2019 32nd International Conference on VLSI Design and 2019 18th International Conference on Embedded Systems (VLSID)

High-Throughput and High-Speed Polar-Decoder VLSI-Architecture for 5G New Radio

Rahul Shrestha, *Member*, *IEEE*, Pooja Bansal, and Srikant Srinivasan, *Member*, *IEEE*
School of Computing & Electrical Engineering
Indian Institute of Technology (IIT) - Mandi
Himachal Pradesh - 175005, India
Email addresses: poojabansal2412@gmail.com; rahul_shrestha@iitmandi.ac.in; srikant@iitmandi.ac.in

***Abstract*—In this paper, we propose a new technique for computing logarithmic-likelihood-ratio (LLR) messages in the processing element (PE) unit of belief-propagation polar decoder that is based on two's complement representation of LLR values. In addition, a new PE-architecture corresponding to this technique has been presented that consumes lesser hardware and has shorter critical-path delay resulting in higher clock-frequency as well as throughput. We have incorporated these PE units in the design of single-column based unidirectional belief-propagation polar-decoder using the round-trip scheduling for decoding the polar code of 1024 code-length and 1/2 code rate. Performance analysis of such decoding algorithm in AWGN channel environment has been carried out and it delivered an adequate bit-error-rate of 10^{-4} at 4.2 dB of signal-to-noise ratio. VLSI architecture of the suggested polar-decoder is ASIC synthesized and post-layout simulated in 90 nm and 65 nm CMOS processes using industry standard EDA tools. At 65 nm-CMOS node, our design has achieved a throughput of 11.9 Gbps and a maximum clock frequency of 1.164 GHz. In comparison with the state-of-the-art implementations, our design delivers 10% and 46% better throughput and clock frequency respectively.**

***Keywords*-Forward error-correction (FEC) codes; polar codes; channel polarization; digital communication; digital architectures; very-large scale-integration (VLSI) design; application-specific integrated-circuits (ASIC), complementary metal-oxide semiconductor (CMOS), electronics design & automation (EDA) tools.**

I. Introduction

Error correction encoding and decoding are key processes performed in the transceivers of wireless communication devices for reliable information exchange in cellular network. Some of the contemporary capacity-achieving codes are turbo, low-density parity-check (LDPC) and polar codes [2], [1]. In the 4G technology, turbo code has been incorporated as a channel code. However, complexity of turbo decoding algorithm is of prime concern as it enhances the power consumption as well as silicon area requirement of 4G transceivers. In the year 2009, Arikan proposed a new coding scheme called polar code [1] which is based on the principle of channel polarization. Since its inception, polar code has been widely accepted by the research communities from academia and industry for its benchmark capacity-achieving performance with alleviated decoding complexity. In the year 2018, 3GPP has finished standardization of Phase-1 (Release-16) 5G technology [3] in which LDPC and polar codes have replaced turbo code in the physical layer specification. Therefore, it is high-time for translational research to develop efficient polar-decoder designs for the "new radio" of 5G communication standard.

Various polar-code construction algorithms are reported in literature and their detailed study with performance comparison has been presented in [12]. The complexity of polar decoding algorithm is $O(N \cdot \log N)$, where N is code length, and is based on two popular techniques: successive-cancellation [1] and belief-propagation decoding techniques [13]. Successive-cancellation decoding is computationally less complex than belief-propagation decoding. However, the former being serial in nature suffers from huge latency problem which reduces the decoder throughput. Therefore, a highly parallel decoding algorithm has been conceived in [13] and is referred to as belief-propagation decoding algorithm. The first of its kind, proposed by Pamuk [13], achieved a throughput of 2.8 Mbps operating at a maximum clock frequency of 160 MHz and number of iterations has been adequately adjusted to achieve error-correction performance in par with successive-cancellation decoding algorithm. Several approaches [7]-[11] have been proposed to reduce the latency and enhance the throughput. In addition, critical-path reduction technique has been reported in [7] by optimizing the architecture of processing element (PE). In this work, we propose a new technique for the PE computations in polar decoder based on 2's complement representation. A new PE architecture has been presented corresponding to this technique and our design has achieved shorter critical path as well as lower hardware complexity in comparison with the reported works. Eventually, we have incorporated our PE design in an overall architecture of single-column polar-decoder. Performance analysis of our polar decoder has been performed in additive-white Gaussian-noise (AWGN) channel condition. Eventually, ASIC synthesis and post-layout simulation of the proposed decoder is carried out using united-microelectronics-corporation (UMC) 65 nm and 90 nm CMOS processes in industry-standard EDA tools.

The rest of the paper is organized as follows. Section II presents a brief mathematical overview of polar code and decoding algorithm. Section III presents the proposed

2380-6923/19/$31.00 ©2019 IEEE
DOI 10.1109/VLSID.2019.00075

technique and architecture of PE as well as overall polar decoder. Experimental results and comparisons have been included in section IV and finally, we conclude this paper in section V.

II. Preliminaries

Polar code is error-correcting binary-linear block-code based on the channel polarization. In this phenomenon, recursive channel combining and splitting operations result in a set of channels, of which some are noisy with a capacity 0 and others are noiseless with a capacity 1. Hence, the polar code can be constructed by assigning the information bits to the reliable channels and freezing the unreliable ones to 0 [1]. Encoding is performed using the generator matrix denoted by G_N for a code length of N, where N=2^n for n>0, and G_N is the n^{th} Knocker power of the kernel matrix F. It is represented as G_N=$F^{\otimes n}$ where n and $\otimes$ denote $\log_2 N$ and Knocker product respectively. Here, the F matrix is given as follows:

$$F = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} \tag{1}$$

The codeword is calculated as $x_1^N = u_1^N G_N$ where x_1^N is the codeword vector and u_1^N is the information vector. Here u_i=0 if $i \in A$ such that A is a set of frozen bits. Polar code is iteratively decoded using factor graph based belief propagation algorithm [6]. For a (N,k) polar code, factor graph has n-stages and (n+1)N nodes, as shown in Fig. 1. There are two types of logarithmic-likelihood-ratios (LLRs) messages associated with each node (j,i) in the graph: one is for left-to-right message $L_{j,i}^t$ and another is right-to-left message $R_{j,i}^t$ where t is the current iteration index. In addition, the message is updated based on the following equation:

$$\begin{aligned} L_{j,i}^t &= f(L_{j+1,2i+1}^{t-1} + R_{j,i+N/2}^t, L_{j+1,2i}^{t-1}) \\ L_{j,i+N/2}^t &= f(L_{j+1,2i}^{t-1}, R_{j,i}^t) + L_{j+1,2i+1}^{t-1} \\ R_{j+1,2i}^t &= f(L_{j+1,2i+1}^{t-1} + R_{j,i+N/2}^t, R_{j,i}^t) \\ R_{j+1,2i+1}^t &= f(L_{j+1,2i}^{t-1}, R_{j,i}^t) + R_{j,i+N/2}^t \end{aligned} \tag{2}$$

where

$$f(x,y) = sign(x) \cdot sign(y) \cdot min(|x|,|y|). \tag{3}$$

This is referred as the min-sum (MS) belief-propagation decoding algorithm. It has a complexity of four compare-selects and four additions. In round-trip scheduling, either of the L or R is computed at one instance of an iteration of 2n-1 cycles [9]. This reduces the complexity to two compare-selects and two additions. Equations from (2) can be generalized into two categories [9]: Type-1 that is represented as d=$a + sign(b) \cdot sign(c) \cdot min(|b|,|c|)$ where a=$L_{j+1,2i}^{t-1}$, b=$R_{j,i}^t$, c=$L_{j+1,2i+1}^{t-1}$ & d=$L_{j,i+N/2}^t$ for L computation; or c=$L_{j+1,2i}^{t-1}$, b=$R_{j,i}^t$, a=$R_{j,i+N/2}^t$ or d=$R_{j+1,2i+1}^t$ for R computation. Similarly for Type-2, d=$sign(a) \cdot sign(b+c) \cdot min(|a|,|b+c|)$ where a=$L_{j+1,2i}^{t-1}$, b=$L_{j+1,2i+1}^{t-1}$, c=$R_{j,i+N/2}^t$ & d=$L_{j,i}^t$ for L computation; or a=$R_{j,i}^t$, b=$L_{j+1,2i+1}^{t-1}$, c=$R_{j,i+N/2}^t$ or d=$R_{j+1,2i}^t$ for R computation. A sign-magnitude based approach to develop architectures for these equations along with a critical path reduction technique is discussed in [7]. In this paper, we propose 2's complement based approach with modified PE architecture which further reduces the critical path.

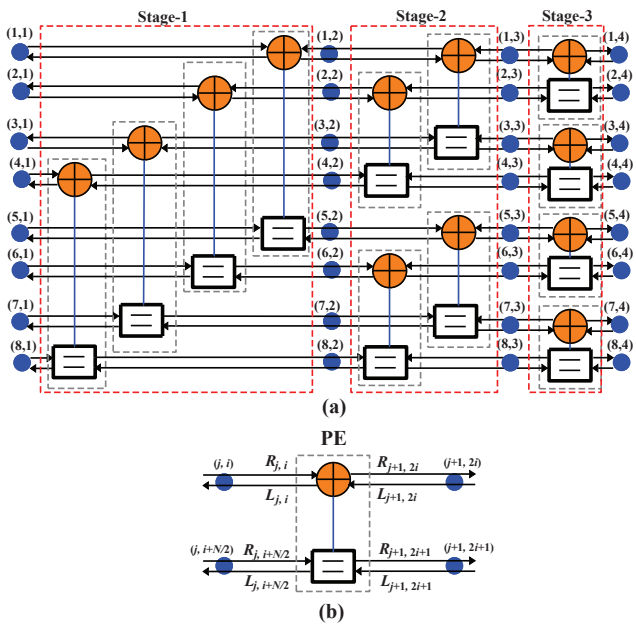

Figure 1. (a) Factor graph for decoding (N=8, k=4) polar code with n stages and (n+1)N nodes (b) Schematic block diagram of a processing element.

III. Proposed VLSI Architectures

A. Processing element architecture

The conventional non-scaled min-sum (MS) algorithm has a critical path delay of approximately four adders (4×T_{adder}) [7]. In order to improve error correction performance, scaled unit has been used and is referred as scaled min-sum (SMS) algorithm in this reported work. It is clear that the inclusion of such scale unit increases the critical path delay to 5×T_{adder} [7]. Such 25% increase in the critical path delay has been tradeoff with the performance improvement. In addition, the technique applied to reduce the critical path delay in SMS algorithm, when applied to non-scaled MS algorithm further reduces its critical path delay from 4×T_{adder} to 3×T_{adder} [7]. On the other hand, we have proposed a new approach which deals with the 2's complement representation for the computations in PE. Assuming that x and y are the inputs, we exploited the most-significant-bits (MSBs) of two operations: (y-x) and (y+x) to

obtain the results from computations carried out in the PE unit of the polar decoder. Thereby for the non-scaled MS algorithm, the new architecture proposed in this paper has reduced the critical path delay to $2{\times}T_{adder}$ in comparison to the reported work [7]. Comprehensive explanation of our approach and the corresponding architectures are presented in this subsection. Our approach has been segregated into

Table I
ILLUSTRATION OF MSBS FOR THE OPERATIONS (y-x) AND (y+x) TO OBTAIN THE RESULT $sign(x){\cdot}sign(y){\cdot}min(|x|,|y|)$

Operand	Case	Operation	MSB	Result $sign(x)\cdot sign(y)\cdot \min(\lvert x\rvert,\lvert y\rvert)$
x=+ve,y=+ve	$\lvert x\rvert > \lvert y\rvert$	$y-x$	1	y
		$y+x$	0	
x=+ve,y=+ve	$\lvert x\rvert < \lvert y\rvert$	$y-x$	0	x
		$y+x$	0	
x=+ve,y=-ve	$\lvert x\rvert > \lvert y\rvert$	$y-x$	0	x
		$y+x$	0	
x=+ve,y=-ve	$\lvert x\rvert < \lvert y\rvert$	$y-x$	1	-x
		$y+x$	1	
x=-ve,y=+ve	$\lvert x\rvert > \lvert y\rvert$	$y-x$	0	-y
		$y+x$	1	
x=-ve,y=+ve	$\lvert x\rvert < \lvert y\rvert$	$y-x$	0	x
		$y+x$	0	
x=-ve,y=-ve	$\lvert x\rvert > \lvert y\rvert$	$y-x$	0	-y
		$y+x$	1	
x=-ve,y=-ve	$\lvert x\rvert < \lvert y\rvert$	$y-x$	1	-x
		$y+x$	1	

four different cases and in each case there are two sub cases, as presented below.

1) Both the input values of x and y are positive: If the magnitude of x is greater than the magnitude of y then the MSBs of (y-x) and (y+x) are 1 and 0 respectively. Therefore, the resulting value of equation 3 is y. Else both the MSBs of (y-x) and (y+x) are 0, which delivers the final result as x, as presented in Table I.

2) The input values of x is positive and y is negative: Here, if the magnitude of x is greater than the magnitude of y then the MSBs of both (y-x) and (y+x) is 0. Hence, the resulting value of equation 3 is x. Else both the MSBs of (y-x) and (y+x) are 1, which delivers the final result as -x (2's complement of x).

3) The input values of x is negative and y is positive: Now, if the magnitude of x is greater than the magnitude of y then the MSBs of (y-x) and (y+x) are 0 and 1 respectively. Hence, the resulting value of equation 3 is -y. Else both the MSBs of (y-x) and (y+x) are 0, which delivers the final result as x, as shown in Table I.

4) Both the input values of x and y are negative: As illustrated in Table I, if the magnitude of x is greater than the magnitude of y then the MSBs of (y-x) and (y+x) are 0 and 1 respectively. Thereby, the resulting value of equation 3 is -y (2's complement of y). Else both the MSBs of (y-x) and (y+x) are 0, which delivers the final result as x.

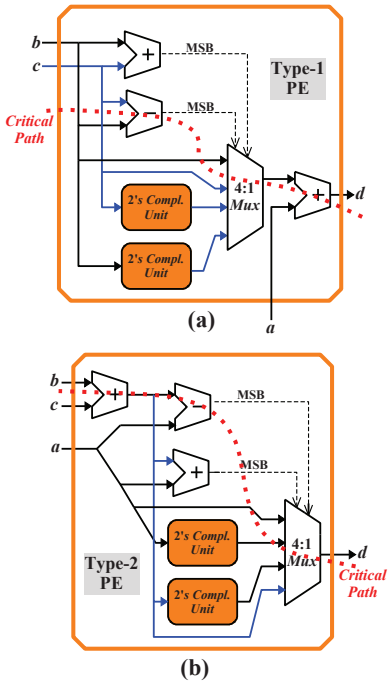

Figure 2. Proposed VLSI architectures of (a) Type-1 (b) Type-2 blocks of the processing element.

Above method indicates that the combination of MSBs of (y-x) and (y+x) operations enables the selection of either x, y, -x or -y (where -x and $-y$ are the 2's complements of x and y respectively). Therefore, the proposed PE-architecture corresponding to this method is shown in Fig. 2 which computes the LLRs for Type-1 and Type-2 categories based on the generalized equations discussed in section II. In these designs, 2's *comp* block determine -x or -y and a 4:1 multiplexer selects the desired output as per the aforementioned cases. Therefore, only subtractor, 4:1 multiplexer and adder are present along the critical path of suggested PE architecture for MS algorithm, as shown in Fig. 2. Hence, the critical path delay (∂_{cr}) equations for the proposed Type-1 and Type-2 blocks of PE-architecture is mathematically represented as

$$\partial_{cr} = T_{sub} + \partial_{4:1-mux} + T_{adder} \approx 2{\times}T_{adder} \tag{4}$$

where T_{sub} and T_{adder} are the delays of q-bit subtractor and adder, respectively. Similarly, $\partial_{4:1-mux}$ represents the delay of 4:1 multiplexer. On the other hand, the critical path delay of the conventional PE-architecture of MS algorithm is $\partial_{cr}{\approx}4{\times}T_{adder}$ which is further optimized by Yuan and Parhi in [7] to

$$\partial_{cr} = 2{\times}T_{sub} + 4{\times}\partial_{2:1-mux} + \partial_{and} + T_{adder} + \partial_{inv} \approx 3{\times}T_{adder} \tag{5}$$

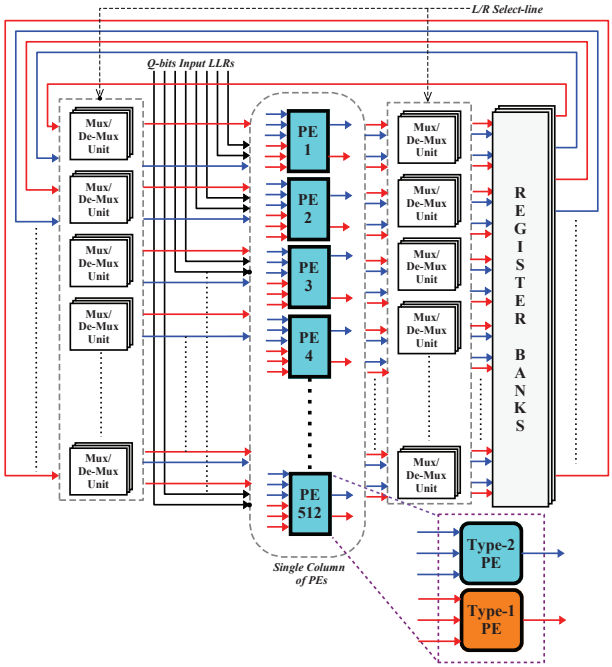

Figure 3. Overall VLSI architecture of unidirectional single-column polar-decoder with round-trip scheduling for N = 1024.

where $\partial_{2:1-mux}$, ∂_{and} and ∂_{inv} are the delays of 2:1 multiplexer, logical 2-input AND-gate and inverter respectively. Thus, our design has shorter critical path as well as lower hardware consumption. To the best of our knowledge, the proposed design is the first approach where the 2's complement input format has been incorporated for the PE architecture. This design can be incorporated with any other scheduling schemes (i.e. one-way, round-trip, halfway, quarter-way [10]) and decoder architectures (i.e. half-column [8], single-column [8], [9], stage-combined [11] and double-column [9]) resulting in further improvement of throughput and hardware efficiency.

B. Overall VLSI architecture of polar decoder

In this paper, proposed PEs are aggregated for the design of single-column and unidirectional polar-decoder with round-trip scheduling and its overall architecture is shown in Fig. 3. This decoder has been designed for a code-length of N=1024 and hence it constitutes $N/2$ (512) PEs in order to process one stage of the factor graph in parallel. We have used 5-bits quantization (Q =5) to represent each LLR (L or R messages) for the PE which includes Type-1 and Type-2 blocks, as shown in Fig. 3. The round trip scheduling used in our design traverses the factor graph in right direction from 1^{st} to 9^{th} stage computing R messages and subsequently traverses in left direction from stage 10^{th} to 1^{st} computing L messages. Since we are using a single-column of PEs in our decoder, such scheduling is ensured with the aid of

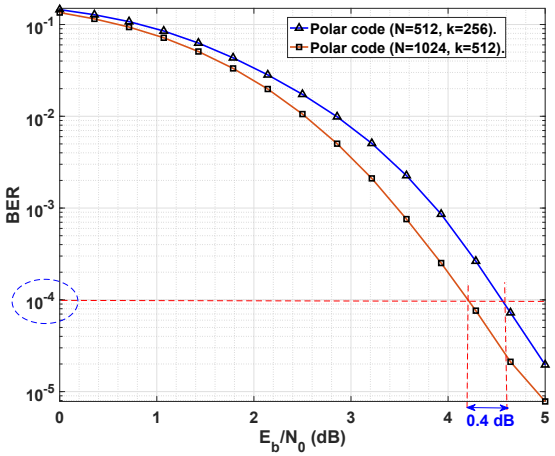

Figure 4. BER performance of belief propagation based polar decoding algorithm using the proposed techniques in AWGN channel environment.

routing unit (that includes multiplexers in parallel) and the control signal that is referred as L/R *Select-line*, as shown in Fig. 3. Such scheduling provides even dissemination of information with one decoding iteration taking $2n$-1 cycles. Once the maximum number of iterations (a pre-defined value) is reached, hard decision of bits are performed based on the total LLRs ($L_{j,1}^{t_{max}} + R_{j,1}^{t_{max}}$) associated with the node (j,1). Multiplexers and de-multiplexers in the routing units are used for input and output routing respectively. In this architecture, the clock period is determined by delays of input routing unit, PE, output routing unit, and register bank.

IV. Performance Analysis, VLSI Design and Comparisons

In order to analyze the bit-error-rate (BER) performance of suggested polar-decoding technique, we performed high-level Monte-Carlo simulation of decoding 10^6 transmitted bits for each of the channel conditions. Specifically, such extensive performance analysis has been carried out in AWGN channel environment where the transmitted and received bits are binary-phase shift-keying (BPSK) modulated and demodulated respectively. We performed such BER simulation for a range of E_b/N_o (in dB) values from 0 to 5 dB and the plots are shown in Fig. 4. These BER plots have been obtained for two different polar codes: (N=1024, k=512) and (N=512, k=256) where the code rate is 1/2. It can be observed from Fig. 4 that the (1024, 512) polar code delivers adequate BER of 10^{-4} at 4.2 dB and it has a coding gain of 0.4 dB compared to the performance of (512, 256) polar code, as shown in Fig. 4.

The overall proposed architecture of polar decoder has been coded using Verilog hardware-description-language (HDL) and its functional verification has been carried out

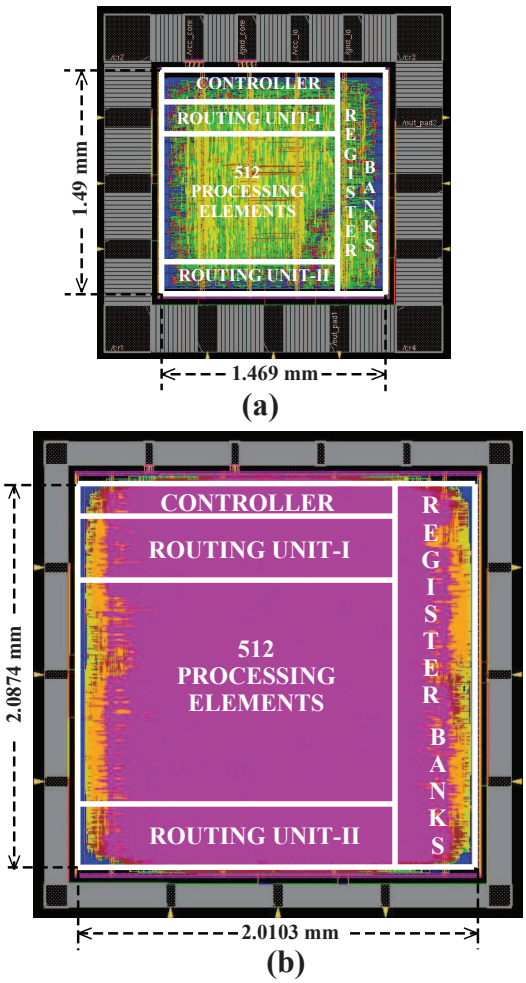

Figure 5. ASIC chip layouts of the proposed belief-propagation polar-decoder in (a) UMC 65 nm-CMOS process (b) UMC 90 nm-CMOS process.

using Synopsys Verilog-Compiler & Simulator (VCS) tool. Subsequently, we synthesized and static-timing analyzed (under worst-case or slow-slow corner) our design using Design-Compiler and Prime-Time tools, respectively, from Synopsys using the standard-cell libraries of Faraday-UMC 90 nm & 65 nm CMOS technology-nodes (Logic SP/HVT Low-K Process). As a result, the generated gate-level netlist of our design has critical path delays of 1.84 ns and 0.859 ns at 90 nm and 65 nm CMOS processes respectively. Physical design (floor-planning, placement, power-planning, clock-tree-synthesis, power-routing, nano-routing etc.) of our netlist were performed using LEF files (7 metal-layer) of UMC 90 nm & 65 nm CMOS processes in Cadence-Innovus tool. Finally, the post-routing STA and post-layout simulation were carried out in Synopsys-VCS environment for reliable functionality. The final chip-layouts of our design have core areas of 4.196 mm^2 at 90 nm-CMOS process and 2.189 mm^2 at 65 nm-CMOS process consuming 370.97 mW and 193.5 mW of total powers, respectively, as shown in Fig. 5. The proposed decoder-architecture has been designed with 5-bits quantization (Q=5) and it consumes 533.827k gate equivalents.

Implementation results of our polar decoder have been compared with the reported works from literature, as shown in Table II. It shows that the achievable clock frequency of our design is 46.31% higher than the highest clock frequency presented by Sun and Zhang in [8]. This is due to the reduced critical path delay of our decoder, as discussed in section III. Additionally, this design requires 19 clock cycles for single iteration and hence for the 5.26 average iterations, it requires 99.94 cycles to deliver first decoded bit (latency). It can be observed from Table II that the proposed decoder achieves 10.1% higher throughput in comparison to the state-of-the-art implementation reported by Abbas et al. [10]. On the other hand, hardware resources required to attain the given throughput is quantified by computing the hardware efficiency shown in Table II. It is to be noted that the lower value of such hardware efficiency is desirable and our work delivers 0.18 of such efficiency.

V. Conclusion

This work presented a new PE architecture with shorter critical-path delay and lower hardware complexity. Since PE is a fundamental unit of the systolic-like polar-decoder architecture, we focused ourselves in the optimization of this unit that will eventually have massive impact in the overall design. This idea has been successfully executed in this paper where the final decoder architecture could achieve profound throughput operating at higher clock frequency. At present, we are at the transition phase from 4G to 5G technology and the first commercial 5G-network is expected to be deployed by the year 2020. The specification of 5G has standardized polar codes for communicating control information in cellular network and hence the throughput requirement of this new-radio physical layer is greater than 10 Gbps. In this work, we have presented a feasible architecture of polar decoder with the throughput>10 Gbps. Therefore, our design can be the potential decoder that can be incorporated for future generation of communication. Lastly, the proposed PE has a general architecture that can be included in any belief propagation polar-decoder design to achieve higher throughput.

References

[1] E. Arikan, “Channel polarization: A method for constructing capacity-achieving codes for symmetric binary-input memory-less channels,” *IEEE Transactions on Information Theory*, vol. 55, no. 7, pp. 3051-3073, 2009.

[2] T. Richardson and S. Kudekar, “Design of low-density parity check codes for 5G new radio,” *IEEE Commun. Mag.*, vol. 56, no. 3, pp. 28-34, Mar. 2018.

Table II
COMPARISON OF THE PROPOSED BELIEF-PROPAGATION POLAR-DECODER WITH THE REPORTED IMPLEMENTATIONS IN LITERATURE FOR $N = 1024$ BITS

	This work	**This work**	[10] TVLSI-2017	[8] ISCAS-2016	[9] VLSIC-2014	[7] ICASSP-2013
Technology (nm)	90	65	65	65	65	-NA-
Area (mm^2)	4.196	2.189	1.6	1.325	1.476	-NA-
Quantization (bits)	5	5	5	6	5	-NA-
Avg. # of Iter.	5.26	5.26	6.34	5	6.57	60
Max. Clock Freq. (GHz)	0.543	1.164	0.334	0.625	0.3	-NA-
Latency (Cycles)	99.94	99.94	31.7	100	65.7	600
Throughput (Gbps)✠	5.6	11.9	10.7	6.4	4.676	1.33
Hardware Efficiency (mm^2/Gbps)♣	0.86	0.18	0.15	0.21	0.32	-NA-

✠: Throughput (Θ_T) = $\frac{F \times N}{\lambda}$ bits/second; where F = Maximum clock frequency, λ = Latency (Cycles), and N = Code Length.
♣: Hardware efficiency = $\frac{\Lambda}{\Theta_T}$ mm^2/Gbps where Λ = Design Area.

[3] Y. Yifei and W. Xinhui, "5G new radio: Physical layer overview," *ZTE Communications*, vol. 15, no. S1, pp. 3-10, Jun. 2017.

[4] E. Arikan, "A performance comparison of polar codes and Reed-Muller codes," *IEEE Communications Letters*, vol. 12, no. 6, pp. 447-449, 2008.

[5] C. Leroux, A. J. Raymond, G. Sarkis, and W. J. Gross, "A Semi-Parallel Successive-Cancellation Decoder for Polar Codes," *IEEE Transactions on Signal Processing*, vol. 61, no. 2, pp. 289-299, 2013.

[6] F. R. Kschischang, B. J. Frey, and H. A. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Transactions on Information Theory*, vol. 47, no. 2, pp. 498-519, 2001.

[7] B. Yuan and K. K. Parhi, "Architecture optimizations for belief propagation polar decoders," *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 2654-2658, 2013.

[8] S. Sun and Z. Zhang, "Architecture and optimization of high-throughput belief propagation decoding of polar codes," *IEEE International Symposium on Circuits and Systems (ISCAS)*, pp. 165-168, 2016.

[9] Y. S. Park, Y. Tao, S. Sun, and Z. Zhang, "A 4.68Gb/s belief propagation polar decoder with bit-splitting register file," *Symposium on VLSI Circuits Digest of Technical Papers 2014*, pp. 1-2, 2014, .

[10] S. M. Abbas, Y. Fan, J. Chen, and C. Y. Tsui, "High-Throughput and Energy-Efficient Belief Propagation Polar Code Decoder," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 25, no. 3, pp. 1098-1111, 2017.

[11] J. Sha, X. Liu, Z. Wang, and X. Zeng, "A memory efficient belief propagation decoder for polar codes," *China Communications*, vol. 12, no. 5, pp. 34-41, 2015.

[12] H. Vangala, E. Viterbo, and Y. Hong, "A comparative study of polar code constructions for the awgn channel," arXiv preprint arXiv:1501.02473, 2015.

[13] A. Pamuk, "An FPGA implementation architecture for decoding of polar codes," *8th International Symposium on Wireless Communication Systems*, pp. 437-441, 2011.