Finding the First Two Minima and Index for Parity Check LDPC Codes

M. Mounica[1] M.Srujana[2]
[1]M.Tech. Student [2]Associate Professor
[1,2]Department of Electronic & Communication Engineering
[1,2]Pathfinder Engineering College (Pech), Thimmapur, Hanamkonda,Warangal Telangana,India

Abstract— This paper represents that hardware efficient design to findout the first rwo minimum values and its index so this is the most important design for the low complexity decoder structure by using min sum algorithm. This proposed algorithm can represents that the usage of comparators are less than previous desgns and which can reusable to compare the results for the first minimum and also used collect the second order elements for second minima. The area time complexity should be improved in this proposed design.

Key words: Low-Density Parity-Check (LDPC) Codes, Tree Structure, Digital IC's, Minimum Values and Index

I. INTRODUCTION

DUE to the powerful error-correcting capability, lowdensity parity-check (LDPC) codes have wide been applied to wireless communication system, personal area networks, and solid-state drives.Toeliminatethe difficult hyperboliccomputations needed withinthe sum–product secret writing rule, recent LDPC decoders square measure implemented supported the min–sum (MS) secret writing rule. within the MS rule, the check-node (CN) operation computes the primary 2 minima and also the index of the primary minimum among several variable-to-check messages given as inputs.

Generally, the hardware block that finds the primary 2 minima, which is termed a looking out module (SM), is enforced by using the balanced tree structure. The number of inputs to be compared in choosing the primary 2 minima is increasing to attain sturdy and long LDPC codes. For example, a recent SM developed for storage applications deals with quite a hundred inputs. The hardware complexness of such a fancy SM takes a major portion within the overall complexity of associate LDPC decoder. Moreover, the realm taken by multiple SMs becomes a lot of hefty during a high-throughput decoder, as large CN operations square measure performed in parallel to increase the secret writing turnout.

A novel tree structure is planned during this transient to attenuate the number of comparators likewise because the area–time (AT) complexity. rather than finding the precise second minimum once finding the primary minimum, the planned rule collects the candidates of the second minimum whereas finding out the primary minimum. The candidate set is well made by reusing the comparison results performed for the primary minimum. Compared to the previous SM, the planned SM reduces the amount of comparators by quite four-hundredth.

II. LITERATURE REVIEW

For a given set of k w-bit inputs, X =, the SM for k inputs produces 3 outputs: 1) the primary minimum value MIN1 = min, 2) the second minimum price, MIN2 = min}, and 3) the index of the primary minimum IDX, that is i if xi is MIN1. 2 2 input primitive units, C1M1 and C1M2, square measure wide accustomed notice associate SM. The C1M1 unit that selects the smaller value from 2 inputs consists of 1 comparator and one w-bit 2-to-1 multiplexor. On the opposite hand, the C1M2 unit is created of one comparator and 2 w-bit 2-to-1 multiplexors to work out both the larger and smaller values.

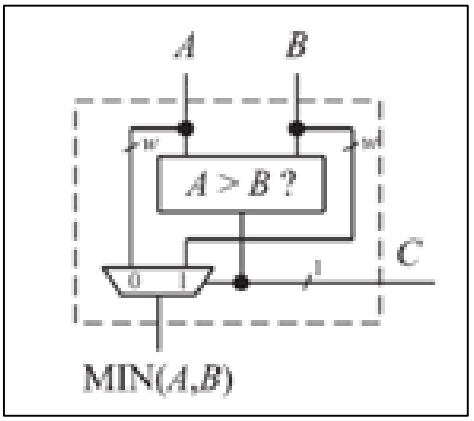

Fig. 1: CIM1 unit

For the sake of simplicity, we tend to focus during this transient on the generation of MIN1 and MIN2, as IDX is obtained exploitation the results of comparisons performed for MIN1. additionally, let the number of inputs k be an influence of two, i.e., k = 2m.When k isn't a power of two, such associate SM is achieved by pruning some leaf nodes of the balanced SM designed with 2m inputs wherever 2m > k, as delineated within the previous literatures.

Fig. a pair of depicts the standard sorting-based SM, referred to as SMsort, handling eight inputs. the general process consists of 2 steps: 1) finding MIN1 with the binary tree structure and 2) selectingMIN2 by suggests that of the multiplexing network controlled by IDX [9]. As shown in Fig. 3, IDX can merely be generated from the comparison results, where cij represents the jth comparison result at the ith step of the binary tree. The multiplexing network generates a candidate set of MIN2, Y =, by using 3 8-to-1 multiplexors.

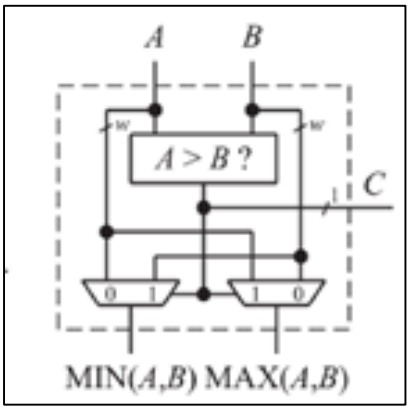

Fig. 2: CIM2 Unit

After selecting 3 candidates, 2 C1M1 units square measure used to determine MIN2. As a result, the SMsort necessitates 9 comparators, 3 8-to-1 multiplexors, and 9 2-to-1

multiplexors to method eight inputs and what is more suffers from the long important delay caused by the serially connected structure. Due to the miscellaneous management at the multiplexing network, the important delay of SMsort is slightly larger than 5TC + 3TM1 + TM9, where TC, TM2, and TM8 indicate the delay of a comparator, a 3-to-2 multiplexor, associated an 9-to-2 multiplexor, correspondingly. For a high-speed realization, the tree-based SM design, referred to as SMtree, was planned in. The SMtree designed for eight inputs is exemplified in Fig. 4, where the processing times for MIN2 and MIN3 square measure virtually identicalas they're each maintained the hierarchical tree construction.

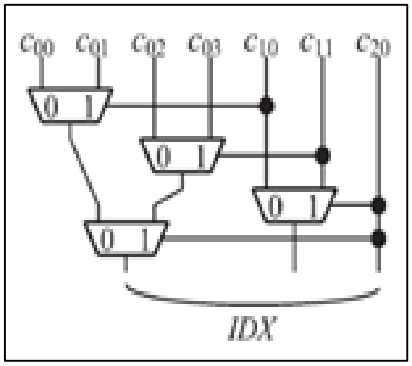

Fig. 3: Index generation block

To calculate detailed MIN2 in every subtree, though, SM sapling requirements more comparators than SMsort. 1 C2M2 units and one 3-to-2 multiplexor square amount in addition accustomed mix 3 subtrees, but the serially connected block needed for locating MIN2 in SMsort is removed so the important delay of SMtree is reduced to 2TC + 4TM1.

A more rapidly tree grounded SM, represented as FM radix, was accomplished by accepting the mixed radix theme. Though, understanding the high radix computation will intensification comparators and multiplexors drastically. Fig. 4. Tree-based looking out module for eight inputs.As the hardware complexness of a comparator is hefty, the preceding tree-based FM can't be price effective once thenumberofinputs isn't little, notably fr recent sturdy GDPC codes targeting a row degree of quite a hundred.Hence, it's compulsory to grow are assignment SM that ll gauge posterior mandible comparators while possession the imperative delay but that of SM kind.

III. PROPOSED DESIGN IMPLEMENTATION

It is potential to scale back the amount of comparators required for the second minimum by reusing the comparison results performed for the primary minimum. Within the planned design, a candidate set Y for MIN2 is 1st made by exploitation the prior comparison results, then a comparison network is in addition constructed to pick MIN2. This ballroom dancing approach is conceptually like SMsort, however the second step is much quicker withinthe planned design. As antecedently mentioned, SMsort needs complicated multiplexing networks to construct the candidate set and so suffers from the long important delay ensuing from k-to-1 multiplexors.

To eliminate the complicated k-to-1 multiplexors, the planned architecture introduces a basic unit, i.e., PROk, that produces the first minimum of k inputs and m (= log2 k) candidatesfor the second minimum, as pictured in Fig. 5(a). Similar to SMtree, a PROk unit is recursively designed with 2 smaller PROk/2 units, as shown in Fig. 5(b). the primary minimum of k inputs, i.e., MIN1, is just selected by comparison two minima, i.e., MIN(L) 1 and MIN(R) 1 , created in PRO(L) k/2 and PRO(R) k/2 units, severally. looking on the comparison result of the C1M2, the PROk decides m − one candidates for the second minimum by choosing either the candidate set of PRO(L)k/2 or that of PRO(R) k/2. If MIN(L) 1 is smaller than MIN(R) 1 , all the m − one candidates of PRO(R) k/2 can't be the second minimum, as a result of MIN(R) 1 is that the smallest price among the 2m−1 inputs on the proper facet. Therefore, m candidates for the second minimum square measure merely shaped by as well as one in every of MIN (L) 1 and MIN(R) 1 to the m − one candidates selected by the result of the C1M2. In short, a PROkunit is accomplished with 2 PROk/2 units, one comparator and m + one 2-to-1 multiplexors. it's apparent that a PRO2 unit processing 2 inputs is a twin of the C1M2 unit.

The major purpose of the PROk is that the candidate set for the second minimum is built in parallel with the looking out for the primary minimum. once finding the primary minimum, another tree structure which will be designed with m − one C1M1 units is employed to find MIN2 among m candidates. Fig. half-dozen shows however the planned SM, spoken asSMpro,is constructed to method eight inputs, wherever a PRO8 unit is followed by a tree structure composed of 2 C1M1 units to find MIN2 among the 3 candidates created within the PRO8 unit. The SMpro in Fig. half-dozen needs nine comparators and twenty 2-to-1 multiplexors to method eight inputs, whereas the SMtree in Fig. four necessitates thirteen comparators and twenty 2-to-1 multiplexors for identical range of inputs. The important delay of SMpro is 5TC + 5TM2, that is way smaller than that of SMsort.Table I compares the hardware complexities and significant delays of 3 totally different SM architectures, wherever the amount of inputs is assumed to be an influence of two, k = 2m. As there are m final candidates for MIN2, SMsort and SMpro need 2m + m − a pair of comparators in determinative 2 minima, which is much smaller than that of SMtree. because the planned design completely removes the 2m-to-1 multiplexors that square measure

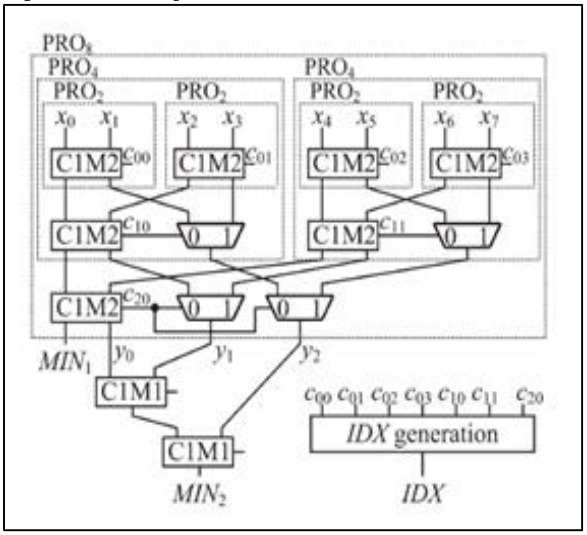

Fig. 4: Proposed Design

IV. IMPLEMENTATION RESULTS

In this paper I have designed this project and I found the two minimum values and the index of the first minima value. This total phenomenon is designed using Verilog HDL. The RTL description is synthesised and simulated in Xilinx ISE 14.5. The simulated wave forms are presented below.

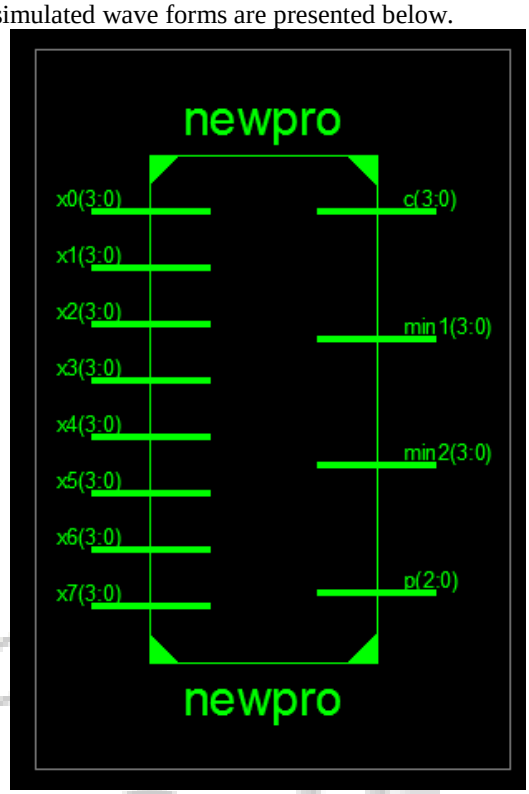

Fig. 5: Top level diagram of the proposed system

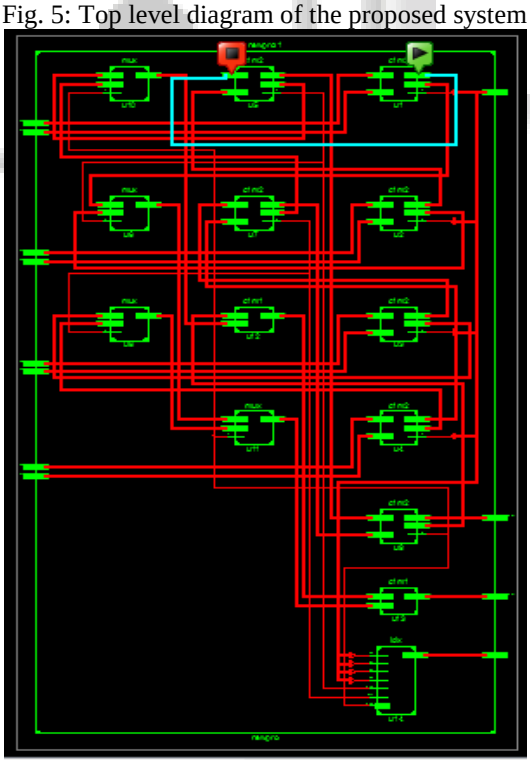

Fig. 6: Internal structure of the proposed system

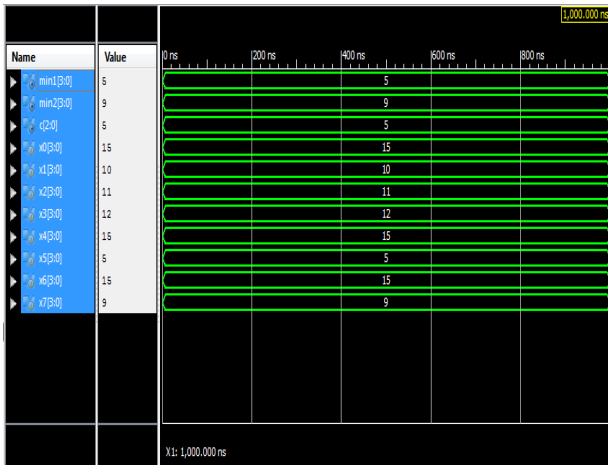

Fig. 7: Simulation result

V. CONCLUSION

We have presented a novel tree structure that finds the first two minima among many inputs. In the proposed structure, the candidates of the second minimum are collected by utilizing the results of comparisons performed for the first minimum. Hence, the proposed structure minimizes the number of comparators, leading to a low-complexity realization. In addition, the second minimum is selected from the candidates by carrying out a few comparison steps. As the proposed structure eliminates the large-sized multiplexing networks, it improves the AT complexity significantly compared to those of the previous state-of-theart SMs.

VI. REFERENCES

[1] IEEE Standard for Local and Metropolitan Area Networks Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, IEEE Std. 802.11n-2009, Oct. 2009.

[2] IEEE Standard for Local and Metropolitan Area Networks Part 16: Air Interface for Broadband Wireless Access Systems, IEEE Std. 802.16-2009, May 2009.

[3] IEEE Standard for Local and Metropolitan Area Networks Part 15.3: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for High Rate Wireless Personal Area Networks (WPANs), IEEE Std. 802.15.3c-2009, Oct. 2009.

[4] J. Kim and W. Sung, "Rate-0.96 LDPC decoding VLSI for soft-decision error correction of NAND Flash memory," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 22, no. 5, pp. 1004–1015, May 2014.

[5] J. Kim, D. Lee, and W. Sung, "Performance of rate 0.96 (68254, 65536) EG-LDPC code for NAND Flash memory error correction," in Proc. IEEE ICC, 2012, pp. 7029–7033.

[6] Y. Sun and J. R. Cavallaro, "VLSI architecture for layered decoding of QC-LDPC codes with high circulant weight," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 21, no. 10, pp. 1960–1964, Oct. 2013.

[7] F. Angarita, J. Valls, V. Almenar, and V. Torres, "Reduced-complexity min–sum decoding algorithm for decoding LDPC codes with low errorfloor," IEEE Trans. Circuits Syst. I, Reg. Papers, vol. 61, no. 7, pp. 2150–2158, Jul. 2014.

[8] Y.-L. Ueng, B.-J. Yang, C.-J. Yang, H.-C. Lee, and J.-D. Yang, "An efficient multi-standard LDPC decoder design using hardware-friendly shuffle decoding," IEEE Trans. Circuits Syst. I, Reg. Papers, vol. 60, no. 3, pp. 743–756, Mar. 2013.
[9] D.E. Knuth, The Art of Computer Programming, 2nd ed. New York, NY, USA: Addison-Wesley, 1998.
[10] Q. Xie, Z. Chen, X. Peng, and S. Goto, "A sorting-based architecture of finding the first two minimum values for LDPC decoding," in Proc. IEEE CSPA, 2011, pp. 95–98.
[11] C.-L. Wey, M.-D. Shieh, and S.-Y. Lin, "Algorithms of finding the first two minimum values and their hardware implementation," IEEE Trans. Circuits Syst. I, Reg. Papers, vol. 55, no. 11, pp. 3430–3437, Dec. 2008.

