Low Hardware Layered Decoding Architecture for LDPC Code

N. Aravind[1], L. Praveen Kumar[2]

[1]M. Tech Student
CVSR College of Engineering
aravind.nalika@gmail.com

[2]Assistant professor, ECE Department
CVSR College of Engineering
praveenkumar.lendale@gmail.com

Abstract: ***Low density parity check (LDPC) codes have been extensively adopted in next-generation forward error correction applications because they achieve very good performance using the iterative decoding approach of the belief-propagation (BP). The basic decoder design for achieving the highest decoding throughput is to allocate processors corresponding to all check and variable nodes, together with an interconnection network. In this fully-parallel decoder architecture, the hardware complexity due to the routing overhead is very large. Therefore, much of the work on LDPC decoder design has been directed towards achieving optimal tradeoffs between hardware complexity and decoding throughput. In particular, a time-multiplexed or folded approach, which is known as partially parallel decoder architecture, has been proposed. Low hardware layered decoding architecture for LDPC code scheme is proposed using only one switch network with direct connections. This method requires only one shuffle network, rather than the two shuffle networks which are used in conventional designs. In addition, this project can be extended to block parallel decoding scheme by suitably mapping between required memory banks and processing units in order to increase the decoding throughput.***

Keywords: Decoding, field-programmable gate array (FPGA), forward error correction and low density parity check (LDPC).

1. Introduction

Low-density parity-check (LDPC) codes are a class of linear block LDPC codes. The name comes from the characteristic of their parity-check matrix which contains only a few 1's in comparison to the amount of 0's. Their main advantage is that they provide a performance which is very close to the capacity for a lot of different channels and linear time complex algorithms for decoding. As their name suggests, LDPC codes are block codes with parity-check matrices that contain only a very small number of non-zero entries. It is the sparseness of H which guarantees both a decoding complexity which increases only linearly with the code length and a minimum distance which also increases linearly with the code length. Aside from the requirement that H be sparse, an LDPC code itself is no different to any other block code. Indeed existing block codes can be successfully used with the LDPC iterative decoding algorithms if they can be represented by a sparse parity-check matrix. Generally, however, finding a sparse parity-check matrix for an existing code is not practical. Instead LDPC codes are designed by constructing a sparse parity-check matrix first and then determining a generator matrix for the code afterwards. The biggest difference between LDPC codes and classical block codes is how they are decoded. Classical block codes are generally decoded with ML like decoding algorithms and so are usually short and designed algebraically to make this task less complex. LDPC codes however are decoded iteratively using a graphical representation of their parity-check matrix and so are designed with the properties of H as a focus.

2. Block Parallel Layered Decoding Architecture

2.1 Layered decoding scheme

Structured regular or irregular LDPC codes are described by an Mb× Nb base matrix Hb with Mb=M/z and Nb=N/z, where M is the number of parity check equations, N is the code length, and z is the size of a square sub-matrix. The parity check matrix H of a structured LDPC code can be viewed as the concatenation of constituent codes, where the number of constituent codes is equal to Mb. The dataflow of a typical layered decoder is shown in Fig. 1. Let R= [r1r2…rMb] denote the check-to-variable messages, where rk corresponds to a constituent code of H for 1≤ K ≤ Mb. Q(K) and Q(K+1) are previously decoded soft output value and the newly decoded soft output value used for updating the next block row, respectively. L (k) denotes the variable-to-check message which has entered the decoding update block, and r(k) represents the updated check-to-variable message at the k^{th} block row, which has entered the decoding update block, and at the k^{th} block row. In Fig. 1, the decoding update block, which was presented with a check node-based processor (CNBP) such as approximations of BP. After the initialization of the layered decoder is achieved using the soft values from the channel in the bit update block, the decoder starts updating messages corresponding to the first constituent code (r1). The switch network (SN) 1 shuffles the channel soft values based on the permutation information obtained from r1. The shifted messages Q (1) from SN 1 and the check-to-variable messages r1 read from memory are used to compute the variable-to-check messages. The decoding update block computes the check-to-variable messagesr1+ based on L (1) and stores r1+ back into memory. The updated posterior messages are computed by adding the recently updated check-to-variable messages to the variable-to-check messages, then reshuffled through SN 2 and finally stored as Q(2) in the bit update.

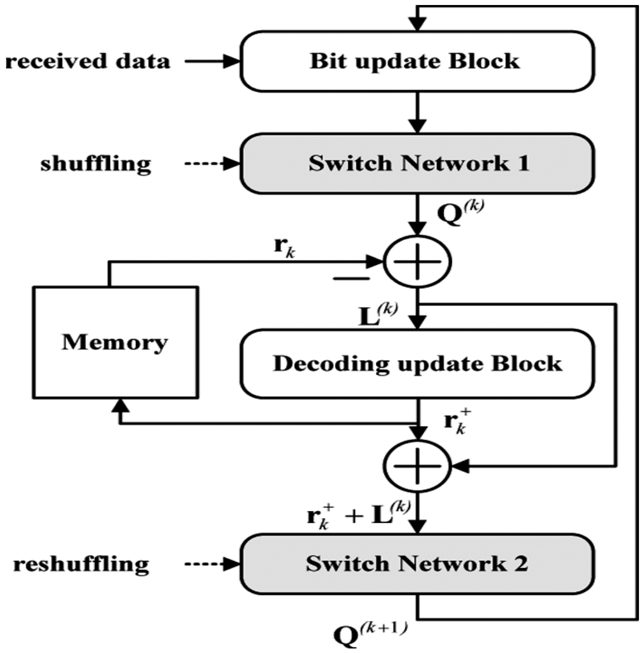
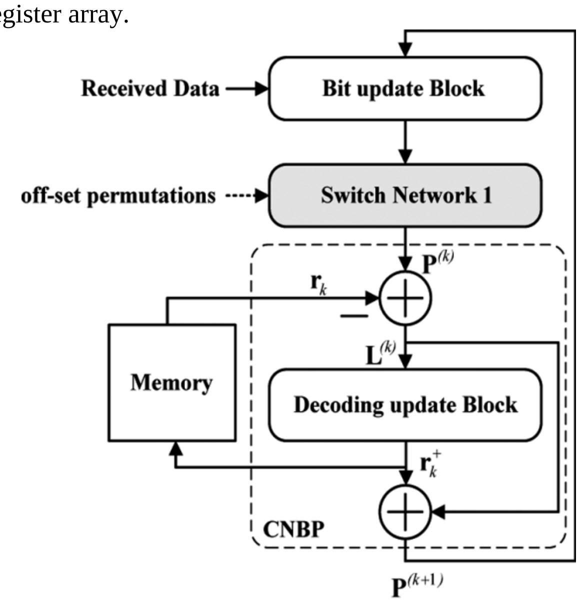


Figure 1: Dataflow of a typical layered decoder

2.2 Proposed layered decoder

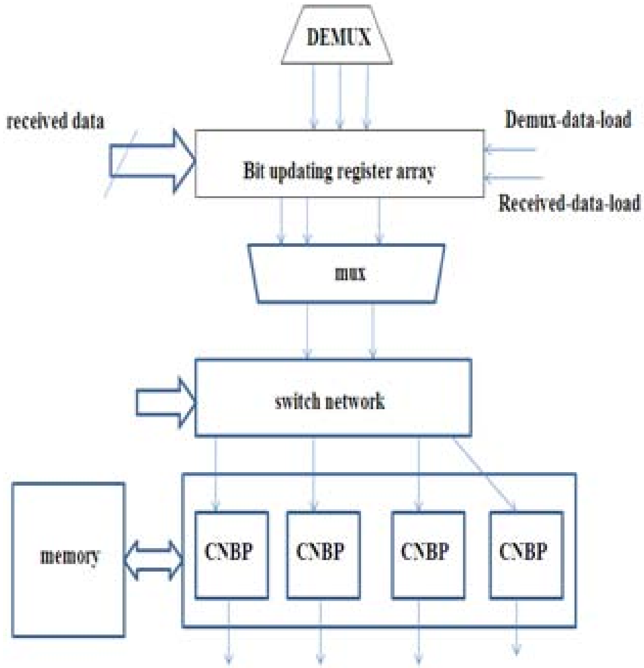
This decoder architecture has the consist of two characteristics: 1) we generate offset shifting values for shuffling and reshuffling messages so that the proposed decoder needs to use only SN 1 rather than two SNs and 2) the number of memory banks for check-to-variable messages is configured to be a row weight of H. The first characteristic is achieved by observing that the operation of the SNs for shuffling and reshuffling messages is overlapped during the updating of the constituent codes of In other words, the SN 2 block in Fig. 1 reshuffles updated output messages corresponding rk to until the decoder reaches the end of one iteration for the complete At the end of a sub-iteration the recently updated outputs are shuffled by SN 1 for the next constituent code. Therefore, the two consecutive operations, reshuffling and shuffling, are not necessary to compute the decoded output within a sub-iteration and this provides an opportunity for reducing the complexity of the interconnections. The second characteristic is used to simultaneously process all messages corresponding to rk in one clock cycle.

The dataflow of the Proposed Layered decoding as shown fig.2 (a) the decoding steps are almost the same as in the conventional decoding with the exception of the ordering patterns in the bit update block and the offset permutations through SN 1. Let P (K) and P (K+1) be the previous and updated soft outputs, respectively. Note that P (K+1) = π Q (K+1) is a permutation of Q (K+1) .The top-level architecture using the layered mode with offset permutations for SN 1 is illustrated in Fig. 2(b). During an initialization operation, the incoming soft message is shifted into the bit updating register array. Then, the registered MUX block simultaneously loads the required messages into SN 1. Following that, SN 1 rotates the input messages by the amount of the offset permutations. The check-to-variable messages and the rotated variable messages loaded into the CNBP blocks are then computed for newly updated check-to-variable messages and rotated soft output messages. The check-to-variable messages are stored in the memory, and the rotated Soft output messages replace the previous messages in the bit-update register array.



(a)

Figure 2: Proposed layered decoder. (a) Modified dataflow with offset permutations



(b)

(b) Block parallel layered decoder architecture.

3. Hardware complexity

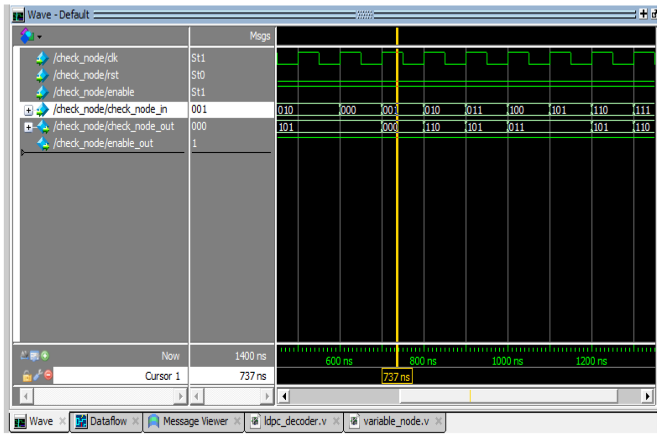
In layered decoding architectures the number of memory bits is reduced by nearly 70% and the number of iterations for achieving the same error rate is also reduced by almost 70% compared with traditional decoder designs. To show the low complexity of the block parallel processor in the layered decoding scheme, we compare it with different decoder architectures. In Xilinx 12.1 version using in layered decoding to dumping the program in Xilinx sparetn 3e xc4vls1200.

Table 1: Xilinx sparetn3e xc4vls1200 FPGA synthesis results

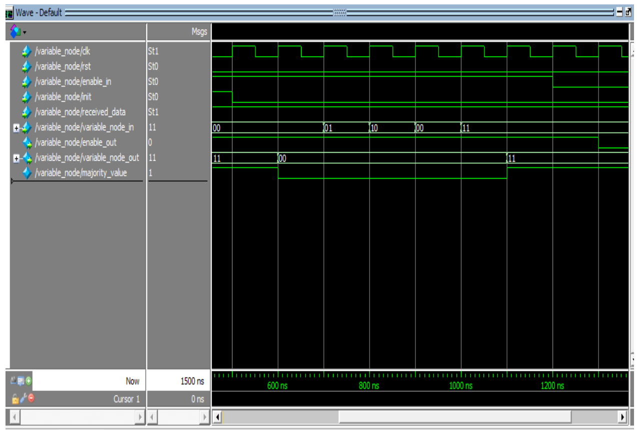
Device Utilization Summary (Estimated Values)			
Logic Utilization	*Used*	*Available*	*Utilization*
Number of Slices	38	768	4%
Number of Size Flip Flops	59	1536	3%
Number of 4 input LUTs	62	1536	4%
Number of Bonded IOBs	16	124	12%
Number of GCLKs	2	8	25%

4. Simulation Results

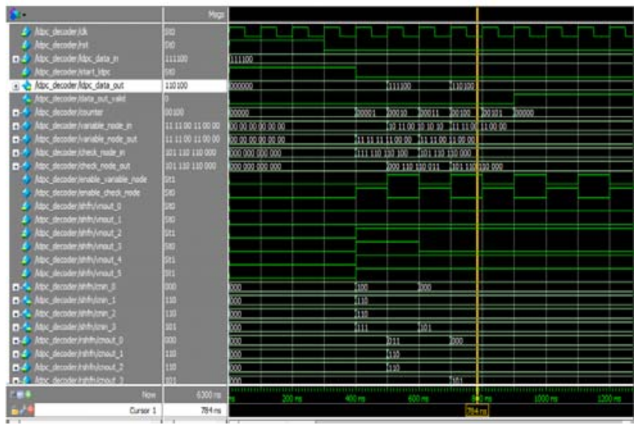
4.1 Check Node Unit



4.2 Variable Node Unit



4.3 LDPC Decoder



5. The advantages LDPC Layered decoder

- As the interconnection between the blocks in the hardware is reduced therefore, the complexity of hardware is less.
- LDPC decoder achieves good coding gain performance as its main advantage is that they provide a performance which is very close to the capacity for a lot of different channels.
- Linear time complex algorithms for decoders.
- As the complexity of hardware is reduced the speed of the LDPC decoder attains high speed compared to other decoders like turbo code decoder.

- LDPC codes exhibits low error floor rate. The error floor rate is defined as the minimum distance is proportional to the code length.

5.1 Future scope

The family of Low Density Parity Check (LDPC) codes is a strong candidate to be used as Forward Error Correction (FEC) in future communication systems due to its strong error correction capability. Most LDPC decoders use the Message Passing algorithm for decoding, which is an iterative algorithm that passes messages between its variable nodes and check nodes. It is not until recently that computation power has become strong enough to make Message Passing on LDPC codes feasible. Although locally simple, the LDPC codes are usually large, which increases the required computation power.

6. Conclusion

The proposed architecture has an efficient architecture for layered LDPC decoding by reducing the interconnection complexity with proper and efficient decoding throughput. Our design requires only a single shuffle network, rather than the two shuffle networks used in prior designs. The results show a significant reduction in the number of required FPGA slices compared to a standard layered decoding architecture.

References

[1] R. G. Gallager,’’ Low-Density Parity-Check Codes.’’ Cambridge, MA: MIT Press, 1963.

[2] A. J. Blanksby and C. J. Howland, “A 690-mW 1-Gb/s 1024-b, rate-1/2 low-density parity- check code decoder,” IEEE J. Solid-State Circuits, vol. 37, no. 3, pp. 404–412, Mar. 2002.

[3] K. K. Parhi, VLSI Digital Signal Processing Systems: Design and Implementation. New York: Wiley, 1999.

[4] T. Zhang and K. K. Parhi, “Joint (3,k)-regular LDPC code and decoder/ encoder design,” IEEE Trans. Signal Process., vol. 52, no. 4, pp. 1065–1079, Apr. 2004.

[5] S. Kim, K. K. Parhi, and R. Liu, “System and method for designing RS-based LDPC code decoder,” U.S. Patent App. US2007-0033484, Feb. 8, 2007.

[6] L. Liu and C.-J. R. Shi, “Sliced message passing: High throughput overlapped decoding of high-rate low-density parity-check codes,” IEEE Trans. Circuits Syst. I, Reg. Papers, vol. 55, no. 11, pp.

[7] M. M. Mansour and N. R. Shanbhag, “A 640-Mb/s 2048-bit programmable LDPC decoder chip,” IEEE J. Solid-State Circuits, vol. 41, no. 3, pp. 684–698, Mar. 2006.

[8] D. E. Hocevar, “A reduced complexity decoder architecture via layered decoding of LDPC codes,” in Proc. IEEE Workshop Signal Process. Syst. (SIPS), Austin, TX, Oct. 2004, pp. 107 112.

[9] Z. Cui, Z. Wang, and Y. Liu, “High-throughput layered LDPC decoding architecture,” IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 17, no. 4, pp. 582–587, Apr. 2009.

[10] K. K. Gunnam, G. S. Cho, and M. B. Yeary, “A parallel VLSI architecture for layered decoding for array LDPC codes,” in Proc. Int. Conf. VLSI Des., Bangalore, India, Jan. 2007, pp. 738–743.

[11] S. Kim, G. E. Sobelman, and H. Lee, “Flexible LDPC decoder architecture for high-throughput applications,” in Proc. IEEE Asia Pacific Conf. Circuits Syst. (APCCAS), Macao, China, Nov. 2008, pp. 45–48.

[12] Draft Standard for Wireless Medium Access Control (MAC)and Physical Layer(PHY) Specifications for High Rate Wireless Personal Area Networks (WPANs): Amendment 2: Millimeter-Wave Based Alternative Physical Layer Extension, IEEE P802.15.3c/D04, 2008. [Online].Available: http://standards.ieee.org

[13] S. Olcer, “Decoder architecture for arrary-code-based LDPC codes,” in Proc. IEEE Global Telecommun. (GLOBECOM) Conf., San Francisco, CA, Dec. 2003, pp. 2046–2050.

Author Profile

N. Aravind received the B. Tech degree in Electronics and Communication Engineering from SLCs institute of Engineering and Technology, JNTU, Hyderabad, in 2010, and currently pursuing the M. Tech degree in VLSI System Design from CVSR College of Engineering, JNTU, Hyderabad respectively. His research interests include digital communication and frontend logic design and hardware/software co verification.

Praveen Kumar Lendale was born in Devarakonda, Andhra Pradesh, India, in 1985. He received the B. Tech degree in Electronics and Communication Engineering from JNTU, Hyderabad, in 2008. He received the M. Tech Masters degree in Communication systems from Aurora Engineering, JNTU, Hyderabad in 2012. His research interest includes Digital Image Processing and Mobile communication.