Low Power Chien Search for BCH Decoder Using RT-Level Power Management

Shu-Yi Wong, Chunhong Chen, and Q. M. Jonathan Wu

***Abstract*—As a major contributor to the Bose–Chaudhuri–Hocquenghem (BCH) decoder's power consumption, Chien search is a critical step in the binary BCH decoding process for many portable applications. This paper proposes a new low-power design strategy by applying register transfer level (RTL) power management with significant power savings. The proposed Chien search is implemented for a (255, 187, 9) code in CMOS 0.18-μm technology, and simulations show 34% power improvement over the conventional method.**

***Index Terms*—Bose–Chaudhuri–Hocquenghem (BCH) decoder, Chien search, low power, power management.**

I. Introduction

Binary Bose–Chaudhuri–Hocquenghem (BCH) code is a popular forward-error correcting code over Galois-field GF(2^m) in the form of (n, k, t), where $n = 2^m - 1$ is the code-word length for some positive integer m, k is the source message length, and t stands for the error-correcting capability in bits. The final step of syndrome-based decoding involves solving an error locator polynomial, as shown in (1), which is commonly implemented by Chien search

$$\begin{cases} \Lambda_t(R_i) = 1 + \sum_{i=1}^{t} R_i \\ \text{where } R_i = \sigma_i X^i \quad \text{and} \quad X = \alpha^z \end{cases} \tag{1}$$

where α is the primitive element of GF(2^m), σ_i's are the coefficients of the polynomial, and R_i represents the registers holding results of multiplication. Fig. 1 (solid-line portion) shows a register-transfer level (RTL) architecture for the serial Chien search [1], which is referred to as the conventional Chien search throughout this paper. The Chien search involves an exhaustive linear search for all possible error positions, and an error is found when $\Lambda_t(R_i) = 0$.

Power-efficient Chien search is important for many applications [2], especially for a large value of t. Existing low-power methods seek power savings by disabling the circuit either when the decoder detects no error [3] or after the last error is found [4]. These methods are effective only for a small value of t when considering the decoder's input error distribution (see Section II-B for details). As the probability of early shutdown decreases with the increasing number of errors, so does the potential power reduction. Parallel Chien search is another low-power strategy but with serious area penalty [5].

This paper presents a novel approach based on RTL power management with the goal of achieving much power savings for large value of t. The main contributions of this work include: 1) developing an approximated model for power estimation; 2) proposing a new circuit architecture that promises better power performance than [4] and less area cost than [5] in BCH codes with higher error correcting capability; and 3) implementing the proposed Chien search for a (255, 187, 9) BCH code with CMOS 0.18-μm technology. The remainder of this paper is organized as follows. Section II focuses on power modeling along with the proposed low-power approach. Section III describes the circuit design and implementation. Simulation results are presented in Section IV and Section V concludes this paper.

Manuscript received May 20, 2009; revised September 14, 2009. First published October 23, 2009; current version published January 21, 2011. This work was supported in part by the AUTO21 Network of Centers of Excellence, Canada, under Grant FC302-FSC.

The authors are with the Department of Electrical and Computer Engineering, University of Windsor, Windsor, ON N9B 3P4, Canada (e-mail: wong11j@uwindsor.ca; cchen@uwindsor.ca; jwu@uwindsor.ca).

Digital Object Identifier 10.1109/TVLSI.2009.2033698

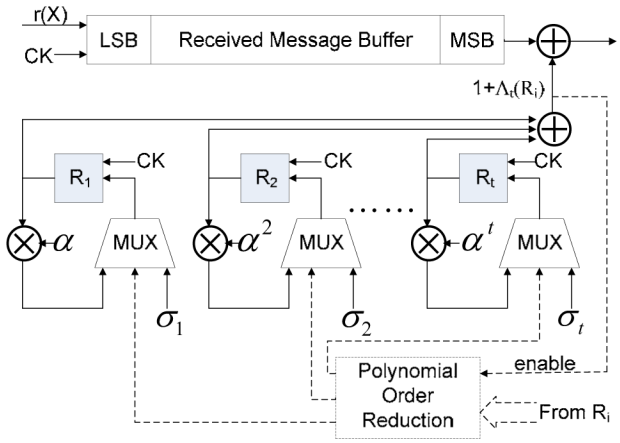

Fig. 1. RTL architecture for Chien search.

II. RTL Power Management for Chien Search

This section begins with a power model for both conventional Chien search and the method of [4]. This model also serves as a vehicle for understanding the potential power efficiency of the proposed power-management method.

A. Power Modeling for Existing Methods

From Fig. 1 (solid-line portion), the conventional Chien search with error correcting capability of t bits includes t identical stages. Each stage corresponds to one of the R_i terms in (1). Therefore, the total average power can be expressed as

$$P_C = \sum_{i=1}^{t} [P_{\text{cgfm}}(i) + P_{\text{reg}}(i) + P_{\text{mux}}(i) + P_{\text{add}}(i)] + P_{\text{ckt}} \tag{2}$$

where $P_{\text{cgfm}}(i), P_{\text{reg}}(i), P_{\text{mux}}(i), P_{\text{add}}(i)$, and P_{ckt} represent the average power consumption due to Constant Galois-field multipliers (CGFMs), registers, multiplexors, adders and the clock-tree, respectively, with i being a dummy variable pointing to the ith order term of (1).

Equation (2) assumes an error locator polynomial of tth order, which, in reality, depends on the number of errors received. Therefore, P_C is not a fixed value. Nevertheless, the total average power can be found by using standard error distribution models such as AWGN (additive white Gaussian noise), as will be discussed in Section II-B.

On the other hand, we seek to provide power analysis for (2) at RTL. This is done by making two assumptions: 1) the power dissipation of Chien search comes mainly from CGFMs and registers and 2) the CGFM and register for each i consume almost the same amount of power. The first assumption is valid as the power dominance of registers has been observed in similar applications [3]. The second assumption is based on the fact that all registers' average power consumption is close to one another, as the Chien search process offers each of the registers with chance of holding every possible finite-field element. A register will consume more power when it is holding values. This can be modeled by the active power of a register (P_{ra}). Even when a register holds a zero value, it still consumes power. We denote this power as the idle power of a register, which equals a constant η times P_{ra} where η is close to one (see Section IV).

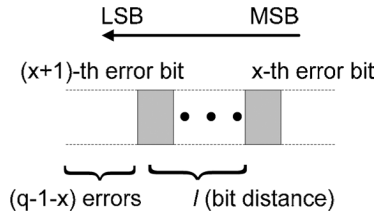

Fig. 2. Power estimation by determining the bit distance l.

To get quantitative insight into the power savings of [4], let us consider a q-errors corrupted message word. For a (n, k, t) BCH decoder with no decoding failure, the Chien search may receive, from the previous stage, any qth order error polynomial, where q may be smaller than t. Since the conventional Chien search always proceeds in one direction, i.e., from the most significant bit to the least significant bit, and takes one clock cycle to progress from one bit to the next, we can introduce the concept of *bit distance* $l(1 \le l \le n - q + 1)$ to denote the number of clock cycles between the xth and the $(x+1)$th error bit, as shown in Fig. 2, where x represents any arbitrary integer from 0 to $q-1$ (the 0-th error bit represents the beginning of a message). With the assumption of equal bit error rate for all bits, we may assign, for each value of l, the number of equally-probable error-patterns, $N_{l,x}$, which possess a bit distance l between the xth and the $(x+1)$th error bit. $N_{l,x}$ is easily determined as follows if we define another variable b as the number of bit gaps between errors outside of the xth and $(x+1)$th errors (where $b = n - q + 1 - l$):

$$N_{l,x} = \binom{q-1+b}{b}. \tag{3}$$

This indicates that $N_{l,x}$ is independent of x, as is the mean of l. The number of clock cycles (denoted by d) counted down from the final error bit should also be determined for modeling [4]. With (3), the mean of both l and d can be calculated as (without proof)

$$\bar{l} = \frac{n+1}{q+1}, \quad \bar{d} = \frac{n-q}{q+1}. \tag{4}$$

For [4], the ratio of $\bar{d}$ to n represents the fractional time the Chien search is turned off, and is hence proportional to the power savings. Thus, the average power savings (normalized to the power for the conventional Chien search) for an n-bit received message carrying q error bits can be estimated as

$$S_q = \frac{\frac{\bar{d}}{n} \cdot P_C}{P_C} = \frac{1 - \frac{q}{n}}{q+1}. \tag{5}$$

Since [4] assumes an equal probability model where the likelihood of receiving q-errors corrupted frames is the same for any value of q, the average power savings of this existing method are given by (with the assumption of $\eta = 1$)

$$\begin{aligned} \bar{S}_{\text{existing}} &= \frac{1}{t}\sum_{q=1}^{t} S_q = \frac{1}{t}\sum_{q=1}^{t} \frac{1-\frac{q}{n}}{q+1} \\ &\approx \frac{1}{t}\sum_{q=1}^{t} \frac{1}{q+1}, \quad \text{if } n \gg t. \end{aligned} \tag{6}$$

However, using the AWGN model may result in unequal probabilities for q-errors corrupted frames. More discussions will be given in the section that follows.

B. Proposed Method

By rewriting (1) as

$$\Lambda_t(X) = (1 + \beta_1 X)(1 + \beta_2 X) \ldots (1 + \beta_t X) \tag{7}$$

we realize that Chien search is equivalent to the process of solving for all the values of β_1 through β_t. Each time a particular error bit is found, the corresponding factor becomes redundant. This means that the CGFMs in Fig. 1 can be made redundant and then disabled with the method of clock-gating by taking out from (7) one polynomial factor at a time. This allows for additional power savings prior to the eventual power-down of the entire Chien search at the final error-bit.

To model the proposed method, we only take into consideration the $P_{\text{cgfm}}(q)$ and $P_{\text{reg}}(q)$ in (2) for reasons already explained in Section II-A. In general, $P_{\text{cgfm}}(q)$ increases with q due to the higher circuit complexity of a finite-field element being multiplied by α^q. For a qth order error polynomial, the q CGFMs are initially active, but only $v(1 \le v \le q)$ CGFMs will stay active as our proposed method progresses. If the clock gating is perfect, when v CGFMs are active, the total power (i.e., the power from both CGFMs and registers) is given by [refer to (2)]

$$p(v) = v \cdot (P_{\text{cgfm}} + P_{\text{reg}}). \tag{8}$$

Since the higher-order CGFMs with potentially more power consumption are the first to be turned off according to the proposed method, the constant components in (8) (P_{cgfm} and P_{reg}) may only result in an underestimate of the average power. However, at RTL where the details of circuit implementation are not available, this approximation is reasonable, particularly when the value of η is close to one (indicating the dominance of registers' idle-mode power), as will be shown in Section IV.

For n-bits and q-errors messages, if all q CGFMs are active initially and turned off one after another subsequently, the average power can be obtained from (4) and (8) as (for $n \gg 1$)

$$\bar{p}_{\text{prop}}(v) = \frac{n+1}{n(v+1)} \cdot \sum_{q=1}^{v} p(q) \approx \frac{v}{2} \cdot (P_{\text{cgfm}} + P_{\text{reg}}). \tag{9}$$

Combining (8) and (9) gives

$$\bar{p}_{\text{prop}}(q) \approx \frac{1}{2} p(q). \tag{10}$$

Based on the definition of $P_{\text{reg}}(q)$ (see Section II-A) and (8), the total power of the conventional Chien search [again, only including the first two terms in (2)] is approximated as

$$p_{\text{conv}}(q) = p(q) + (t - q)\eta P_{\text{ra}}. \tag{11}$$

Unlike the conventional Chien search, the proposed method introduces q extra states, and the decoding time becomes $n + q$ (instead of n) clock cycles. Thus, the control logic needs to be modified, but with minimal power implication, as will be shown later in Section IV. To maintain the same throughput, the clock frequency can be increased by a factor of q/n. Since q is usually much smaller than n, the extra power will not be significant compared to other power components. A simple model, which assumes that the power consumption due to an extra state is the same as that for processing a normal error-free bit, should suffice. Thus, the extra increase in power can simply be modeled by a factor of q/n. With the equal-probability assumption, the average power consumption of the proposed method (normalized to that of the conventional Chien search) can be written as

$$\bar{P}_{\text{prop}} = \frac{0.5 \times \sum_{q=1}^{t} \left[\left(1 + \frac{q}{n}\right) \cdot p(q)\right]}{\sum_{q=1}^{t} p_{\text{conv}}(q)} \tag{12}$$

where $p(q)$ and $p_{\text{conv}}(q)$ are given by (8) and (11), respectively. Generally speaking, (12) requires the detailed component power data from the circuit. However, the fact that the P_{ra} term is a dominant component in (8) allows a quick estimation of power savings by discarding

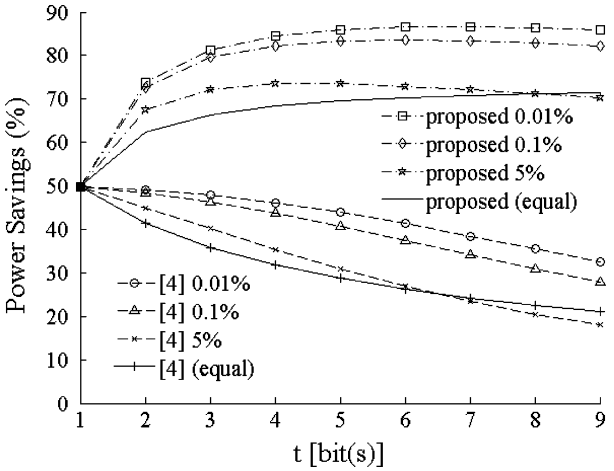

Fig. 3. Power savings for both [4] and the proposed method for a variety of percentages of uncorrectable frames.

other insignificant terms, leading to the final power savings of $\bar{S}_{\rm prop} = 1 - \bar{P}_{\rm prop}$ which is calculated approximately as

$$\bar{S}_{\rm prop} \approx 1 - \left[\frac{6n}{3n+2t+1} \cdot \left(1 + \frac{t-1}{t+1} \cdot \eta\right)\right]^{-1}. \quad (13)$$

The previous expression generally promises much better results than (6). For instance, assuming $\eta = 1$, (13) produces the power savings of about 50% for $t = 1$ and 72% for $t = 9$, compared to only 21% given by (6) for $t = 9$ (assuming $n \gg t$).

With the AWGN model, if p_e represents the bit error rate, the word probability (p_w) of q-errors messages is given by

$$p_w(q, p_e) = \binom{n}{q} \cdot p_e^q \cdot (1 - p_e)^{n-q}. \quad (14)$$

The average power consumption (denoted by $P_{\rm AWGN}(t)$) of the proposed method can be modified by adjusting (12) as

$$P_{\rm AWGN}(t) = \frac{\sum_{q=1}^{t} \left\{\left(1 + \frac{q}{n}\right) \cdot p_w(q, p_e) \cdot p(q)\right\}}{2 \cdot \sum_{q=1}^{t} \left\{p_w(q, p_e) \cdot p_{\rm conv}(q)\right\}} \quad (15)$$

where $p(q), p_{\rm conv}(q)$ and $p_w(q, p_e)$ are given by (8), (11) and (14), respectively. To provide a fair comparison, a higher-t circuit is given a lower signal-to-noise ratio (SNR) such that the bit error rate (p_e) becomes a function of t as the probabilities of receiving uncorrectable frames are assumed to be equal for all values of t. For instance, according to (14), with the assumption of 0.1% of uncorrectable frames, 95% of received frames are error-free for $t = 1$ and the number drops below 50% for $t = 4$. Fig. 3 shows the comparison of the AWGN (shown as dotted lines) and equal-probability (shown as solid lines) models during the error correction for a variety of percentages of uncorrectable frames. It can be seen from Fig. 3 that the AWGN model generally results in more power savings. However, at 5% of uncorrected frames, the power savings for both models are very close to each other (only a few percent different). Thus, the equal-probability model given by (6) and (13) can be used to provide the worst-case power estimation when the decoder is driven close to the edge of its error correcting capability.

C. Polynomial-Order-Reduction (POR) Algorithm

Since the proposed method seeks to reduce the order of $\Lambda_t(R_i)$, the variable i no longer represents a fixed list. To distinguish this change, a variable $w(w = 1, 2, \ldots, u)$ is introduced, where u is the current order of the error polynomial, leading to the following:

$$\begin{cases} \Lambda_u(R_w) = 1 + \sum_{w=1}^{u} R_w \\ R_w = \sigma_w(\varepsilon) X^w \end{cases}. \quad (16)$$

where the coefficients of X are functions of ε, and $\varepsilon(\varepsilon = 1, 2, \ldots, t)$ corresponds to the number of polynomial order reductions already performed. For instance, $\sigma_w(0)$ represents the initial value of coefficients when $u = t$, and $\sigma_w(t)$ represents zero when $u = 0$ (i.e., all roots are factored out). Assuming that the Chien search finds an error position β, it is always possible to break the expression of $\sigma_w(\varepsilon)$ into two groups of terms

$$\sigma_w(\varepsilon) = \gamma_w(\varepsilon) + \rho_w(\varepsilon) \quad (17)$$

where $\gamma_w(\varepsilon)$ is the sum of all β-carrying product terms, and $\rho_w(\varepsilon)$ the sum of all product terms that are without β. Since $\gamma_w(\varepsilon)$ always contains a common factor β, what remains (after extracting the β) is the sum of product terms from $\binom{u-1}{w-1}$ combinations of any error positions except β, which is exactly the same as $\rho_{w-1}(\varepsilon)$ that also carries the sum of product terms from same combinations. Therefore, we have

$$\gamma_w(\varepsilon) = \beta \rho_{w-1}(\varepsilon). \quad (18)$$

The goal of POR is to find the update value $R'_{\rm w}$ for $R_{\rm w}$. To this end, the coefficients of X, i.e., $\sigma_w(\varepsilon + 1)$, are required according to (16). Also, the order of the polynomial is updated by the process from u to $u' = u - 1$. The value of $\sigma_w(\varepsilon + 1)$ is the sum of product terms from $\binom{u'}{w}$ combinations, and is given by

$$\sigma_w(\varepsilon + 1) = \gamma_{w+1}(\varepsilon) \cdot \frac{1}{\beta}. \quad (19)$$

By combining (16)–(19), we can prove

$$R'_w = \sum_{\lambda=w+1}^{u} R_\lambda \quad (20)$$

which means that the update value for the wth order register is simply the sum of all higher-order register values. The highest order register should be updated with zero and become redundant. During this updating process, the order of $\Lambda_u(R_i)$ in (16) is reduced by one at a time, eventually bringing the polynomial to zero after all errors are detected. Fig. 1 illustrates the proposed Chien search architecture with the POR (dotted-line portion).

III. Circuit Design

This section deals with some circuit design problems so that the power and area overhead can be kept at a minimum level.

A. Adder for POR

Since (1) contains all intermediate summation terms for satisfying (20), the POR can potentially be implemented using the same adder in the conventional Chien search circuit without much area overhead. However, the required additional outputs from these summation terms will incur significant power overhead. To minimize the negative power impact, an additional adder was used instead for our design. Fig. 4 shows the new adder block for the POR, which has outputs for driving the multiplexors with other CGFM stages (see the dotted lines of Fig. 1). While this causes an increase in area, the new adder is gated

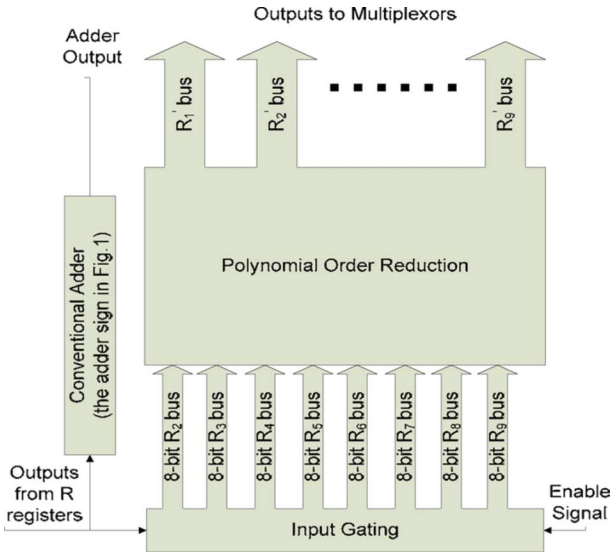

Fig. 4. Adder block for POR.

by a low-load input buffer which activates the circuit only for t out of n clock cycles. Considering the fact that n is usually much greater than t, the extra power consumption will be minimal.

B. Layout and Area Overhead

To verify the effectiveness of our method, a (255, 187, 9) BCH code was implemented for both conventional and proposed Chien search with CMOS 0.18-μm technology. *Synopsys Design Vision* and *Cadence Encounter* were used for logic synthesis and layout design, respectively. Overall area of the proposed circuit increases by 24% to 0.041 mm^2 due to the POR, compared to 0.033 mm^2 for the conventional circuit and the 106% area penalty for the two-fold parallel circuit [5].

C. Parallel Chien Search and Power Overhead

By employing similar approaches from [3], two-fold parallelism may be achieved using the proposed method with a pair of PORs, multiplexors and CGFMs for each polynomial term of (1), resulting in nearly the same area overhead of 24%. Since the circuit searches twice as fast with the same number of extra states, the equal-probability power overhead (on average) is approximately t/n (see Section II-B), which is about 4% for the (255, 187, 9) code. BCH decoder interleaving [3] is another way of achieving high throughput with multiple independent Chien search circuits. For such a case, the proposed method can be applied directly with the same area overhead of 24% and the power overhead of roughly $t/2n$ (or 2% for the above code), independent of the level of interleaving.

IV. Simulation Results

Both conventional and proposed Chien search circuits for the (255, 187, 9) BCH code were simulated with volumes of random q-errors $(q = 1, 2, \ldots, 9)$ messages. Our results show that with the equal-probability model, combination of CGFMs and registers consumes only 56% of total power in the conventional Chien search. The power savings for the proposed search circuit are shown in Fig. 5 (top curve), from which we see 60% power reduction at $t = 9$. In comparison with (13) (refer to the top solid curve in Fig. 3), there is only a slight discrepancy which gradually expands to 12% at $t = 9$. This is due to non-unity $\eta(0.62)$ and the neglected terms in deriving (13). It can also be seen from Fig. 3 that the results for the AWGN model are generally better due to the higher percentage of frames being received with fewer errors. Similarly, a relative improvement can be expected over the power (both the solid-line and circled curves of Fig. 5) for the performance of the proposed circuit under AWGN.

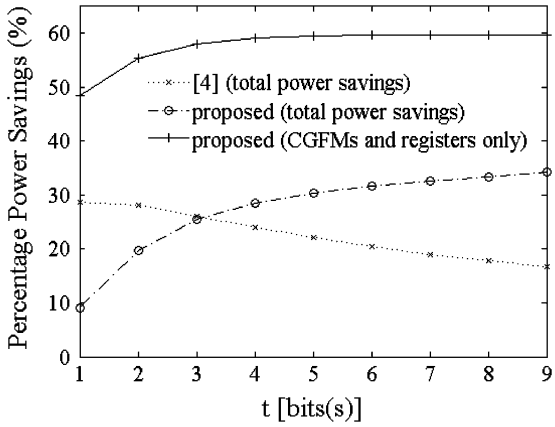

Fig. 5. Power savings for $(255, k, t)$ BCH codes.

In addition to power reduction from CGFMs and registers, there are also moderate power savings in the multiplexors and adders. However, the power in clock-tree increases due to the power overhead from latches and buffers. The control logic only consumes small amount of power. Mainly due to the increased power consumption in clock-tree, the overall power savings of the proposed method decrease to 34% for $t = 9$, as shown in Fig. 5.

The power savings of [4] are also shown in Fig. 5 along with the results from the proposed method for comparison. Generally speaking, the proposed method starts to outperform [4] when $t = 4$, and its advantage becomes increasingly obvious with further rise of t. At $t = 9$, in particular, the power savings of the proposed circuit nearly double that of [4].

V. Conclusion

We have presented a novel method in applying RTL power management to the design of low-power Chien search circuit. The method gradually disables CGFMs by clock gating and yields more power savings than the existing approach which keeps CGFMs active until all errors have been detected. The proposed Chien search has been implemented for a (255,187, 9) code with CMOS 0.18-μm technology. Experimental results have shown that with equal probability of having one to nine error-bits in messages, the proposed method provides 34% improvement over the conventional Chien search in terms of total power, and up to 60% power reduction when only power-dominant components (CGFMs and registers) are considered.

References

[1] R. T. Chien, "Cyclic decoding procedure for the Bose-Chaudhuri-Hocquenghem codes," *IEEE Trans. Inf. Theory*, vol. IT-10, no. 4, pp. 357–363, Oct. 1964.

[2] S. Y. Wong, C. Chen, and Q. M. J. Wu, "Power-management-based chien search for low power BCH decoder," in *Proc. 14th ACM/IEEE Int. Symp. Low Power Electron. Des. (ISLPED)*, Aug. 2009, pp. 299–302.

[3] L. Song, M. Yu, and M. S. Shaffer, "10- and 40-Gb/s forward error correction devices for optical communications," *IEEE J. Solid-State Circuits*, vol. 37, no. 11, pp. 1565–1573, Nov. 2002.

[4] Y. Wu, "Low power decoding of BCH codes," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, May 2004, vol. 2, pp. II-369–372.

[5] A. Raghupathy and K. J. R. Liu, "Algorithm-based low-power/high-speed Reed-Solomon decoder design," *IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process.*, vol. 47, no. 11, pp. 1254–1270, Nov. 2000.