] M. J. Koshkouei, E. Kampert, A. D. Moore, M. D. Higgins, "Impact of Battery State of Charge on In-Situ Power Line Communication Within an Intelligent Electric Vehicle," in *Proc. 2022 IEEE 25th ITSC*, Macau, China, 2022, pp. 3855-3860, doi:10.1109/itsc55140.2022.9921800.

[24] M. J. Koshkouei, E. Kampert, A. D. Moore, M. D. Higgins, "Impact of Lithium-Ion Battery State of Charge on In Situ QAM-Based Power Line Communication," *Sensors*, vol. 22, no. 16, 6144, Aug. 2022, doi:10.3390/s22166144.

[25] T. A. Vincent, B. Gulsoy, J. E.H. Sansom, J. Marco, "In-situ instrumentation of cells and power line communication data acquisition towards smart cell development," *Journal of Energy Storage*, vol. 50, 104218, June 2022, doi: 10.1016/j.est.2022.104218.

[26] T. S. Vivek, P. Sivraj, "Software Framework for Physical Layer of Home-Plug GreenPhy," In *Proc. 2023 OITS*, Raipur, India, 2023, pp. 975-981, doi: 10.1109/ocit59427.2023.10430705.

[27] HomePlug Green PHY; The Standard For In-Home Smart Grid Powerline Communications. HomePlug Powerline Alliance: Portland, OR, USA, 2010 [Online]. Available: https://content.codico.com/fileadmin/media/download/datasheets/powerline-communication/plc-homeplug-green-phy/homeplug_green_phy_whitepaper.pdf.

[28] S. Park, E. Lee, Y. -H. Noh, D. -H. Choi, J. -g. Yook, "Accurate Modeling of CCS Combo Type 1 Cable and Its Communication Performance Analysis for High-Speed EV-EVSE Charging System," *Energies*, vol. 16, no. 16, 5947, Aug. 2023, doi: 10.3390/en16165947.

[29] Y. M. Chung, "Performance Comparisons of Broadband Power Line Communication Technologies," *Appl. Sci*., vol. 10, no. 9, 3306, May 2020, doi: 10.3390/app10093306.

[30] J. J. Jeon, Y. S. Kim, I. K. Min, S. H. Kim, "Timing Analysis and Structure for HPGP PHY FEC Blocks", in *Proc. ICICPE*, Nha Trang, Vietnam, 2024, pp. 124-125.

[31] J. L. Lakshmi, J. Jayakumari, "A reduced complexity rate-matching and channel interleaver/de-interleaver for 5G NR, " *Engineering Research Express*, vol. 6, no. 2, 025301, Apr. 2024, doi:10.1088/2631-8695/ad37f1.

[32] K. Huo, Z. Hu, D. Liu, "Design and Implementation of Shared Memory for Turbo and LDPC Code Interleaver," *Wireless Communications and Mobile Computing*, Feb. 2022, doi:10.1155/2022/5782199.

[33] S. Talakoub, L. Sabeti, B. Shahrrava, M. Ahmadi, "An Improved Max-Log-MAP Algorithm for Turbo Decoding and Turbo Equalization," *IEEE Transactions on Instrumentation and Measurement*, vol. 56, no. 3, pp. 1058-1063, June 2007, doi: 10.1109/tim.2007.894228.

[34] X. Xu, "Turbo Decoding Theory: Derivation and Performance Evaluation," in *Proc. 2024 4th ACCTCS*, Shenyang, China, 2024, pp. 795-799, doi: 10.1109/acctcs61748.2024.00146.

[35] Y. Gao, M. R. Soleymani, "Spectrally efficient non-binary turbo codes: Beyond DVB-RCS standard," in *Proc. 2002 9th International Symposium on Antenna Technology and Applied Electromagnetics*, St. Hubert, QC, Canada, 2002, pp. 1-1.

[36] H. Balta, R. Lucaciu, A. Mihaescu, "Study on intra-symbol interleaving for Multi-Non-Binary Turbo Codes," in *Proc. 2015 38th International Conference on Telecommunications and Signal Processing*, Prague, Czech Republic, 2015, pp. 1-4, doi:10.1109/tsp.2015.7296401.