International Journal of Research
Available at https://edupediapublications.org/journals

p-ISSN: 2348-6848
e-ISSN: 2348-795X
Volume 04 Issue 06
May 2017

AN EFFICIENT HIGH SECURED ARCHITECTURE FOR M-TURBO DECODERS USING S- BOX

CH.SPANDANA
M.TECH – SCHOLAR – E.C.E
Dept. of E.C.E
MALINENI LAKSHMAIAH WOMENS ENGENERING COLLEGE
GUNTUR DT.

K.SWETHA
ASSISTANT PROFESSOR
Dept. of E.C.E
MALINENI LAKSHMAIAH WOMENS ENGENERING COLLEGE
GUNTUR DT.

Abstract- To accelerate the majority logic decoding of difference set low density parity check codes the error detection in memory applications was proposed. This is useful as majority logic decoding can be implemented serially with simple hardware but requires a large decoding time. For memory applications, the increase of the memory access time takes place. S-Box Codes are the class of linear block codes which provide near capacity performance on large collection of data transmission channels.

I.INTRODUCTION

Now-a-days data communication has become essential for the life. During data transmission from source to receiver the errors may be introduced and transmitting channel is subjected to noise. So data must be transmitted with confidentiality and errors must be detected & corrected. There are different methods for implementing the secure communication. This project is an improvement for secure communication.

They were ignored for many years since they were thought to be impractical. But with present day technology they are very practical. They may have some implementation advantages if their performance is similar to turbo codes. The characteristic of their parity-check matrix which contains only a few 1's in comparison to the amount of 0's.They provide a performance which is very close to the capacity for a lot of different channels and linear time complex algorithms for decoding and it is the main advantage of it. LDPC codes have been defined pseudo-randomly. However, a random construction method also means that code properties are not guaranteed for any individual code, and are thus not easy to control or even determine. As well, implementation complexity is increased by the need to store, and encodes, codes described by random parity-check matrices.

II.LITERATURE SURVEY

Sum Product Algorithm: Efficient implementations of the sum-product algorithm (SPA) are presented for decoding low-density parity-check (LDPC) codes using log-likelihood ratios (LLR) as messages between symbol and parity-check nodes. The "decoding" algorithm for codes on graphs is the basic of the Sum product algorithm. For finite cycle-free graphs, it is finite and exact. However, because all its operations are local, it may also be applied to graphs with cycles; then it becomes iterative and approximate, but in coding applications it often works very well. For capacity-approaching codes it has become the standard decoding algorithm (e.g., LDPC codes, turbo codes). It operates in a distributed manner, with nodes in the graph exchanging statistical information via a sequence of "message-passing" updates. The general form of the forward-backward algorithm is the sum product algorithm.

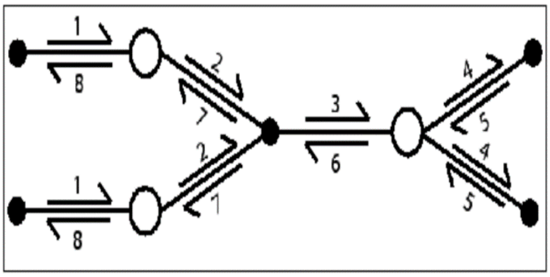


Fig. 1: Sum Product Message Passing Diagram

From some set of singly-connected function nodes to the variable nodes that they depend upon the first messages are passed. No computation is necessary in this step because the messages are simple identity messages that specify the apriority knowledge stored in the function that creates them. The neighbouring function node is connected to more than one variable is assumed; it must wait for all but one of its neighbours to send it a message. Once those messages are received, it creates a message of its own and passes that message along to the variable that did not send it a message. Iterative decoding of binary low-density parity-check (LDPC) codes using the sum-product algorithm (SPA) has recently been shown to approach the capacity of the additive white Gaussian noise (AWGN) channel within 0.0045 db. It has been shown that the direct implementation of original form of SPA is sensitive to quantization effects. An implication of the SPA that reduces the complexity of the parity check update at the cost of some loss in performance. This implication has been derived by operating in the log-likelihood domain. Log-likelihood ratios (LLR) are used as messages between symbol and parity check nodes. The family of LDPC decoding algorithms presented here is called LLRSPA. In particular, serial and parallel implementations are investigated, leading to trellis and tree topologies, respectively.

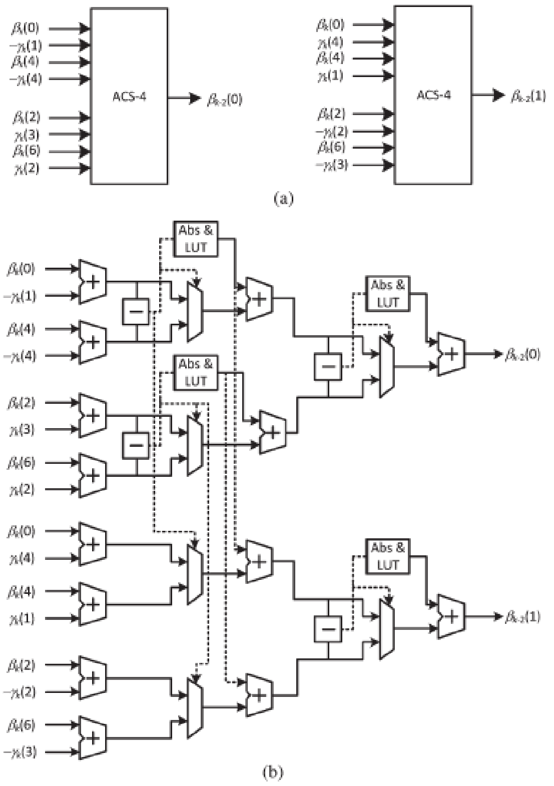


Fig.2. (a) Conventional radix-4 ACS architecture for two concurrent metrics computation. (b) Proposed radix-4 ACS architecture based on conventional architecture for two concurrent metrics computation.

To realize the $\beta k{-}2(0)$ value, each of the A and B values are proposed to be implemented by a radix-2 architecture, and finally, a third radix-2 architecture is used to achieve. The value of $\beta k{-}2(1)$ can be also similarly implemented as depicted. Radix-2 architecture employs a comparator and an LUT dealing with distances between two input values to select the maximum value, which then adds the selected amount to the maximum value. It is worth noting that the distance between two input values of (26) is $\beta k(0) - \beta k(4) + \gamma k(4) - \gamma k(1)$, which is equal to the distance between two input values of. The distances between each two input values of (27) and (29) are also equal. Therefore, the comparator and LUT units for the computation of (28) and (29) are omitted, leading to a novel architecture, as shown in Fig. 2(b). Hereafter, this proposed architecture is referred to as the maximum shared resource (MSR) architecture. This property is true for each pair of {(16), (17)}, {(18), (19)}, {(20), (21)} and {(22), (23)} for the backward recursion metrics and for each pair of {$\alpha(0)$, $\alpha(4)$}, {$\alpha(1)$, $\alpha(5)$}, {$\alpha(2)$, $\alpha(6)$} and {$\alpha(3)$, $\alpha(7)$} for the forward recursion metrics. In fact, using the proposed MSR architecture, the redundant computation is avoided, alleviating the area overhead in conventional schemes.

III.PROPOSED SYSTEM

In this extended version, we investigate the uniformity problem and the need for remasking in more detail. We prove that under certain circumstances, it is enough to remask only a fraction of the shares. Moreover, we argue that if there is enough remasking, we do not need to share functions uniformly. This observation helps us to further reduce the area and randomness requirements. We provide two new implementations.

The first one is similar to the one in, but it uses at least three shares in all the operations, including the linear ones. We use it to investigate the increase in security when moving from at least two to at least three shares, and to quantify the associated cost. The second implementation is based on the one in but modified according to our findings regarding uniformity and remasking.

Our three implementations need the same number of clock cycles to complete the calculation, and allow us therefore to focus on some trade-offs between area and additional randomness. For round operations and key schedule which requires only one S-box instance and loads the plaintext and key byte-wise in row-wise order we use a serial implementation.

The data unit consists of: the initial round of key addition and a final round. The need of the architecture of a standard round composed of both the transformation and the inverse transformation is for encryption and decryption respectively are performed using the same hardware resources. This implementation generates one set of sub key and reuses for calculating all other sub keys in real-time.

1. Byte Sub: In this architecture each block is replaced by the substitution in S-Box table consisting of the byte of the block.

2. Shift Row: In this transformation the rows of the block state are shifted over different offsets. The amount of shifts is determined by the block length. The shift row operation using combinational logic considering the offset by which a row should be shifted was implemented by proposed architecture.

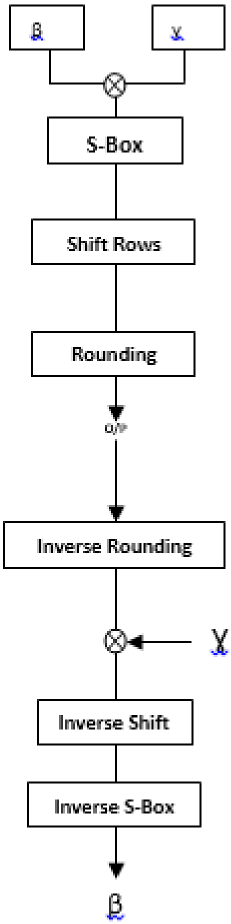
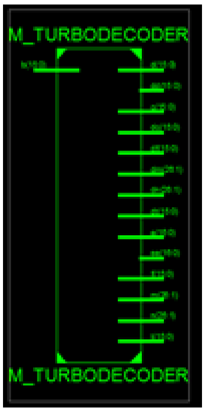


Fig. 3: Proposed System

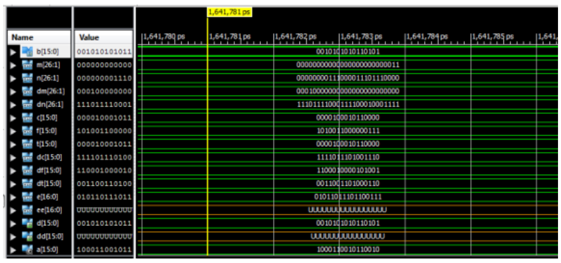
If this technique is used to protect cascaded functions, then extra measures like the binary data discussed in the previous section need to be taken, such that the input for the following nonlinear operation is again a uniform masking. A similar situation occurs when the technique is used to protect functional blocks acting in parallel on (partially) the same inputs.

IV.RESULTS

RTL SCHEMATIC:



OUTPUT WAVEFORM:



Name	Value	
b[15:0]	001010101011	0010101010110101
m[26:1]	000000000000	00000000000000000000000011
n[26:1]	000000001110	00000000111000011101110000
dm[26:1]	000100000000	00010000000000000000000000
dn[26:1]	111011110001	11101111000111100010001111
c[15:0]	000010001011	0000100010110000
f[15:0]	101001100000	1010011000000111
t[15:0]	000010001011	0000100010110000
dc[15:0]	111101110100	1111011101001110
df[15:0]	110001000010	1100010000101001
dt[15:0]	001100110100	0011001101000110
e[16:0]	010110111011	01011011101100111
ee[16:0]	UUUUUUUUUUUU	UUUUUUUUUUUUUUUUU
d[15:0]	001010101011	0010101010110101
dd[15:0]	UUUUUUUUUUUU	UUUUUUUUUUUUUUUU
a[15:0]	100011001011	1000110010110010

V.CONCLUSION

In this brief, the detection of errors during the first iterations of serial one step Majority Logic Decoding of S-Box codes has been studied. To reduce the decoding time by stopping the decoding process is the main objective when no errors are detected. The tested combinations of errors affecting up to four bits are detected in the first three iterations of decoding is showed by simulation results. These results extend the Ones recently presented for S-Box codes, making the modified one step majority logic decoding more attractive for memory applications. The larger choice of word lengths and error correction capabilities was now by designer.

VI.REFERENCE

[1] R. G. Gallager, "Low-density parity-check codes," IRE Trans. Inf. Theory, vol. IT-8, no. 1, pp. 21–28, Jan. 1962.

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

[3] D. J. MacKay, "Good error-correcting codes based on very sparse matrices," IEEE Trans. Inf. Theory, vol. 45, no. 2, pp. 399–432, Mar. 1999.

[4] T. J. Richardson and R. L. Urbanke, "The capacity of low density parity-check codes under message-passing decoding," IEEE Trans. Inf. Theory, vol. 47, no. 2, pp. 599– 618, Feb. 2001.

[5] R. M. Tanner, "A recursive approach to low complexity codes," IEEE Trans. Inf. Theory, vol. IT-27, no. 5, pp. 533– 547, Sep. 1981.

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

[7] Y. Kou, S. Lin, and M. P. C. Fossorier, "Low-density parity-check codes based on finite geometries: A rediscovery and new results," IEEE Trans. Inf. Theory, vol. 47, no. 7, pp. 2711–2736, Nov. 2001.

[8] G. Fettweis and H. Meyr, "Parallel Viterbi algorithm implementation: Breaking the ACS-bottleneck," IEEE Trans. Commun., vol. 37, no. 8, pp. 785–790, Aug. 1989.

[9] C. Studer, C. Benkeser, S. Belfanti, and Q. Huang, "Design and implementation of a parallel turbo-decoder ASIC for 3GPP-LTE," IEEE J. Solid-State Circuits, vol. 46, no. 1, pp. 8–17, Jan. 2011.

[10] Z. Wang, "High-speed recursion architectures for MAP-based turbo decoders," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 15, no. 4, pp. 470–474, Apr. 2007.

[11] C. Studer, S. Fateh, C. Benkeser, and Q. Huang, "Implementation tradeoffs of soft-input soft-output MAP decoders for convolutional codes," IEEE Trans. Circuits Syst. I, Reg. Papers, vol. 59, no. 11, pp. 2774–2783, Nov. 2012.

[12] M. Bickerstaff, L. Davis, C. Thomas, D. Garrett, and C. Nicol, "A 24 Mb/s radix-4 logMAP turbo decoder for 3GPP-HSDPA mobile wireless," in Proc. IEEE ISSCC Tech. Dig. Papers, 2003, vol. 1, pp. 150–484.

[13] S. Papaharalabos, P. Mathiopoulos, G. Masera, and M. Martina, "On optimal and near-optimal turbo decoding using generalized max operator," IEEE Commun. Let., vol. 13, no. 7, pp. 522–524, Jul. 2009.

[14] J.-F. Cheng and T. Ottosson, "Linearly approximated log-MAP algorithms for turbo decoding," in Proc. IEEE VTC—SpringTokyo, Japan, , May 2000, vol. 3, pp. 2252–2256.

[15] S. Talakoub, L. Sabeti, B. Shahrrava, and M. Ahmadi, "An improved maxlog-MAP algorithm for turbo decoding and turbo equalization," IEEE Trans. Instrum. Meas., vol. 56, no. 3, pp. 1058–1063, Jun. 2007.

CH.SPANDANA studied B.Tech in NRI Institute of Technology. At present, she is pursuing M.Tech in Malineni Lakshmaiah Women's Engineering College (MLWEC).Her area of interest is VLSI and Communications.

K.SWETHA studied B.Tech in Vignan Institute of Information Technology and M.Tech in JNTUH .At present she is working as assistant professor at Malineni Lakshmaiah Women's Engineering College with 7 years of experience. Her area of interest is Antennas.