ISSN 2581 – 3781

AJCSE
ASIAN JOURNAL OF COMPUTER SCIENCE ENGINEERING

Available Online at www.ajcse.info
Asian Journal of Computer Science Engineering 2021;6(2):35-44

REVIEW ARTICLE

FPGA Hardware Design of Different LDPC Applications: Survey

Arwa H. Ashou, Dhafir A. Alneema

Department of Computer Engineering, Collage of Engineering, University of Mosul, Iraq

Received on: 28-02-2021; Revised on: 15-04-2021; Accepted on: 30-05-2021

ABSTRACT

Low-Density Parity-Check (LDPC) error correction decoders emerge as a suitable path as long as offers a resilient error correction performance and its appropriateness to comparable hardware operation. This paper has been presented a case study to evaluate the use of LDPC code designs based on various features, such as flexibility, high processing speed, and the parallelism of Field-Programmable Gate Array (FPGA) devices. Hence, it has categorized the differences of key factors in FPGA-based LDPC decoder design and three crucial performance features are defined, such as processing throughput, processing latency, and hardware resource requirements. Furthermore, this word supports the concerned researchers to comprehend the differences between various related word and their results of most popular techniques.

Key words: Bit flopping algorithm, Field programmable gate array, Forward error correction, Low-density parity-check, Mini-Sum algorithm, Vivado HLS tool

INTRODUCTION

Communication systems use a medium or channel to transmit data between ends (source and destination). Therefore, the reliability of data on the destination side is relied on outdoor noise which able to affect the signal demonstrating. Hence, the definition of noise is error delivered data, as long as the channel capacity is wider than the data rate that means the transmission is reliable.[1] Accordingly, in 1962, Robert Gallager was a PhD student and suggested a direct block code based on a thin check matrix which defined as a low-density block-check (LDPC) code to provide the structure technique, including the theoretical evidence of its performance and the iterative decoding algorithm.[2,3] However, at that time due to some challenges, such as specifications of computers growth and hardware operation, LDPC codes were not properly comprehended.[4] Besides, Mackay, and Neal discovered that LDPC codes have a major similarity performance.[5] Furthermore, due to the complication of the LDPC encoding and decoding codes with hardware; they were ignored for a long time until Mackay and Neal developed them; as well as it was overlooked for a long time until discovered Turbo codes and then used as a part of LDPC codes. In contrast, LDPC codes involved the researcher's consideration and become a necessary research point.[6] As indicated in Develi and Kabalci, Cocco *et al*.[7,8] that LDPC is maintaining a moral enactment transmits it to have a great family of error-correction codes working in existing data communication systems. Besides, Malema, Develi and Kabalci[1,7] emphasized that LDPC has various advantages as following:

1. Representing improved block error performance
2. Error bases in considerable low bit error rate (BER) values
3. The complication of decoding rises with the size of the blocks
4. Can be utilized in several wireless communication applications and can be used for different channel coding on particular communications, such as WiMAX, Wi-Fi, and ethernet
5. Able to present a great presentation in mobile communication systems.

Address for correspondence:
Arwa H. Ashou,
E-mail: arwahafidh@gmail.com

LDPC is direct codes with a meager equality check medium H. The meagerness of the parity check matrix H means that it holds quite a few 1 s among many 0 s. The sparseness allows LDPC codes to grow the minimum space. Naturally, the minimum distance of LDPC codes linearly grows based on the code-word length.[9] The LDPC codes have been confirmed and extensively used in cellular organizations.[9] However, it can be divided into regular and irregular codes.[10] It can be regular when the number of 1's in each column (wc) is constant for every column and the number of 1's in each row (wr) is constant for all row. The LDPC code by Gallager is regular and constructed by randomly choosing the locations of 1 s with the fixed numbers in each row and column.[9] The regular LDPC code has various features, such as the same number of parity check equations is a part of each coded bit. Nevertheless, it is irregular if wc and wr are not constants.[11] The BER presentation of irregular LDPC codes is better than regular by up to 0.5 Decibel (dB).[12] The BER considers as satisfactory with a very high-speed data rate in current communication. Hence, it has become essential to sustenance the method called Forward Error Correction (FEC) to moderate the high error. LDPC codes are one of the most important FEC codes which are used for the next generation.[13,14]

LDPC CODES REPRESENTATION

A graphical illustration uses to decode LDPC codes (Tanner graph) matrix, less structured, and deliberate with the assets of H as the emphasis in Johnson.[15] Bit, variable, symbol, and components of Tanner's graph. The bit and check nodes individually signify code-word bits and parity equations, the Tanner graph demonstrates in Figure 1. The advantage signifies a construction between bit and checks bulges, so when the bit explains in the consistent parity check equation. Thus, in a Tanner graph, the amount of edges matches the number of 1s in the parity check matrix H. The squares characterize parity check equations and the circles show bit nodes in Figure 1.

Error correction of LDPC codes is determined by the length of the code-word and the distinguishing of the check matrix. Better performances with more code-word were given by the decoder, for example, the large extent of the G matrix and worthy check

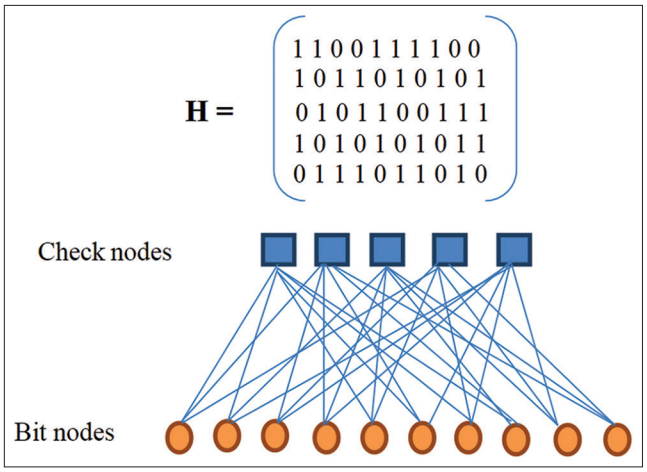

Figure 1: A Tanner graph of H matrix 10 columns and 5 rows[12]

matrix. Therefore, to accomplish improved BER act with LDPC codes nearby channel ability, the length of the LDPC code-word using thousands of bits. In addition, a large minimum distance should be kept by a moral LDPC code domain with no short cycles in its Tanner graph,[11] as long as the smallest direction can be improved by growing the width or the column weight. Furthermore, in the Tanner graph an arrangement of linked nodes beginning and finish at the similar node with no more than one node. The number of edges in a cycle is named series length and the smallest size of the series in a graph signifies the width of the graph.[10]

The LDPC codes have two types as follows: Non-binary and binary LDPC codes. The binary one display-worthy error correction ability and network volume used for huge block lengths. The short code-word lengths exhibit inferior performance due to short cycles in the parity matrix. In contrast, non-binary LDPC codes including non-zero elements of the H matrix are well-defined in Galois area GF (q), while NB-LDPC codes were established to have better performance than binary codes at short and medium code-word lengths.[16] Although higher order is the act at the charge of encoder/hardware complexity,[11] also it presents a good performance using iterative Belief Propagation (BP) at medium code lengths (500contrast, non-binary LDPC codes including non-zero elements oan 16QAM). Figure 2, shows an example of an evaluation of a binary LDPC code and a non-binary LDPC code.[9]

LDPC code has been approved as the error-correcting code in DVB-S2, 3GPP 5G New Radio, and Satellite Digital Television Broadcasting Standard

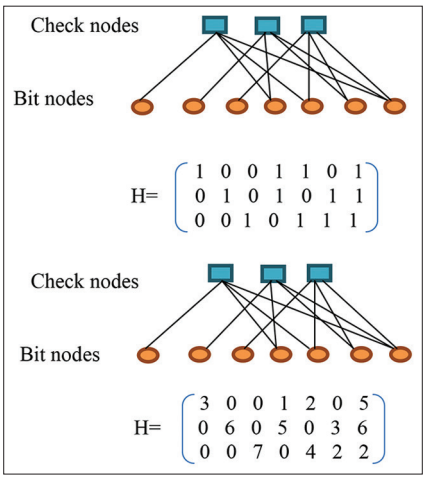

Figure 2: Tanner graph of binary and non-binary LDPC code[3]

(second generation),[17] as well as IEEE 802.11n WLAN and IEEE 802.16e WiMAX,[11] Equally, it is used in Ethernet, such as 10GBase-T Ethernet, and different standards too.[18,19] The LDPC codes have two types of algorithms to pass messages, such as hard decision decoding algorithm such as Bit Flipping (BF) algorithm and soft-decision decoding algorithm like BP algorithm.[11] BF algorithms created for supporting a simple decoder implementation. Hence, a hard decision decoding performs less than soft-decision decoders.[4]

An error can be happened by the channel noise, multi-path disappearing channel, or various sources; so interference can be described on some data bits. In other words, transmission in the noisy communication channel can accuse random and burst errors. As a result, bad quality transmission can lead to the BER.[11] However, enhancing throughput or the network performance to reduce BER can be supported and done using the error detection and correction techniques.[9,11] Furthermore, channel coding has been used to advance the show of wireless digital communication applications, and wireless network at higher data rates. For that reason Alabady,[11] mentioned that there is a necessity to investigate in the coding area to plan and design codes for channels with specific power and bandwidth. In practical implementation, these codes provide a high degree of parallelism. The complexity and throughput of an LDPC decoder depend on many parameters such as block length, code rate, processing node complexity, interconnection complexity, no of iterations and parallelism level. Accordingly, there is a sense of balance between the performance of the decoder and the difficulty of decoding. The parallelism in decoding provides important throughput if correctly used. Furthermore, the flexibility in FPGA is appropriate for designing LDPC decoder rather than in general-purpose processor.[20]

RELATED WORK

Many types of research were supposed and implemented LDPC codes through FPGA based on different features, such as FPGA device, used algorithm, code length, throughput, iteration, BRAM, registers FF, LUT, max frequency, and slices (Xilinx), as shown in Table 1 and the following: Sun *et al*.[21] presented that efficient FPGA implementation of LDPC codes was done. However, the greatest challenge was the generation of VHDL code, as long as it requires a long time to write and contract the system. In the same way, Kim *et al*.[22] displayed that an incompletely parallel LDPC decoder construction is made on a Xilinx Virtex-5 FPGA for a rate-1/2 code of size 648 bits. XC6VLX240T, also FPGAs were used to evaluate the complication of the suggested design by Loi.[23] In contrast, using LDPC decoder architecture for the Sum-Product algorithm (SPA) with a code word length of 2048 presents a low FER area and offers indicators of the error traces. As well, the implementation was done using Virtex-II Pro XC2VP70 series and Xilinx FPGA to achieve the following: FER resolution of 10–8 within 1 h, a throughput of 240 Mb/s on the FPGA, and a throughput of 260 kb/s on an Intel Xeon 2.4 GHz processor. Equally, a performance evaluation of MS algorithm for a high parallel FPGA-based test bed for LDPC code with the code word length equal to 4923 and Eight Xilinx Virtex-2 Pro FPGAs was achieved to get the first investigation for the error floor of LDPC codes at FER resolution of 10–12 in magnetic recording channel. The results illustrate a FER of 10–10 in <6 h and a FER of 10–12 in 24 days; so the designers emphasized that a scale of 10–12 had taken 10 years in software simulation.[24] Unlike, Mhaske[25] proposed procedures to do a high-throughput presentation for an MSA-based decoder for QC-LDPC codes. Hence, the IEEE 802.11n decoder was applied to gain throughput of 608Mb/s (at 260MHz) and latency of 5.7_s using Xilinx Kintex-7 FPGA. Thus, the reduced time was shown using FPGA IP with a complex signal processing algorithm.

Table 1: Different specifications and results obtained from various researches

Ref.	FPGA Device	Algorithm	Code length	Throughput	Iteration	BRAM	FF	LUT	Max.Freq.
[21]	XilinxVirtex 4	MS (non-binary)	486×972	50 Mb/s	20				131 MHz
[58]	Virtex-2 Pro	SP	2048	240 Mb/s					
[24]	Virtex-2 Pro	MS	4923	NA					
[35]	Virtex-5/T-2	MS	3369	332 Mb/s					
[33]	Xilinx V4	QCI	(3969, 3213)	1.474 Gb/s	15	330		98,003	195.7 MHz
[33]	Xilinx Virtex 2	MS	(768, 1536)	114 Mb/s	3				162 MHz
[33]	Xilinx Virtex 4	Normalized MS	(756, 3969)	82.4 Mb/s	15				200 MHz
[33]	XilinxVirtex 4	Normalized MS	(1022, 8176)	625 Mb/s	15				212 MHz
[37]	Xilinx Vertex 5	SMP	648	16.2 Gb/s	3.8		5963	14239	188 MHz
[37]	XilinxVirtex 4	MS (modified)	(600, 1200)	8.3 Gb/s	8.9				123 MHz
[37]	XilinxVirtex 5	MS (modified)	(324, 648)	4.33 Gb/s	8.4				113 MHz
[22]	Xilinx V5	Partially parallel	(648,324)	110 Mb/s	8	24		19,761	100 MHz
[23]	XC6VLX240T					31%	17%	60%	214.5 MHz
[38]	Virtex5	SPA	(648,324) regular				12443	58787	
[38]	Virtex5	SMPA	(648,324) regular				5963	14239	
[38]	Virtex5	BFA	(648,324) regular				2069	3577	
[39]	Virtex-5 FPGA	SP	2304	344 Mb/s					
[59]	Virtex-5	belief propagation	576	380 Mb/s					
[59]	Xilinx Virtex 5	MS with correction	336 x672	475 Mb/s					100 MHz
[41]	Xilinx Virtex 5	SP (modified)	1022 x8176	522 M b/s					228 MHz
[41]	Xilinx Virtex 5	SP (modified)	1022 ×8176	522 Mb/s	-				228 MHz
[60]	Xilinx V5	(3,6) Regular MMS	(1152, 576)	11.7 Gb/s	10	-		39,024	
[61]	Altera StratixIV EP4SGX230	PN- LDPC-CC (3, 6)		2.4 Gb/s	9	-		-	75 MHz
[42]	Spartan-6	648		27Mb/s	3	56-9k		24	
[51]	Xilinx V6	GF(q) DPC Decoding	(480, 240)	6 Mb/s	10	26		10,916	180 MHz
Ref.	**FPGA Device**	**Algorithm**	**Code length**	**Throughput**	**Iteration**	**BRAM**	**FF**	**LUT**	**Max.Freq.**
[62]	Xilinx Virtex-5	BP Decoding Schedule	64800, 32400	540 Mb/s	10				150 MHz
[46]	Altera Stratix 5 D5	BP Decoding Schedule	1152, 576, 48	103.9 Mb/s	10	75.3	42.3	42.9	222.6 MHz
[46]	Stratix-5		768	99 Mb/s	10	1,641-20k			
[46]	Stratix-5		1,152	104 Mb/s	10	1,703-20k			
[46]	Stratix-5		1,920	81 Mb/s	10	1,516-20k			
[63]	Altera StratixIV EP4SGX230	Digit-online LDPC	(576, 432)	740 Mb/s	10	-	-		90 MHz
[28]	Virtex6	MSC	(756,252) regular				1447	1977	
[47]	Altera Stratix 5 D5	BP Decoding Schedule	1944, 972, 81	21 Mb/s	10	67	36	41	157 MHz
[47]	Stratix-5		1,944	21	10	1,349-20K			
[29]	Xilinx,Vertex 4	Stochastic	1024	353 Mb/s	NA		-	-	212 MHz
[44]	Virtex 5		2304	1096 Mb/s	8	232			114 MHz
[25]	Kintex-7k410t			337 Mb/s		4.7	9.1	8.7	
[25]	Kintex-7k410t			608 Mb/s		6.4	5.3	8.2	
[48]	Artix-7		816	3	10	32-18K		9,087	

(*Contd...*)

Table 1: (*Continued*)

Ref.	FPGA Device	Algorithm	Code length	Throughput	Iteration	BRAM	FF	LUT	Max.Freq.
[54]	Altera Arria10	MS	128	28.02 Gb/s	4		16519	33,016	218.91 MHz
[54]	Altera, Arria10	MS	128	28.02 Gb/s	4		16519	33,016	218.91 MHz
[54]	Xilinx, Vertex US	MS	128	35.26 Gb/s	4		16068	43067	275.48MHz
[54]	Xilinx Vertex US	MS	256	65.67 Gb/s	4		31995	86809	256.54 MHz
[64]	Artix-7		816	50 Mb/s	5	28-8K		5,720	
[64]	Stratix-5		1,944	134 Mb/s	5	63-20K			
[64]	Spartan-7		648	141 Mb/s	3	22-18K		4,143	
[64]	Stratix-5		768	62 Mb/s	5	25-20k		12	
[26]	Xilinx Kintex-7 K410T	BP Decoding	(1944, 972, 81)	608 Mb/s	4	6.4	5.3	8.2	200 MHz
[65]	XC6VLX240T					30.5%	19.5%	47%	225MHz
[65]	ZINC XC7Z030					25%	18.5%	54%	238.5MHz

FF: Flip flop, MS: Mini-Sum algorithm, SP: Sum-Product, QCI: Quasi-Cyclic irregular, SMP: Simplified message passing, LUT: Lookup tables, BF: Bit flipping algorithm

The Mhaske *et al*.[26] suggested strategies to accomplish a high-throughput FPGA architecture and high-level synthesis (HLS) compilation for Quasi-Cyclic LDPC codes using circulant-1 uniqueness matrix structure. This research constrained on splitting the node processing in the MS algorithm based pipelining with no need to employing extra hardware resources. In addition, Xilinx Kintex-7 FPGA with the LabVIEW FPGA Compiler in the LabVIEW Communication System Design Suite was used to validate this architecture and using IEEE 802.11n decoder. The Mosleh *et al*.[6] designed and implemented an effective project for log domain decoder using Xilinx system with FPGA to provide a BER outcome same to theory controls. Furthermore, proposed BF algorithm decoding using VHDL simulation to reduce the number of iterations. In addition, the LDPC decoder has been implemented with an FPGA device using the Xilinx system to display efficient design. This research engaged the services of OFDM to run many techniques with stating the bandwidth consumption. Besides, the development of LDPC codes using BF algorithm and Chaos Shift Keying (DCSK) communication system to be applied on Xilinx Spartan-6 FPGA development kit using Xilinx SG tools. The effects demonstrate that the coding obtains 3 dB at a BER of 10-4 compared to the DCSK without LDPC code.[13] Besides, Hasan *et al*.[27] designed a system based LDPC codes, Xilinx system generator, Vivado tool, and FPGA device Kintex7 (XC7K325T-2FFG900C) for evaluating the performance of BER, complexity and the exhibits time. In this research, the outcomes displayed that the raising of the Noise Ratio (SNR) can improve BER value significantly.

In addition, Dias[28] described that the proposes of using the HLS technique and Xilinx Virtex 6 FPGA to apply LDPC decoder on hardware was accomplished. This project presented the stages exploited to synthesize hardware employed the nearby the possibility of the interactive model of the system. Similarly, an implementation of 1024, 512 based on a fully parallel LDPC decoder was displayed. The claim of increasing Xilinx up to 709 Mbps as long as the throughput of 650 Mbps was 61 MHz for Stratix IV device. Thus, the result presented the number of iterations up to 32 which affects the latency to be double.[29] Equally, Devrari *et al*.[30] implemented LDPC decoder using Shift-Register to reduce the complexity and performance analysis. The modified sum-product (MSP) method was tested to decode the signal. Using hardware chip design, such as Vivado 17.4, programmed, VHDL is assessed on Virtex-5 FPGA and timing parameters with FPGA. Furthermore, the Chinese Digital Television Terrestrial Broadcasting standard was proposed using FPGA based QC-LDPC decoder and the soft-decision Mini-Sum (MS) algorithm.[31] Besides, Ji *et al*.[32] describes a parallel GPU operation using the soft-decision MS decoder for QC-LDPC codes to target the WiMax and WiFi standards. However, the original QC-LDPC (n;k) codes of the (n;k) pair was equal to

1944; 972 for WiFi and 2304; 1152 for WiMax, as well as takes tens of times that smaller than the ones active in QC-LDPC cryptosystems. In addition, Chen *et al*.[33] highlighted that a methodology of the incompletely parallel decoders using a few variable numbers and check node processors to prepare some processing in parallel has been given lower throughputs and takes much fewer hardware resources like wire routing. In contrast, PW Group[34] presents non-binary decoders for different order implementations. Moreover, Ji *et al*.[24] demonstrated that the test bed created based on FPGA and the investigation of different levels of parallelism on resource procedure and throughput.

The, Cai *et al*.[35] suggesting a scalable FPGA-based vector decoder implementation for the MS algorithm on the LDPC codes with the code word length up to 3369. It applied the data on the subsequent iteration into embedded memory blocks to reduce LUT-RAM usage and accomplish a maximum amount of 332 Mb/s. Furthermore, Bonello *et al*.[36] deliberated that an overview of the essentials LDPC codes was shown. Hence, different features of LDPC codes indicated briefly, such as desirable properties of encoding and decoding along with its logical effects. Not only that but also, two optimization methods, such as factorization and folding, were described to offer actual deployment of block RAM resources according to the block RAMs which existing in FPGAs and dual-ported with very fast access times.[33] Too, the Chandrasetty and Aziz[37] modified design of decoders to reduce the complication of decoding LDPC codes based on FPGA was supposed. Next, Cocco *et al*.[8] expounded a modified 2-bit MS algorithm based on LDPC decoding to decrease the density of the decoder with a minor drop in the BER performance. The[38] approach using a Reduced Complexity Message Passing algorithm for large LDPC codes was executed. The major impression of this approach was to link the adeptness of the SPA based BF algorithm. Thus, the decoder was advanced through Verilog and the developed BF algorithm to accomplish the reduction of average number decoding and compares it with the BF algorithm too. As a result, the decoder needs fewer hardware resources compared to the SPA. The income has given 16.2 Gbps at 188 MHz using a basic message passing algorithm. However, this approach has not shared the hardware resource consumptions as long as it spends 3.8 cycles on average to correct a code word. In addition, design and tested a backtracking scheme for detecting error patterns at FER based low as 10–10 is offered. Hence, a test bed of the SPA for the LDPC code based on FPGA with the code word lengths of 1056, 1944, and 2304 was presented, thus a throughput of 344 Mb/s is a code word length of 2304.[39] Besides, Balatsoukas-Stimming and Dollas[40] demonstrated that a code length of 1000 and 1152 was used to implement a fast fully parallel FPGA-based design of LDPC codes. Accordingly, specific FPGA optimizations for decreasing the lookup tables (LUTs) were considered. The final result was high; nevertheless, the architecture requires flexibility. A modified design to reduce the difficulty of decoding LDPC codes based on FPGA was done.[41]

The[42] examination of Vivado HLS tools using LDPC codes to find out the effectiveness was achieved. In this research, a discussion of the mapping over the hardware with the demonstration of the ability for implementing non-trivial designs was shown. Furthermore, Andrade *et al*.[43] determined that implementation of LDPC decoders with HLS was offered. Hence, a design and application of HLS architectures with RTL levels of the act were made and presented a productive result with higher logic consumption. In contrast, several hardware executions for LDPC codes have designed using limited parallel architectures and memory-shared, such as 2304 bit, and 1/2-rate LDPC code using 232 memory blocks was designed. The architectures were affecting by the specific size of memory, bandwidth consumption, weak memory usage, and interconnection complication which unusual in current multiplexers. Consequently, the income displayed a difficulty using pipeline and is not movable or synthesizable on ASIC implements.

In other words, the result showed that throughput has hundreds of megabits-per-second (Mbps).[44] Another work the, Andrade *et al*.[45] utilized programmable LDPC decoders to write a regular survey that supported by the model-based design and HLS including hardware faster LDPC decoders applications as indicated in Scheiber *et al*., Andrade *et al*., Andrade *et al*., Roh *et al*.[42,46-48] Therefore, this survey focuses on the project issues, challenges and practices based on the

reconfigurable programmable LDPC decoders. Besides, Andrade *et al*.[49] compared three various HLS tools to demonstrate performance within the similar instruction of scale as physically enhanced design. Moreover, El Haroussi and Abdelmounim[50] designed and implemented an FPGA of parallel architecture for low difficulty and LDPC decoder. Furthermore, VHDL design and Believe Propagation (BP) algorithm basic MS algorithm including a co-simulation on Simulink platform in BER were considered. Moreover, Sulek *et al*.[51] illustrated that higher-order Galois-field (GF) (non-binary) and LDPC decoders on FPGAs were explained in the literature. Related to GF (q=2p) codes, the decoding difficulty raises with 2pdc as the greatest nonzero objects in the parity check matrix row. Hence, non-binary decoders confirm the additional accessible FPGA resources such as formation logic blocks, block RAMs, and multipliers. Furthermore, using LDPC decoder based FPGA to reduce the difficulty of obtaining a high data rate was done.[52] Equally, a studying of parallel and digit-online block LDPC decoder test on FPG for WiMAX 576-bit, rate-3/4 codes, and power capacities through DE4 board was achieved. Including, an effort for changing LLR precision on the max clock frequency and logic usage, and power iteration for a 6-bit LLR decoder was given.[28]

Not only that but, in[53] a discussion of many implementations using FPGA of LDPC was illustrated. The research deliberates the survey in various limitations, such as LDPC code, algorithm, architecture, number of iterations, characteristics of the applied hardware liked flexibility, bandwidth consumption, spread energy productivity, processing energy effectiveness, hardware necessities, latency, and output. In the same way, a performance evaluation of different FPGA based on LDPC decoders, characteristics, approximately 140 FPGA based LDPC decoders using different technical parameters was reported. Finally, this research presented that is was failed to highlight the significance of LDPC decoding algorithms and the role of decoder design.[53]

Raheem[54] designed an FPGA based architecture of LDPC decoder considering the high parallelism, flexibility and computation speed of the FPGAs, and parameterized to approve any difference of the LDPC decoder. The main idea is running input, low latency, restrained resource, and actual high throughput of 65Gbps using pipelining. Accordingly, the throughput of FPGA was 16.2 Gbps.

An overview of cellular communication systems based on three types of decoders, such as LDPC, Polar, and Turbo codes ASIC, was implemented.[55] Next, the project of production particular message bits using row and column structures, channel (AWGN), Variable Node Unit, Check Node Unit, MS algorithm, AWGN channel, and LDPC decoder decodes to the encoded message was demonstrated. As a result, the effective structure has considered reducing the complication and receiving a smaller amount of resources. As well as, used multiplexed storage construction for getting minimum FPGA resources and keeping node message.[56] In, de Souza and Nazar[57] validated that the performance of LDPC decoders on single error for destroying decoding. Is categorizing the maximum number of mechanisms to reduce action by 89 % though covering 55% of their area. Hence, opposite to what has noticed in ASICs, such as a single SEU that can present different degradation in FPGA-based LDPC decoders and therefore cannot be disregarded in systems. Practically, using this method offers diverse design selections with dissimilar area and BER.

Next,[6] LDPC codes based on Digital Video Broadcasting - Satellite - Second Generation (DVB-S2) was implemented. Hence, using LDPC codes enables the construction of DVB-S2 to permit 360 functional units to apply at the same time. Thus, a novel work of Range Addressable Looks Up Table (RALUT) to simplifies the LDPC decoding algorithm was achieved. Usually, RALUTs are regularly spread on input instead of demonstrating the LUT input regularly. Thus, the non-uniform scale used to assign values near zero. Not only that but also, Zynq XC7Z030 was used to evaluate the complexity of the designed system. Therefore, the result presents increasing speed due to the LUT method which requests more memory. Table 1, shows the results of different researches using LDPC codes based FPGA devices. l paragraphs must be indented.

CONCLUSION

In this research, diverse types of potential techniques in LDPC codes have been comprehensively reviewed and compared their

limitations that were involved LDPC based FPGA implementations Thus, presented fundamentals in the design of LDPC codes using different algorithms, hardware, techniques, etc. In this research, much-related work was reviewed and specified based on using LDPC codes, Bit Flopping (BF) algorithm, MS algorithm, Vivado HLS tool, FPGA, SPA, BER and FER performance, VHDL, MSP method, Kintex7, Xilinx, Stratix, ZINC, etc. Therefore, different elements were evaluated and analyzed, such as FF, MS, SP, QCI, SMP, LUT, BRAM, Max-Freq., and BF. In other words, this work illustrated the usability of representing codes by the table and how this naturally leads to Quasi Cyclic LDPC codes, next which the current technique was comprehensive to the superior case-based LDPC codes. In other words, a discussion of different design approaches for LDPC codes with their results was demonstrated.

REFERENCES

1. Malema GA; School of Electrical and Electronic Engineering. Low-Density Parity-Check Codes: Construction and Implementation. School of Electrical and Electronic Engineering; 2007.
2. Gallager R. Low-density parity-check codes. Massachusetts: Institute of Technology; 1962.
3. Kuc M, Sułek W, Kania D. Low power QC-LDPC decoder based on token ring architecture. Energies 2020;23:6310.
4. Zoni D, Galimberti A, Fornaciari W. Efficient and scalable FPGA-oriented design of QC-LDPC bit-flipping decoders for post-quantum cryptography. IEEE Access 2020;8:163419-33.
5. Fang P. Application analysis of low-density parity check code for wireless network. IOP Conf Ser Earth Environ Sci 2021;1:012043.
6. Mosleh MF, Hasan FS, Azeez RM. Design and implementation of log domain decoder. Int J Elect Comput Eng 2020;2:1454-68.
7. Develi I, Kabalci Y. A comparative simulation study on the performance of LDPC coded communication systems over Weibull fading channels. J Appl Res Technol 2016;2:101-7.
8. Cocco M, Dielissen J, Heijligers M, Hekstra A, Huisken J. A Scalable Architecture for LDPC Decoding. Vol. 1. Proceedings-Design, Automation and Test in Europe; 2004. p. 88-93.
9. Kim H. Design and Optimization for 5G Wireless Communications. New York: John Wiley and Sons; 2020.
10. Rao KD. Channel Coding Techniques for Wireless Communications. Berlin: Springer; 2015.
11. Alabady SA. Binary and nonnications for 5G Wireless CommunicationsA survey. Int J Inf Eng Appl 2018;1:104-17.
12. Jiang Y. A Practical Guide to Error-control Coding Using MATLAB. Artech House; 2010.
13. Mahmood FS, Mosleh F, Abdulhameed AH. FPGA hardware co-simulation of DCSK based on bit-flipping LDPC decoding using xilinx system generator. Int J Adv Sci Technol 2020;29:18.
14. Ghaffari F, Unal B, Akoglu A, Le K, Declercq D, Vasić B. Efficient FPGA Implementation of Probabilistic Gallager B LDPC Decoder. In: 2017 24th IEEE International Conference on Electronics, Circuits and Systems (ICECS); 2017. p. 178-81.
15. Johnson SJ. Iterative Error Correction: Turbo, Low-density Parity-check and Repeat-accumulate Codes. Cambridge, United Kingdom: Cambridge University Press; 2010.
16. Noorshams N, Wainwright MJ. Stochastic belief propagation: A low-complexity alternative to the sum-product algorithm. IEEE Trans Inf Theory 2012;59:1981-2000.
17. Parkvall S, Dahlman E, Furuskar A, Frenne M. NR: The new 5G radio access technology. IEEE Commun Standards Mag 2017;1:24-30.
18. Chiueh TD, Tsai PY, Wei LI, Chiueh TD. Baseband Receiver Design for Wireless MIMO-OFDM Communications. Hoboken, New Jersey: Wiley Online Library; 2012.
19. Kong W. Low Density Parity Check Encoder and Decoder on SiLago Coarse Grain Reconfigurable Architecture. Sweden: Degree Project in Electrical Engineering; 2019.
20. Raju P, Prasad PS. Design of an LDPC decoder and its performance. Int J Elect Electron Commun 2017;1:012033.
21. Sun Y, Zhang Y, Hu J, Zhang Z. FPGA Implementation of Nonbinary Quasi-cyclic LDPC Decoder Based on EMS Algorithm. In: 2009 International Conference on Communications, Circuits and Systems; 2009. p. 1061-5.
22. Kim MH, Park TD, Kim CS, Jung W. An FPGA Design of Low Power LDPC Decoder for High-speed Wireless LAN. In: 2010 IEEE 12th International Conference on Communication Technology; 2010. p. 1460-3.
23. Loi KC. Field-programmable Gate-array (FPGA) Implementation of Low-density Parity-Check (LDPC) Decoder in Digital Video Broadcasting-Second Generation Satellite (DVB-S2). University of Saskatchewan; 2010.
24. Cai Y, Jeon S, Mai K, Kumar BV. Highly parallel FPGA emulation for LDPC error floor characterization in perpendicular magnetic recording channel. IEEE Trans Mag 2009;45:3761-4.
25. Mhaske S. High-throughput FPGA QC-LDPC Decoder Architecture for 5G Wireless. New Jersey: Rutgers University-Graduate School-New Brunswick; 2015.
26. Mhaske S, Kee H, Ly T, Aziz A, Spasojevic P. FPGA-based channel coding architectures for 5G wireless using high-level synthesis. Int J Reconfigurable Comput 2017;2017:3689308.
27. Hasan F, Azeez R, Mosleh M. Designing LDPC system using prob domain decoder in Xilinx system generator. MS and E 2020;745:012033.
28. Dias AF. High Level Synthesis of a Min-sum C LDPC

Decoder; 2014.
29. Quraishi Q. Area efficient FPGA based LDPC decoder using stochastic decoding scheme. Int J Sci Eng Res 2015;3:564.
30. Devrari A, Kumar A, Chauhan H, Kumar A. Design and FPGA implementation of LDPC decoder chip for communication system using VHDL. Int J Recent Technol Eng 2019;8:200-6.
31. Jiang N, Peng K, Song J, Pan C, Yang Z. High-throughput QC-LDPC decoders. IEEE Trans Broadcast 2009;55:251-9.
32. Ji H, Cho J, Sung W. Massively Parallel Implementation of Cyclic LDPC Codes on a General Purpose Graphics Processing Unit. In: 2009 IEEE Workshop on Signal Processing Systems; 2009. p. 285-90.
33. Chen X, Kang J, Lin S, Akella V. Memory System Optimization for FPGA-based Implementation of Quasi-cyclic LDPC Codes Decoders. Vol. 58. IEEE Transactions on Circuits and Systems I: Regular Papers; 2010. p. 98-111.
34. PW Group. IEEE Standard for Information Technology-Local and Metropolitan Area Networks-Specific Requirements-part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment 6: Wireless Access in Vehicular Environments. IEEE Std; 2010. p. 802.
35. Chen X, Kang J, Lin S, Akella V. Accelerating FPGA-based Emulation of Quasi-cyclic LDPC Codes with Vector Processing. In: 2009 Design, Automation and Test in Europe Conference and Exhibition; 2009. p. 1530-5.
36. Bonello N, Chen S, Hanzo L. Low-density parity-check codes and their rateless relatives. IEEE Commun Surv Tutorials 2010;13:3-26.
37. Chandrasetty V, Aziz SM. FPGA implementation of high performance LDPC decoder using modified 2-bit min-sum algorithm. In: 2010 2nd International Conference on Computer Research and Development; 2010. p. 881-5.
38. Chandrasetty V, Aziz SM. FPGA implementation of a LDPC decoder using a reduced complexity message passing algorithm. J Netw 2011;6:36.
39. Chen X, Kang J, Lin S, Akella V. Hardware Implementation of a Backtracking-based Reconfigurable Decoder for Lowering the Error Floor of Quasi-cyclic LDPC Codes. Vol. 58. IEEE Transactions on Circuits and Systems I: Regular Papers; 2011. p. 2931-43.
40. Balatsoukas-Stimming A, Dollas A. FPGA-based Design and Implementation of a Multi-GBPS LDPC Decoder. In: 22nd International Conference on Field Programmable Logic and Applications (FPL); 2012. p. 262-9.
41. Khati SS, Bisht P, Pujari SC. Improved Decoder Design for LDPC Codes Based on Selective Node Processing. In: 2012 World Congress on Information and Communication Technologies; 2012. p. 413-8.
42. Scheiber E, Bruck GH, Jung P. Implementation of an LDPC Decoder for IEEE 802.11 n Using Vivado High-level Synthesis. In: Proceeding IEEE ICECS; 2013. p. 45-8.
43. Andrade J, George N, Karras K, Novo D, Silva V, Ienne P, *et al*. From Low-architectural Expertise up to High-throughput Non-binary LDPC Decoders: Optimization Guidelines Using High-level Synthesis. In: 2015 25th International Conference on Field Programmable Logic and Applications (FPL); 2015. p. 1-8.
44. Chandrasetty A, Aziz SM. Resource efficient LDPC decoders for multimedia communication. Integration 2015;48:213-20.
45. Andrade J, Falcao G, Silva V, Sousa L. A survey on programmable LDPC decoders. IEEE Access 2016;4:6704-18.
46. Andrade J, Falcao G, Silva V. Flexible design of wide-pipeline-based WiMAX QC-LDPC decoder architectures on FPGAs using high-level synthesis. Electron Lett 2014;50:839-40.
47. Andrade J, Pratas F, Falcao G, Silva V, Sousa L. Combining Flexibility with Low Power: Dataflow and Wide-pipeline LDPC Decoding Engines in the Gbit/s era. In: 2014 IEEE 25th International Conference on Application-Specific Systems, Architectures and Processors; 2014. p. 264-9.
48. Roh SD, Cho K, Chung KS. Implementation of an LDPC Decoder on a Heterogeneous FPGA-CPU Platform Using SDSoC. In: 2016 IEEE Region 10 Conference (TENCON); 2016. p. 2555-8.
49. Andrade J, George N, Karras K, Novo D, Pratas F, Sousa L, *et al*. Design space exploration of LDPC decoders using high-level synthesis. IEEE Access 2017;5:14600-15.
50. El Haroussi BM, Abdelmounim E. VHDL design and FPGA implementation of LDPC decoder for high data rate. Int J Adv Comput Sci Appl 2017;8:435.
51. Sulek W, Kucharczyk M, Dziwoki G. GF (q) LDPC Decoder Design for FPGA Implementation. In: 2013 IEEE 10th Consumer Communications and Networking Conference (CCNC); 2013. p. 460-5.
52. Porcello JC. Designing and Implementing Low Density Parity Check (ldpc) Decoders Using Fpgas. In: 2014 IEEE Aerospace Conference; 2014. p. 1-7.
53. Hailes P, Xu L, Maunder RG, Al-Hashimi BM, Hanzo L. A survey of FPGA-based LDPC decoders. IEEE Commun Surv Tutorials 2015;18:1098-122.
54. Raheem M. High Throughput, Fully Parallel, Pipelined FPGA Implementation of LDPC Decoder. School of Science and Engineering, Karachi Institute of Economics and Technology; 2016.
55. Shao S, Hailes P, Wang TY, Wu JY, Maunder RG, Al-Hashimi BM, *et al*. Survey of turbo, LDPC, and polar decoder ASIC implementations. IEEE Commun Sur Tutorials 2019;21:2309-33.
56. Pawankar S, Mohota N. High Performance LDPC Decoder Design Using FPGA. In: 2019 9th International Conference on Emerging Trends in Engineering and Technology-Signal and Information Processing (ICETET-SIP-19); 2019. p. 1-4.
57. de Souza EN, Nazar GL. Cost-effective Resilient FPGA-based LDPC Decoder Architecture. In: 2019 IEEE 25th International Symposium on On-Line Testing and Robust System Design (IOLTS); 2019. p. 84-9.
58. Zhang Z, Dolecek L,Nikolic B, Anantharam V, Wainwright MJ. Design of LDPC decoders for improved

low error rate performance: quantization and algorithm choices. IEEE Trans Commun 2009;57:3258-68.
59. Li H, Park YS, Zhang Z. Reconfigurable Architecture and Automated Design Flow for Rapid FPGA-based LDPC Code Emulation. In: Proceedings of the ACM/SIGDA International Symposium on Field Programmable Gate Arrays; 2012. p. 167-70.
60. Chandrasetty VA, Aziz SM. An area efficient LDPC decoder using a reduced complexity min-sum algorithm. Integration 2012;45:141-8.
61. Li SY, Brandon TL, Elliott DG, Gaudet VC. Power Characterization of a Gbit/s fpga Convolutional Ldpc Decoder. In: 2012 IEEE Workshop on Signal Processing Systems; 2012. p. 294-9.
62. Pratas F, Andrade J, Falcao G, Silva V, Sousa L. Open the Gates: Using High-level Synthesis towards programmable LDPC decoders on FPGAs. In: 2013 IEEE Global Conference on Signal and Information Processing; 2013. p. 1274-7.
63. Singh M. Power Characterization of a Digit-Online FPGA Implementation of a Low-Density Parity-Check Decoder for WiMAX Applications. Waterloo, Canada: University of Waterloo; 2014.
64. Delomier Y, Le Gal B, Crenne J, Jego C. Model-based Design of Efficient LDPC Decoder Architectures. In: 2018 IEEE 10th International Symposium on Turbo Codes and Iterative Information Processing (ISTC); 2018. p. 1-5.
65. Yasoubi M. An Efficient Hardware Implementation of LDPC Decoder. Canada: Concordia University; 2020.