International Journal of Engineering Technology Science and Research
IJETSR
www.ijetsr.com
ISSN 2394 – 3386
Volume 4, Issue 9
September 2017

Implementation of Power Efficient Parallel Chien Search Architecture using Modified Booth Encoding

I.Kanaka Durgamma
P.G Student VLSI Design (ECE), SVECW
K.Murthy Raju
Sr. Assistant Professor, Dept of ECE, SVECW

ABSTARCT*-The short horizontal Bose chaudari Hocquenghem (BCH) Chien search for signs of a new power-saving (CS) structure is proposed. For syndrome-based decoding, CS plays an important role in identifying the areas of error, but incurs a huge waste of exhaustive computation power consumption. The proposed architecture, the process of searching for the binary representation of the matrix is decomposed in two steps. This is neither new low power architecture for parallel CS provided. By reducing access to the second stage of the conventional CS to achieve significant power savings is decomposed in two steps. Error operate under the same ownership, the less energy the size of the CS in the construction sector in different configurations, and error correction capability of the horizontal factor compared to traditional construction. Power saving horizontal factor or increase the size of the field will become more and more important.Further this project is enhanced by replacing by multiplier architecture with radix8 modified booth multiplication algorithm for more power and area reduction. Radix4 modified booth encoding algorithm produces 50 percent reduction in partial products.*

KEYWORDS-Bose ChaudhurHocquenghem(BCH)codes,ChienSearch(CS),lowpower,two step approach,error correction code,low complexity.

I. INTRODUCTION

Among several error-correction codes used to improve corrupted code words in digital communication and storage systems, the Bose–Chaudhuri–Hocquenghem(BCH) code[1],[2] is one of the most conventional algebraic codes due to its powerful error-correction performance and low-cost hardware complexity. The binary BCH code has been employed in various systems such as advanced solid-state storages [3], [4]and optical fiber communication systems [5], and most of these applications are connected demanding ever higher decoding throughput and more larger error-correction capability. In general, decoding a BCH code that can correct t bits at maximum is composed of three major blocks,namely,syndrome calculation (SC), key-equation solver(KES), and Chien search (CS)[1], [2]. Given a received code word $R(x)$, the SC calculate $2t$ syndromes, and the KES produce the error locator polynomial $\Lambda(x)$ using the syndromes. Finally, error position $E(x)$ is calculated by locating roots of $\Lambda(x)$ based on the CS architecture.In a parallel BCH decoder, the CS is a important contributor to the power consumption and takes up to a half of overall power consumption [6]. More termination procedure presented in [6] and [7] are to remove redundant after finding the last error. In this brief, we present a new approach in which the parallel CS is decomposed into two steps. The first step is entering every cycle, but the second step is activated only when the first step is successful, producing in a less number of access. The proposed two-step approach is conceptually similar to that in [9]. Although the two-step approach, mostly , effect to the increase in critical path delay and latency, the disadvantages are proposed in this brief by employing an efficient pipelined structure.Otherwise the previous architectures [6]–[8], the proposed architecture can save the power consumption regardless of error locations.

II. PARALLEL CS ARCHITECTURE

Consider a binary BCH (n, k, t) code over GF($2m$), where n is the code length, k is the data length, and t is the maximal number of correctable error bits. More precisely, $n = k + mt$, where m is the field dimension that satisfies $2m - 1 \geq n$. When the syndrome-based decoding [1], [2], the error locator polynomial delivered by the KES is expressed as

International Journal of Engineering Technology Science and Research
IJETSR
www.ijetsr.com
ISSN 2394 – 3386
Volume 4, Issue 9
September 2017

$$\Lambda(x) = \sum_{j=1}^{t} \lambda \quad ^{j} + 1 = Y(x) + 1. \qquad (1)$$

In order to find the error position $E(x)$, the CS iteratively substitutes α^i into (1) for $1 \le i \le n$ and detects the presence of an error when $\Lambda(\alpha^i) = 0$ or $Y(\alpha^i) = 1$. In practice, p-parallel CS architecture is exactly implemented to achieve a high throughput, where the parallel factor p is the number of α^i substitutions performed at the same time.

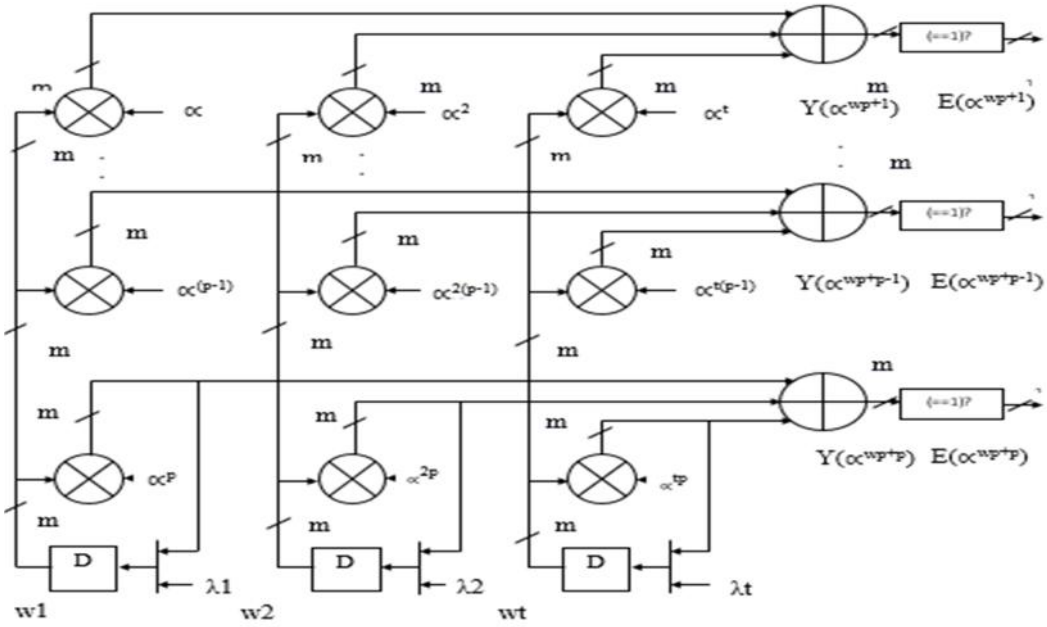

Fig 1: Conventional p-parallel Chien Search architecture using serial multiplier

As shown in Fig. 1, an intermediate value ωj in the jth registers is commonly find to p finite field multipliers (FFMs) are located in the same column. As a result, the p-parallel CS architecture is consists of pt FFMs, p t-input m-bit finite field adders, p m-bit comparators, t m-bit registers, and t m-bit multiplexers. Although all elements over GF(2^m) can be denoted as a $1\times m$ matrix, the computation in the CS can be defined by using the two matrices [10]–[12].

Fig. 1 determines the p-parallel CS architecture that reduces the number of cycles from n to $[n/p]$ by calculating

$$Y(\alpha^{wp+i}) = \sum_{j=1}^{t} \lambda j \alpha^{wpj} \alpha^{ij} = \sum_{j=1}^{t} wj(w)\alpha^{ij} \text{ for } 1 \le i \le p \qquad (2)$$

III. PROPOSED TWO-STEP CS ARCHITECTURE

The CS determines the presence of an error when $Y(\alpha^{wp+i})$ is 1, which indicates that α^{wp+i} is a part of the error locator polynomial. In the GF of dimension m, the multiplicative identity element, $\alpha^0$is defined as 1, i.e., $0(m-1{:}1)1(0)$, more precisely. The main idea comes from the fact that the absence of errors is guaranteed if some bits of $Y(\alpha^{wp+i})$ are not equal to those of $0(m-1{:}1)1(0)$. In the case of GF(2^4), for example, no presence of errors is guaranteed if $Y(\alpha^{wp+i})(3{:}2) \# 0$. Similar to [9], a two-step approach is employed for early detection. In other words, the possibility of error presence is found by searching only the l MSBs rather than the entire m bits.

Except the FFMs in the pth row, which is in fact used to update the registers, the two-step approach can be applied to the other FFMs in the p-parallel CS shown in Fig. 1. The two-step approach, in general, induces the longer critical path since one computation is decomposed into two small computations in series. To resolve the problem, the long critical path can be broken by inserting delay elements, which makes the two computations operate in a pipelined manner. Thus, the partial FFM for the LSBs is activated at the next clock

International Journal of Engineering Technology Science and Research
IJETSR
www.ijetsr.com
ISSN 2394 – 3386
Volume 4, Issue 9
September 2017

cycle only when the partial FFM for the MSBs results in zero. Since the intermediate values in the registers are updated every cycle, the straightforward pipelining method is to latch all the intermediate values into separate registers to provide them to the partial FFM for the LSBs at the next cycle.

However, this method demands a large amount of hardware resources. To prevent the increase in hardware complexity, in [9], the update of intermediate values was postponed when the former condition is satisfied. Hence, additional clock cycles are inevitable as one cycle is additionally taken whenever one of the $p - 1$ former computations is successful.

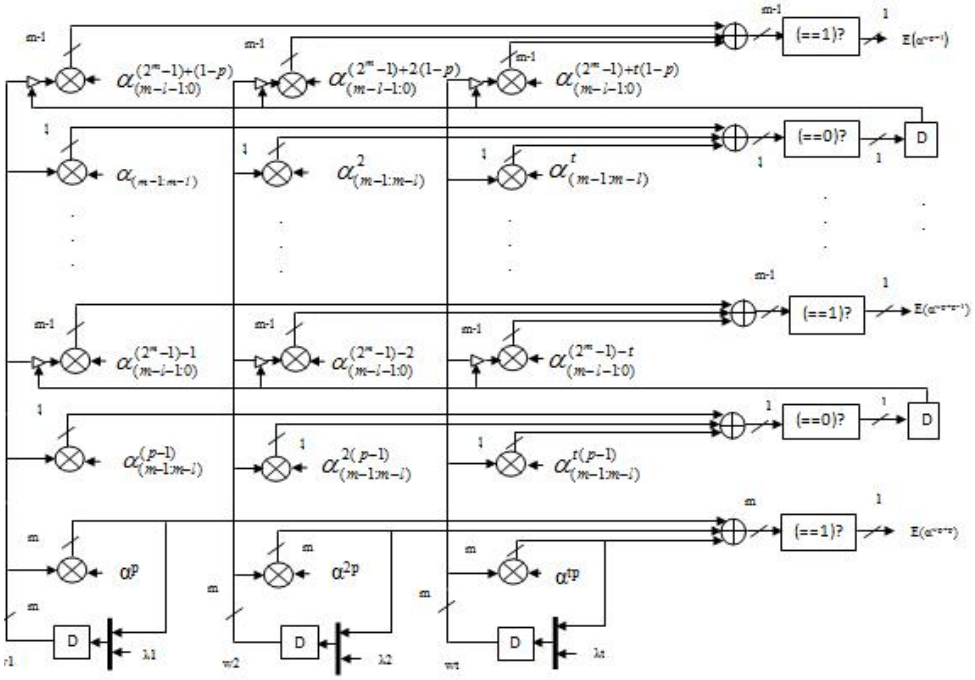

Fig 2: Two step architecture using Modified Booth multiplier for p-parallel CS

Fig.2 determines the low-power CS architecture based on the proposed two-step approach. In this architecture having multipliers,ripple carry adders,multiplexers,d-latches.the proposed architecture having modified booth multiplier.so the power can be decreased.Give the intermediate values from the register,the first partial FFM responsible for the remaining m-l LSB at the next clock cycle only when the output of the remainder is 0.then reduce the dynamic switching power by disabling the latter partial FFMS.So each elements are possible to intermediate register activated every 2^l clock cycles on the average power.

IV. SIMULATION RESULTS

XILINX suite design 14.7 is used for simulating the model in this by using VERILOG HDL code. Fig 3:shows the simulation result for 2x1 multiplexer ,in this having two inputs i0,i1 depending on selection line outputs are produced. Fig 4: shows the simulation result for 4-bit serial multiplier here multiplication operation is performed. Fig 5: shows the simulation result for Ripple carry adder based on c_{in} value get the final outputs . Fig 6: shows the simulation result for Modified Booth Multiplier, in this following algorithm to perform multiplication operation. . Fig 7: shows the simulation result for conventional Two-Step structure for Parallel Chien Search architecture using modified Booth multiplier ,where it gives eight outputs by adding two inputs.and Fig 8,Fig 9: shows the simulation result for The RTL Schematic of two-step structure for parallel CS using modified Booth multiplier and modified Booth algorithm .

International Journal of Engineering Technology Science and Research
IJETSR
www.ijetsr.com
ISSN 2394 – 3386
Volume 4, Issue 9
September 2017

Fig 3: Simulation results for 2x1 multiplexer

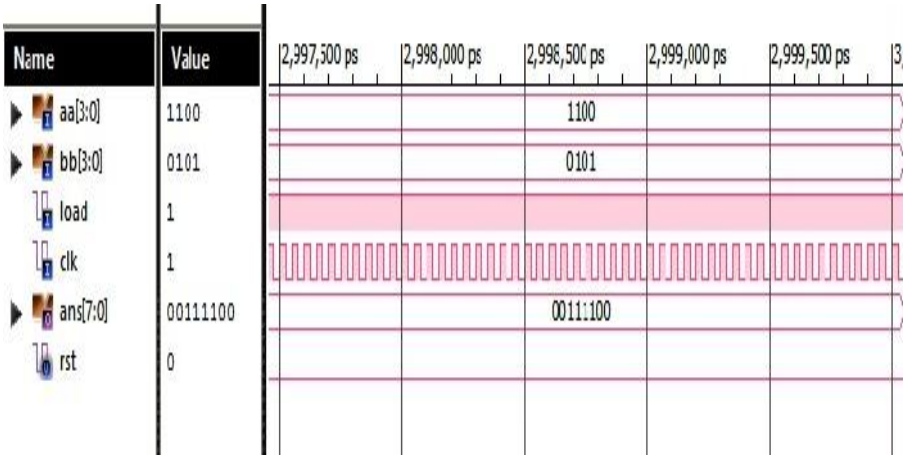

Fig 4: Simulation results for four bit serial multiplier

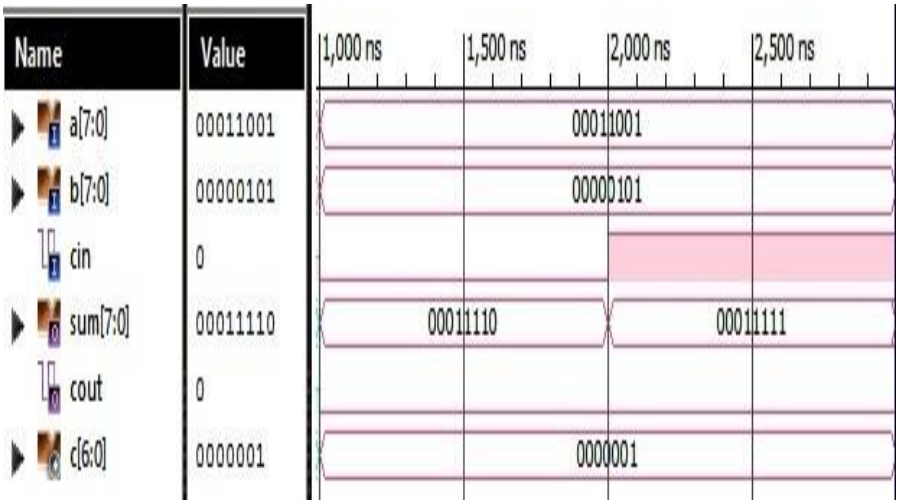

Fig 5: Simulation results for Ripple Carry Adder

International Journal of Engineering Technology Science and Research
IJETSR
www.ijetsr.com
ISSN 2394 – 3386
Volume 4, Issue 9
September 2017

Fig 6: Simulation results for Modified Booth multiplier

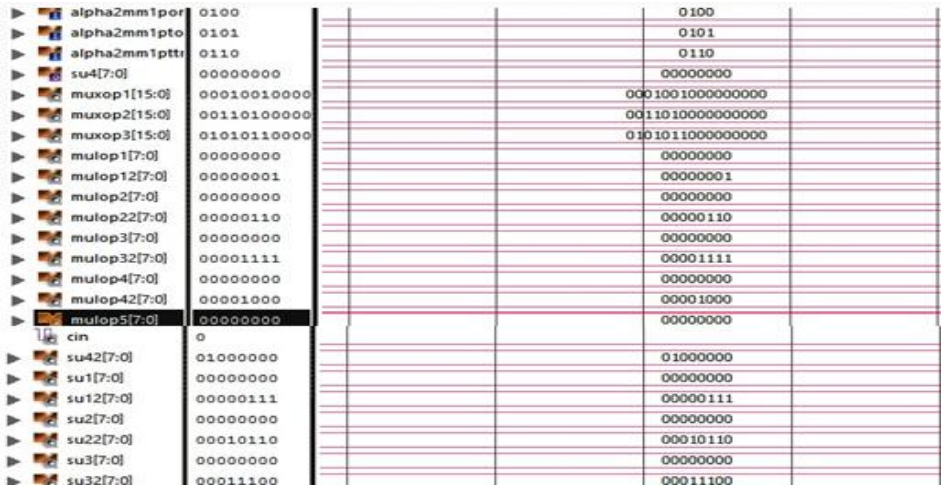

Fig 7: Two step architecture using Modified Booth multiplier for p-parallel CS

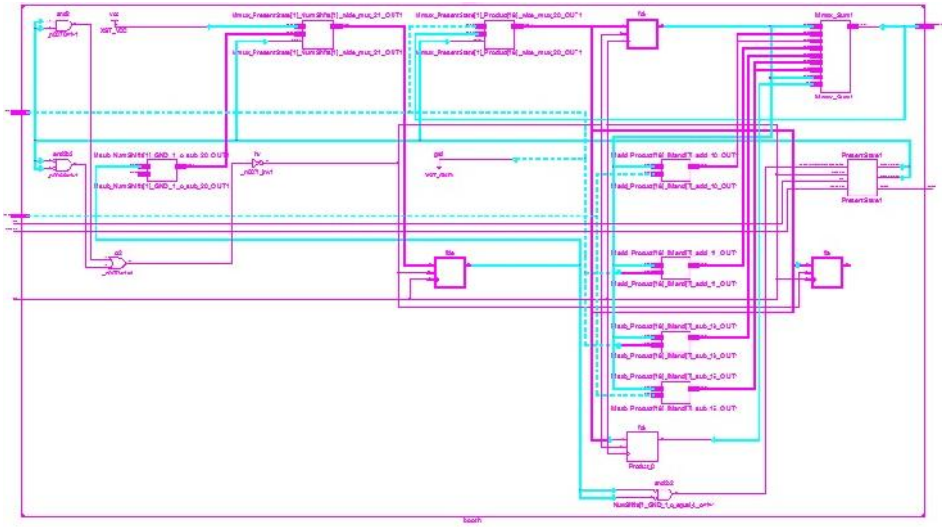

Fig 8: RTL Schematic for Modified Booth multiplier

International Journal of Engineering Technology Science and Research
IJETSR
www.ijetsr.com
ISSN 2394 – 3386
Volume 4, Issue 9
September 2017

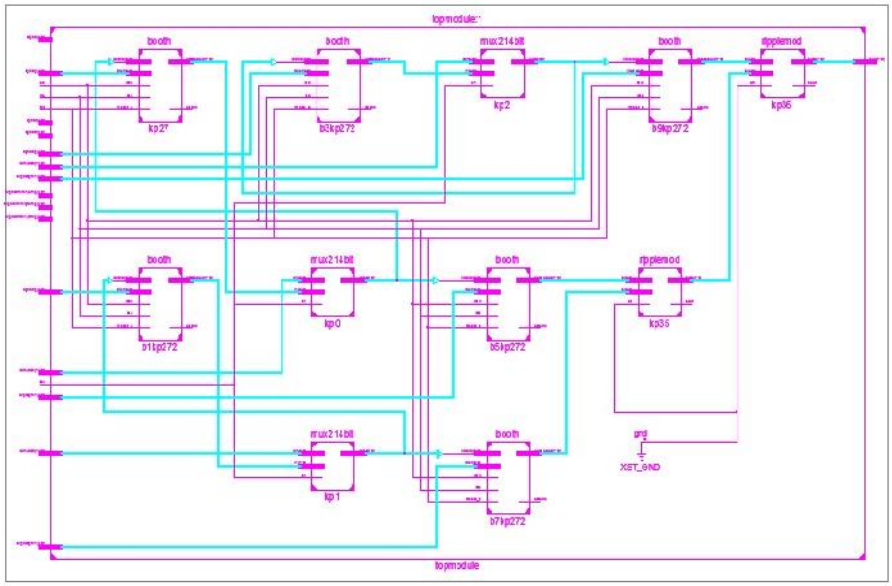

Fig 9: RTL Schematic for Two step architecture using Modified Booth multiplier for p-parallel CS

Table 1. Power comparison table for Two step architecture using for p-parallel CS and Two step architecture using Modified Booth multiplier for p- parallel CS

Power report of the system	Power in watts
Two step architecture using serial multiplier for p-parallel CS	2.043
Two step architecture using Modified Booth multiplier for p-parallel CS	1.455

V. CONCLUSION

This is a new low-power architecture for parallel CS provided. By reducing access to the second stage of the conventional CS to achieve significant power savings is decomposed in two steps. Error operate under the same ownership, the less energy the size of the CS in the construction sector in different configurations, and error-correction capability of the horizontal factor compared to traditional construction. Final implementation with Radix 8 modified booth encoding algorithm yields reduction in density and power consumption

REFERENCES

[1] Y. Lee, H. Yoo, and I.-C. Park, "High-throughput and low-complexity BCH architecture for solid- state drives," IEEE Trans. Very Large Scale Integr. Syst., vol. 22, no. 5, pp. 1183–1187, May 2014.
[2] Y. Lin, C. Yang, C. Hsu, H. Chang, and C. Lee, "A MPCN-based parallel architecture in BCH decoders for NAND Flash memory devices," IEEE Trans. Circuits Syst. II, Exp. Briefs, vol. 58, no. 10, pp. 682– 686,Oct. 2011.
[3] Y. Lee, H. Yoo, and I.-C. Park, "Low-complexity parallel Chien search structure using two-dimensional optimization," IEEE Trans. Circuits Syst. II, Exp. Briefs, vol. 58, no. 8, pp. 522–526, Aug. 2011.
[4] J. Cho and W. Sung, "Strength-reduced parallel Chien search architecture for strong BCH codes," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 55, no. 5, pp. 427–431, May 2008.
[5] S. Wong, C. Chen, and Q. M. Wu, "Low power Chien search for BCH decoder using RT-level power management," IEEE Trans. Very Large Scale Integr. Syst., vol. 19, no. 2, pp. 338–341, Feb. 2011.
[6] Y. Wu, "Low power decoding of BCH codes," in *Proc. IEEE ISCAS*, May 2004, pp. II-369–II-372.

[7] Y. Chen and K. K. Parhi, "Small area parallel Chien search architectures for long BCH codes," *IEEE Trans. Very Large Scale Integr. Syst.*, vol. 12, no. 5, pp. 545–549 May 2004.
[8] H. Weingarten, E. Sterin, O. A. Kanter, and M. Katz, "Low Power Chien-Search Based BCH/RS Decoding System for Flash Memory, Mobile Communications Devices and Other Applications," U.S. Patent 2010 013 1831 A1, May 27, 2010.
[9] H. Choi, W. Liu, and W. Sung, "VLSI implementation of BCH error correction for multilevel cell NAND Flash memory," *IEEE Trans. Very Large Scale Integr. Syst.*, vol. 18, no. 5, pp. 843–847, May 2010.