FPGA Implementation of Non-binary LDPC Decoder using Stochastic Computation

S.Naveen
Department of Electronics and
communication Engineering
SSN college of Engineering
Chennai, India

M.Anbuselvi
Department of Electronics and
Communication Engineering
SSN college of Engineering
Chennai, India

ABSTRACT

Low density parity check (LDPC) codes, a class of linear block code has the superior performance closer to the Shannon's limit. Non-binary LDPC (NB-LDPC) is an extension of the binary LDPC, works on the higher order Galois field. The design of efficient hardware architecture for the NB-LDPC code depends on various factors like input message format, code length, kind of modulation and the type of channel. Non-Binary LDPC codes are designed with the better performance metrics using stochastic computation. The increased computation complexity of the NB-LDPC put forth the major challenge on the hardware realization of the decoder architecture. This paper presents the design of efficient hardware architecture for NB-LDPC decoder based on stochastic computation. The designed architecture is targeted to Xilinx VIrtex device and the synthesis reports are tabulated.

Index Terms

Xilinx Vertex, Stochastic Decoding, Latching

1.INTRODUCTION

low density parity check codes (LDPC) have become one of the most attractive error correction codes due to its excellent performance [1] and suitability in high data rate applications. In terms of performance, binary LDPC codes start to show their weaknesses when the code word length is small or moderate, or when higher order modulation is used for transmission. For these cases, non-binary LDPC (NB-LDPC) codes designed in high order Galois fields have shown great potential. LDPC codes can be iteratively decoded using the well-known Sum-Product Algorithm (SPA) or its approximation, the min-sum algorithm (MSA). The SPA is a message passing algorithm in which two types of processing nodes, known as variable nodes (VNs) and Check Nodes (CNs) iteratively exchange their probability messages and the correctness of the symbols received from the channel. Stochastic decoding [2] is a newly proposed approach for low-complexity iterative decoding of error-correcting codes. The stochastic representation results in simple hardware structure for VNs and CNs. More importantly, because of its bit-serial nature, stochastic representation significantly reduces the number of wires between VNs and CNs which results in reducing the interconnection complexity of the architecture.

Stochastic decoding is, however, prone to the latching problem [3, 4]. It has been observed that cycles in a code graph correlate stochastic messages in a way that a group of nodes sticks into fixed states for several decoding iterations and dramatically degrades the convergence of the decoder [3]. The latching problem prevented early stochastic methods [2, 5-8] from being applied to practical LDPC codes. To overcome this problem, the concept of edge memories (EMs) was introduced in [9]. EMs significantly reduce the chance of latching in the decoder, hence, enabling demonstration of stochastic LDPC decoding of practical LDPC codes. The potential of EM-based stochastic decoding for high-throughput LDPC decoder hardware implementations was recently validated in [10]. The EM-based stochastic decoding method was also extended to the Turbo product codes in [11] and also for decoding non-binary LDPC codes in [12]. Implementation is done for binary LDPC decoders only. Architecture for NB-LDPC is proposed in the references.

In this paper, we provide the design of stochastic based NB-LDPC decoders of code length 504, which is used in under water communications. The remainder of this paper is organized as follows. Section 2 provides the background on stochastic computation. Section 3 elaborates our contribution, area report and power analysis of the designed architecture. Section 4 presents the conclusions.

2. STOCHASTIC COMPUTATION

Stochastic computing is a collection of techniques that represent continuous values by streams of random bits. Complex computations can then be computed by simple bit-wise operations on the streams. The main advantage of stochastic computing is very low cost implementation of arithmetic operations. The drawbacks are exponential increase in bit-stream length, corresponding increase in computation time and low bandwidth. In the NB-LDPC decoder a VNU has latching in Edge Memory is a main issue. Various architectures were proposed in order to eradicate the latching problem.

2.1 Latching

When transmitted symbols at a node are correlated then it is possible for a group of nodes and will make it to be in a idle state. This state is maintained solely by the message within a cycle. No pattern of independent messages is sufficient to restore the cycle to proper function. This phenomenon is called latching.

2.2 NB- LDPC decoder

The performance of stochastic decoding has comparable performance for different SNRs with respect to Sum Product Algorithm (SPA) decoding. The problem of latching is described for stochastic decoding. This method has significant potential for high throughput or low complexity iterative decoding [2].
For each iteration *i*, the following steps are to be performed:

Initialization:

For each *m* and *n*, initialize the value of $Q_{mn}^{a(0)}$ tof_n^a, i.e.

$$Q_{mn}^{a(0)} = f_n^a = \prod_{j=1}^{m} g_{n_j}^{a_j}$$

where a_i is the *i*'th bit of binary representation of *a*.

Check Node Update:

$$R_{mn}^{a(i)} = \sum_{c:c_n=x} P(z_m = 0|c) \prod_{n'\in N(m)\backslash n} q_{mn}^{a(i)}$$

Variable node updates:

For each *m* and *n*, update

$$q_{mn}^{a(i)} = a_{mn} f_n^a \prod_{m'\in M(n)\backslash m} R_{m'n}^{a(i)}$$

Tentative decoding:

For each *n*, make a tentative decoding

$$q_n^a = a_n f_n^a \prod_{m'\in M(n)} R_{m'n}^{a(i)}$$

$$\acute{x}_n = \arg \overset{max}{x} q_n^a$$

If H $\acute{x}$=0, the decoding terminates. Otherwise, we repeat the iterative process until either the decoding is successful or a failure is declared after a fixed maximum number of iterations.

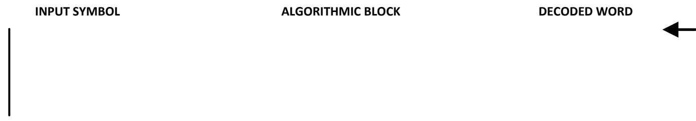
Edge Memories (EMs) is finite depth buffers inserted between variable and check nodes. Output obtained from variable node is taken when equality constraints are satisfied otherwise, a symbol is randomly selected from the stored regenerative symbol. This method has good decoding performance however EMs occupies considerable silicon area when implemented in ASICs.

3. OUR CONTRIBUTION

The architecture of Non-binary LDPC decoder varies in the conversion stages. The initial block is the non-binary to stochastic converter. An input of non- binary data GF (4) is first converted into stochastic stream of non-binary symbols. The next decoding module consists of a Check Node Unit (CNU) and Variable Node Unit (VNU) where a decoding process takes place and finally a stochastic stream is converted into a non-binary data.

3.1 Converter Units

The non-binary to stochastic number converter is a converter which converts a non-binary number into a stochastic stream of symbols. For an example finite field for GF (4) works over {0, 1, 2, 3} respectively. A random number generator is the one generates a number randomly for each clock cycle, for generating a random number Linear Feedback Shift Register (LFSR) is used. A LFSR is a shift register whose input is a linear function of a present state depends on previous state. Mostly used linear function of a single bit is XOR, thus normally it is a shift register whose input is driven by XOR of some bits of the over all shift register value. Various applications of LFSR are counters, encryption, compression, and pseudo-random bit sequences generation



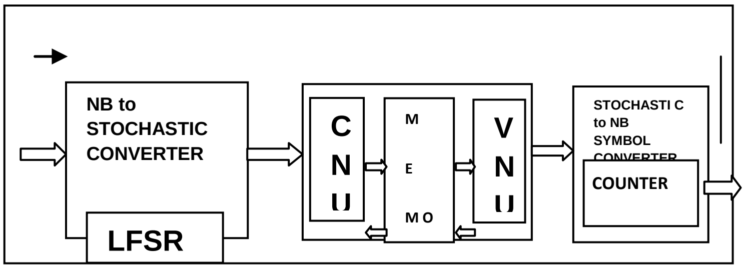

NB to STOCHASTIC CONVERTER
LFSR
C N U
M E M O
V N U
STOCHASTI C to NB SYMBOL CONVERTER
COUNTER

Fig 1.Block diagram for NB-LDPC decoder using stochastic computation

A stochastic number obtained from the variable node update is given to the counter. A counter counts the number of one's in the stochastic stream then the binary value of the counter output is decoded non-binary symbol.

3.2 Decoder Units

The check node unit (CNU) in non-binary LDPC performs the addition operation. Since the equivalent representation of an addition operation in stochastic computation is XOR gate. A stream of non-binary symbols is given as input to the CNU in which the addition is done and it is stored in flip-flop.

$$p_c = p_a(1-p_b) + p_b(1-p_a)$$

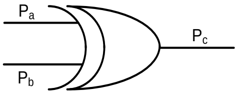


Fig 2: Check Node Unit

A multiplication operation takes place in the variable node unit. A variable node unit of a non-binary LDPC decoder block consists of a memory unit and gates such as AND and XNOR respectively.

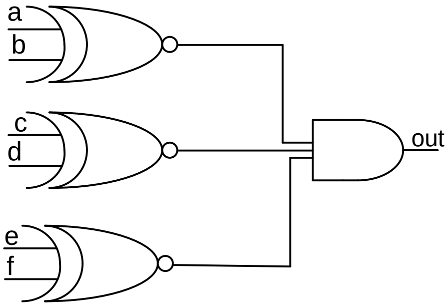


Fig 3: Variable Node Unit

The input to the variable node unit is the output from the check node unit is given, it forms a product and it is stored in a memory unit. For each variation in a data the values of a variable node unit are updated in a memory hence the block is variable node update.

A random address is given to a memory according to that a data is retrieved from the memory location. Thus form a variable node unit of a non-binary architecture.

3.3 Device Utilization and power analysis

The top module for NB LDPC decoder GF (4) 504 bit is modeled and synthesized in Vertex of Xilinx family using

Xilinx ISE tool. An area report is tabulated as shown in Table 1.

Table 1. Area report of NB LDPC decoder GF (4) 504

Logic Utilization	**Used**	**Available**
No. of slice registers	4	301440
No. of LUT's	5	150720
No. of bonded IOBs	177	400
No. of BUF G/BUFG CTRLS	2	32
No. of occupied slices	4	37,680
Average fan out non clock nets	2.14	

The power is estimated for the synthesized NB-LDPC decoder of finite field GF (4) with code length of 504 using Xilinx datasheet is shown in Table 2

Figure 4 shows the graph for various logics used in the designed architecture and Fig.5 shows the graph for power vs. junction temperature.

Table 2. Power analysis summary for NB LDPC decoder GF (4) 504

Total on chip power	2.729W
Junction temperature	28.1^{o}C
Thermal margin	71.9^{o} & 62.2W

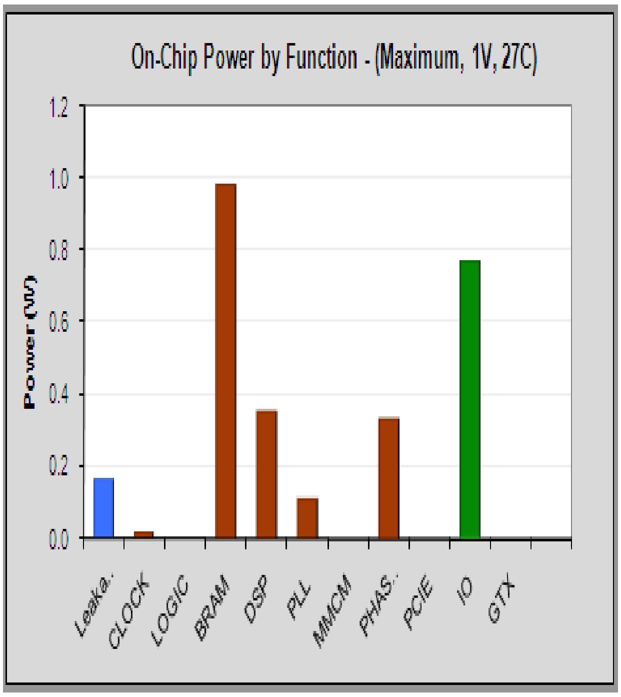


Fig 4: Power utilized by various logics in architecture

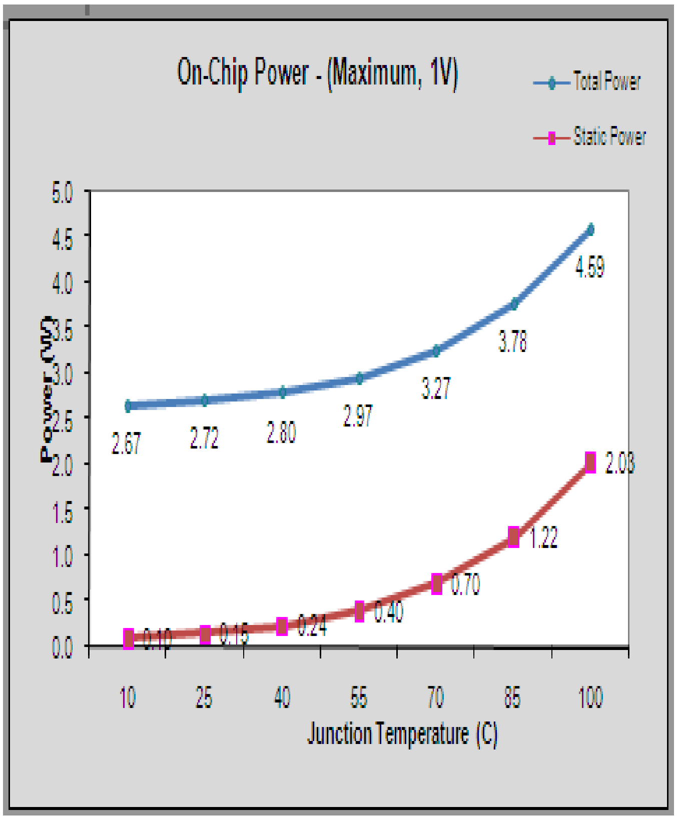


Fig 5. Junction temperature vs. power

4. CONCLUSION

The Stochastic Computation based Non-Binary LDPC decoder architecture is modeled. The designed architecture consists of a non-binary stream to stochastic number generator, decoding algorithmic modules namely CNU and VNU, and the reverse conversion of stochastic stream to the corresponding non-binary number. The designed blocks are integrated as a top module. The top module is synthesized, targeted to Xilinx Vertex and its metric is tabulated and its power analysis is done.

An optimization for a modified non-binary architecture using stochastic computation can be done and it can also be extended to higher order Galois Field with various parameter analyses.

5. REFERENCES

[1] MacKay, D. J. C., & Neal, R. M. 1996. "Near Shannon limit performance of low density parity check codes". Electronics Letters, 32(18), 1645–1646.

[2] Gaudet, V., & Rapley, A. 2003." Iterative decoding using stochastic computation". Electronics Letters, 39(3), 299–301.

[3] Sharifi Tehrani, S., Gross, W. J., & Mannor, S. 2006. "Stochastic decoding of LDPC codes". IEEE Communication Letters, 10(10), 716–718

[4] Winstead, C., Gaudet, V., Rapley, A., & Schlegel, C. 2005. "Stochastic iterative decoders". In IEEE ISIT pp. 1116–1120.

[5] Gross, W. J., Gaudet, V., & Milner, A. 2005. "Stochastic implementation of LDPC decoders". In The 39th

Asilomar conf. on signals, systems, and computers pp. 713–717 Pacific Grove, CA.

[6] Rapley, A., Winstead, C., Gaudet, V., & Schlegel, C.2003. "Stochastic iterative decoding on factor graphs". In Proc. of the 3rd int. symp. on turbo codes and related topics pp. 507–510 Brest, France.

[7] Winstead, C. 2005. "Error-control decoders and probabilistic computation". In Tohoku Univ. 3rd SOIM-COE conf. (pp. 349–352). Sendai, Japan.

[8] harifi Tehrani, S., Mannor, S., & Gross, W. J. 2008. "Fully parallel stochastic LDPC decoders". IEEE Transactions on Signal Processing, 56(11), 5692–5703.

[9] Sharifi Tehrani, S., Jego, C., Zhu, B., & Gross, W. J.2008 "Stohastic decoding of linear block codes with high-densiy parity-check matrices". IEEE Transactions on Signal Processing, 56(11), 5733–5739.

[10] Sarkis, G., Mannor, S., & Gross, W. 2009. "Stochastic decoding of LDPC codes over GF(q)".In ICC 2009 symposium on selected areas in communications. Dresden, Germany

[11] Ali Nandrei, Shie Mannor, Mohamad Sawan Warren J.Gross 2011."Relaxed stochastic decoder of LDPC codes" IEEE transaction on signal processing vol 59 pp5617-5626.

[12] Anthony Rapley, Vincent Gaudet and Chris Winstead 2005."On the simulation of stochastic iterative decoder architectures". IEEE.