This paper presents a two-iteration concatenated Bose-
Chaudhuri-Hocquenghem (BCH) code and its high-speed
low-complexity two-parallel decoder architecture for 100
Gb/s optical communications. The proposed architecture
features a very high data processing rate as well as excellent
error correction capability. A low-complexity syndrome
computation architecture and a high-speed dual-processing
pipelined simplified inverseless Berlekamp-Massey (Dual-
pSiBM) key equation solver architecture were applied to the
proposed concatenated BCH decoder with an aim of
implementing a high-speed low-complexity decoder
architecture. The proposed two-iteration concatenated BCH
code structure with block interleaving methods allows the
decoder to achieve 8.91dB of net coding gain performance
at 10^{-15} decoder output bit error rate to compensate for
serious transmission quality degradation. Thus, it has
potential applications in next generation forward error
correction schemes for 100 Gb/s optical communications.

Index Terms— 100G, BCH, concatenated codes, decoder,
FEC, low complexity, optical communications

1. INTRODUCTION

The Bose-Chaudhuri-Hocquenghem (BCH) codes are a
class of powerful multiple error-correcting cyclic codes [1].
The BCH codes are used in a broad class of error correcting
codes such as optical fiber communication systems, second
generation Digital Video Broadcasting (DVB-S2) and
digital communication systems.

The Reed-Solomon (RS) (255,239) code has been used and
standardized in the International Telecommunication
Union Telecommunication Standardization Sector (ITU-T)
G.975 and G.709 [2]. This code has a net coding gain
(NCG) of 6.2dB at a 10^{-15} decoder output bit error rate
(BER) with 6.69% redundancy. However, for high-speed
(40 Gb/s and beyond) optical fiber communication systems,
more powerful forward error correction (FEC) codes have
become necessary in order to achieve higher correction
ability than the RS(255, 239) code and compensate for
serious transmission quality degradation. Thus, several
Super-FEC schemes are considered and recommended in
the ITU-T G.975.1 recommendations [2].

Furthermore, the standardization of a hard-decision
FEC, which allows the redundancy ratio up to 7%, for a 100
Gb/s optical channel transport unit 4 (OTU4) is under
discussion at the ITU-T. As a result, the RS(255,239) code
has become mandatory for short-reach systems. However,
no specific FEC was determined as a standard for metro and
long-haul systems, although several candidates proposed
their own FEC codes.

In this paper, we propose a two-iteration concatenated
BCH code and its high-speed low-complexity two-parallel
decoder architecture for 100 Gb/s optical communication
systems. Also, a low-complexity syndrome computation
(SC) architecture and a novel Dual-processing pipelined
Simplified inverseless Berlekamp-Massey (Dual-pSiBM)
key equation solver (KES) architecture are proposed with
the aim of reducing the hardware complexity and improving
the clock frequency. The rest of this paper is organized as
follows.

In Section 2, we propose a two-iteration concatenated
BCH code scheme with two-parallel processing, and
Section 3 shows the proposed high-speed low-complexity
concatenated BCH decoder architecture. In Section 4, we
present implementation results and performance
comparisons. Finally, we provide conclusions in Section 5.

2. PROPOSED CONCATENATED BCH CODE

2.1. Conventional three-iteration concatenated BCH

code

The conventional concatenated BCH code described in I.3
subclause of the G.975.1 recommendation [2] uses the
BCH(2040,1930) and BCH(3860,3824) codes, which can
correct up to 10 and 3 bit errors per inner and outer
codeword, respectively. Furthermore, the conventional
concatenated BCH code uses three-iteration decoding which
provides 8.99dB NCG at 10^{-15} decoder output BER without
additive redundancy compared to the RS(255,239) code.
This technique can improve the error correction capability
without decreasing the code rate. The 10 Gb/s concatenated
BCH decoder using this code was proposed in [3]. Also, this
architecture can be used up to 40 Gb/s systems in its present form simply increasing the clock frequency using the pipelining technique. However, it is hard to achieve 100 Gb/s throughput in its present form because it will require a clock frequency of 800 MHz. To solve this problem, two-parallel architecture was proposed with a frame converter in [4]. But the two-parallel structure significantly increased hardware complexity.

2.2. Proposed two-iteration concatenated BCH code and its two-parallel processing scheme

We propose two-iteration concatenated BCH code with the aim of reducing the hardware complexity of the decoder for 100 Gb/s optical communications systems. Before discussing its detailed structure, one issue should be mentioned. At the standardization meeting held by ITU-T at March 2009, there was a proposal which adopts a fixed stuff byte in the OTU4 frame to compensate for a difference in nominal bit-rate with the optical channel data unit 4 (ODU4) frame [5]. Since the fixed stuff byte is not used in the ODU4 frame, it can be used as additional parity in the OTU4 frame. Using these additional parity bits we chose BCH(2040,1952) code and BCH(3904,3820) code as the inner code and outer code, respectively. Each inner and outer code can correct up to 8 and 7 bit errors per codeword respectively. From our C simulation result, the proposed code has provided better BER performance than the conventional concatenated BCH code when applied the same number of iterations. Figure 1 shows the performance simulation result. From our C simulation using binary phase shift keying (BPSK) transmission over the additive white Gaussian noise (AWGN) channel, the proposed two-iterative decoding code provides 8.91dB of NCG at the 10^{-15} decoder output BER, which is only 0.08dB lower than the conventional three-iteration concatenated BCH code. Also, with three-iterative decoding, the proposed code provides 9.01dB of NCG at the 10^{-15} decoder output BER, which is 0.02dB higher than the conventional three-iteration concatenated BCH code.

Figure 1 Bit error rate performance comparison.

100 Gb/s throughput with a practical clock frequency. The A->B and B->A blocks, shown in Figure 2 (a), at the input and output of the encoder and decoder represent frame converters which are required for two-parallel processing. The original OTU4 frame structure, namely the A-format, is not suitable for two-parallel processing because data alignment inside the OTU4 frame is serial. Thus we need to convert the serial OTU4 frame structure to a two-parallel frame structure, namely B-format. With the B-format OTU4 frame structure, two-parallel processing of encoding and decoding is possible.

After converted, each B-format frame is processed in the outer encoder and it aligned into an 8 BCH(3904,3820) codewords, as shown in Figure 2 (b). Also, the 8 B-format frames are collected in the interleaver and block-interleaved. Then the block-interleaved data in each frame aligned into 16 BCH(2040,1952) codewords, as shown in Figure 2 (c). The decoding process is executed in the reverse order. The interleaving/deinterleaving scheme of the proposed two-iteration concatenated BCH code is the same with that of the conventional three-iteration concatenated BCH code, which is well described in [2].

3. PROPOSED TWO-PARALLEL CONCATENATED BCH DECODER ARCHITECTURE

Figure 3 shows the block diagrams of the proposed two-iteration two-parallel concatenated BCH decoder, which has two-parallel BCH(3904,3820) and BCH(2040,1952) decoders. Figure 3 (a) shows the block diagram of the outer decoder, which processes 16 interleaved BCH(3904, 3820) codewords simultaneously. Since the single OTU4 frame consists of 8 BCH(3904,3820) codes, each upper and lower 128-bit section, which is same with a single symbol bit length of the B-format OTU4 frame, processes 8 BCH(3904,3820) codewords. The outer decoder has 16 SC blocks, 1 shared Dual-pSiBM KES block, and 16 Chien
Figure 2 Two-parallel concatenated BCH schemes: (a) two-parallel concatenated BCH scheme, (b) BCH(3904,3820) frame format for OTU4 and (c) BCH(2040,1952) frame format for OTU4.

search and error correction blocks. Each SC block processes 16 parallel bits and calculates on a Galois-field (GF) (2^11) symbolic base.

Figure 3 (b) shows the block diagram of the inner decoder, which processes 32 interleaved BCH(2040,1952) codewords simultaneously. Since the single OTU4 frame consist of 16 BCH(2040,1952) code, the inner decoder has 32 SC blocks, 2 shared Dual-pSiBM KES block, and 32 Chien search and error correction blocks. Each SC block processes 8 parallel bits and calculates on a GF(2^11)

codeword #1
codeword #2
codeword #7
codeword #8
1 Symbol
1 Frame = 3840 bits
8 bit payload data
2 bit parity bit
2 bit parity bit
128 bit payload data

Figure 3 Block diagrams of (a) BCH(3904, 3820) outer decoder, and (b) BCH(2040, 1952) inner decoder.

Symbolic base. Both the inner and outer decoder has a throughput of 256 bits per one clock cycle.

3.1. Syndrome computation block

The SC block calculates all the syndromes $S_i (1 \leq i \leq 2^{t-1})$ by putting the roots of generator polynomial $G(x)$ into the received codeword polynomial $R(x)$ as shown in the Equation (1) and (2).

$$R(x) = r_0 \cdot x^{2^t-1} + r_1 \cdot x^{2^t-2} + \cdots + r_{2^t-1} \cdot x + r_0$$ (1)

$$S_i = r_{2^t-i} \cdot \alpha^{i(2^t-1)} + r_{2^t-i} \cdot \alpha^{i(2^t-2)} + \cdots + r_{2^t-i} \cdot \alpha^i + r_0$$ (2)

Since Equation (3) always holds in BCH decoding [1], we can reduce the hardware complexity of the SC block significantly using Equation (3).

$$S_{2i} = (S_i)^2 \quad (i = 1,2,3,\cdots, t)$$ (3)

Figure 4 (a) shows the conventional SC block, in which each $S_i$ value is calculated in a SC cell in GF(2^11) symbol-based where $m$ is 11 and 12 bits for the inner and outer decoder, respectively. Since the SC block receives symbol-based codewords, which are 8 and 16 bits in the inner and outer decoder, respectively, parallel implementation of the SC cell is required as shown in Figure 4 (a). These parallel
Figure 4 Block diagrams of (a) conventional syndrome computation block for an inner and outer decoder, (b) proposed syndrome computation block for the outer decoder, and (c) proposed syndrome computation block for the inner decoder. SC cells require a lot of constant $GF$ multipliers, which result in huge hardware complexity. To reduce the hardware complexity, we applied Equation (3) to the SC block and replaced several SC cells with an equivalent number of $GF$ squaring units as shown in Figure 4 (b) and (c). The hardware complexity of the $GF$ squaring units is very low, specifically 6 XOR gates for the inner decoder and 23 XOR gates for the outer decoder. Thus, they have much less hardware complexity than the parallel SC cells. By implementing the SC block in this manner, the hardware complexity of the SC block is reduced by 8.8% and 22.9% for inner and outer decoders, respectively.

3.2. Dual-processing pipelined SiBM key equation solver block

BCH decoders can be implemented using the Berlekamp-Massey (BM) algorithm to solve the key equation $S(x)\Lambda(x)=\Omega(x) \mod x^t$ for an error locator polynomial $\Lambda(x)$ of BCH decoding procedure. Many conventional low-complexity BCH decoders have used the BM algorithm to solve the key equation. However, it is difficult to apply pipelining techniques because of the feedback loop. The conventional SiBM KES architecture in [6] is a simplified version of a well-known RiBM KES architecture. The SiBM KES architecture has an advantage in hardware complexity compared to the RiBM KES architecture. However, as its critical path delay is $T_{\text{multi}} + T_{\text{add}}$, it is difficult to get a high clock frequency required for a BCH decoder which targeted at 100 Gb/s optical communication systems. Therefore, a pipelined $GF$ multiplier is needed to obtain a higher clock frequency. Since the SiBM KES architecture has a feedback loop which computes a discrepancy value, it is difficult to apply pipelining techniques. The SiBM requires $t$ clock cycles to compute the error locator polynomial. Detailed SiBM KES algorithm and architecture were discussed in [6].

Figure 5 (a) shows the proposed Dual-pSiBM KES architecture, in which a total of $2t$ processing elements (PEs) are connected sequentially in two rows, and one main control unit block offers appropriate control signals to the PEs. Each PE processor has two pipelined $GF$ multipliers, which have a critical path delay of $3T_{\text{xor}} + T_{\text{add}}$. The polynomial update now requires 2 clock cycles instead of 1 clock cycle due to pipelining in the feedback loop. As a result, total computation cycles now become $2t$ clock cycles instead of $t$ clock cycles. That is, to compute an error locator polynomial, redundant $t$ clock cycles are required because of pipelining on the feedback loop. Moreover, meaningless data is stored in the registers inside $GF$ multipliers at every even-clock cycle. Thus, this property the dual syndrome polynomial processing architecture using redundant clock cycles occurred by pipelining in the feedback loop.

In the proposed Dual-pSiBM KES architecture, two syndrome polynomials of different codewords are inputted into the KES block at the same time and two error locator polynomials are calculated simultaneously. Figure 5 (c) shows a timing chart of the Dual-pSiBM architecture. At the first clock cycle, the first syndrome polynomial (A) is inputted to the KES block. And at the second clock cycle, the second syndrome polynomial (B) is inputted to the KES block consecutively. After initializing, the first calculation of the syndrome polynomial (A) is completed and stored in the polynomial register in the PE. At the same time, syndrome polynomial (B) is processed and stored in the pipelined registers of the multiplier. This processing concurrently operates during $2t$ clock cycles. In other words, a pipelined $GF$ multiplier calculates syndrome polynomial (A) on the odd clock cycles and syndrome polynomial (B) at the even clock cycles. After $2t$ clock cycles, each
syndrome polynomial is processed \(t\) times by the pipelined GF multiplier. That is, calculation of the first error locator polynomial is completed at the \(2t-1\) clock cycle and second one at the \(2t\) clock cycle. In this manner, two individual error locator polynomials are calculated during \(2t\) clock cycles separately. Therefore, the proposed architecture can share the same number of channels as the conventional SiBM architecture at the cost of additional registers to hold the second syndromes. However, maximum clock frequency is increased dramatically without penalty in latency in spite of the pipelining on the feedback loop.

3.3. Chien search and error correction block

The error locator polynomial \(\Lambda(x)\) is obtained by the KES block. The Chien search block searches the error locations by finding roots of \(\Lambda(x)\). The inverted power of roots indicates the error location in the codeword. The Chien search block was parallelized using a same equivalent circuit used in our previous work [4]. The error correction block corrects errors by XORing outputs of the FIFO and Chien search blocks.

3.4. Interleaver/Deinterleaver and frame converter block

The two-parallel interleaver/deinterleaver has same structure with the one used in our previous work [4]. And as mentioned earlier, the two-parallel architecture needs the frame converter in order to parallelize two serial frames at input and output port of the proposed two-iteration concatenated BCH decoder. The frame converter can be implemented using a SRAM memory. The required memory size for each interleaver/deinterleaver and frame converter is 32,640 bytes and 8,160 bytes, respectively.

4. IMPLEMENTATION RESULTS AND COMPARISON

The proposed two-parallel two-iteration concatenated BCH architecture was modeled in Verilog HDL and simulated to verify their functionality using a test pattern generated from a C simulator. After complete verification of the design functionality, it was synthesized and made layout using appropriate time and area constraints. Both simulation and synthesis steps were carried out using a SYNOPSYS design tool and 90-nm CMOS technology optimized for a 1.1V supply voltage.

4.1. Performance comparison of KES architectures

Table 1 summarizes implementation results of the KES architectures for both inner and outer decoders. The clock frequency and gate count were measured after layout for both inner and outer decoders. From our post-layout simulation results, the conventional SiBM architecture can operate at a clock frequency of approximately 330MHz. In contrast, the proposed Dual-pSiBM KES architecture can operate at approximately 430MHz. Thus the Dual-pSiBM KES architecture has a higher clock frequency compared to the conventional SiBM architectures. The hardware complexity of the proposed Dual-pSiBM architecture is higher than the conventional KES architectures due to pipeline registers required for holding second syndromes. In short, proposed Dual-pSiBM architecture provides excellent clock speed, but at the cost of a modest increase in the hardware complexity.

4.2. Performance comparison of concatenated BCH decoders

Table 2 shows the post-layout implementation results of the conventional three-iteration concatenated BCH architecture...
Table 1 Implementation result of the single KES blocks for BCH(3904,3820) and BCH(2040,1952) decoders using 90-nm CMOS technology.

<table>
<thead>
<tr>
<th>Design</th>
<th>Gate Count</th>
<th>Clock Rate (MHz)</th>
<th>Latency (clocks)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dual-pSiBM for BCH(3904,3820)</td>
<td>31,000</td>
<td>430</td>
<td>14</td>
</tr>
<tr>
<td>SiBM for BCH(3904,3820)</td>
<td>19,000</td>
<td>320</td>
<td>7</td>
</tr>
<tr>
<td>Dual-pSiBM for BCH(2040,1952)</td>
<td>21,000</td>
<td>440</td>
<td>16</td>
</tr>
<tr>
<td>SiBM for BCH(2040,1952)</td>
<td>18,000</td>
<td>330</td>
<td>8</td>
</tr>
</tbody>
</table>

Table 2 Implementation results of the concatenated BCH decoder architectures, using IBM 90-nm CMOS 1.1V library.

<table>
<thead>
<tr>
<th>Design</th>
<th>Redundancy ratio</th>
<th>Net coding gain at 10^-10 output BER</th>
<th>Total # of Gates</th>
<th>Area Usage (mm²)</th>
<th>Memory Size (bytes)</th>
<th>Clock rate (MHz)</th>
<th>Latency (clocks)</th>
<th>Throughput (Gb/s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>G975.1-I.3 three-iteration Concatenated BCH decoder</td>
<td>6.69%</td>
<td>8.99dB</td>
<td>2,781,000</td>
<td>9.5</td>
<td>252,576</td>
<td>320</td>
<td>8,148 (25.5μs)</td>
<td>81.9</td>
</tr>
<tr>
<td>Proposed two-iteration Concatenated BCH decoder</td>
<td>6.81%</td>
<td>8.91dB</td>
<td>1,928,000</td>
<td>6.3</td>
<td>154,464</td>
<td>430</td>
<td>5,082 (11.8μs)</td>
<td>110.1</td>
</tr>
</tbody>
</table>

and the proposed two-iteration concatenated BCH decoder architecture. The total number of gates and area usage for the proposed two-iteration concatenated BCH decoder are 1,928,000 and 6.3mm² respectively, excluding the RAM used in the interleavers/deinterleavers, frame converters and FIFOs. The required memory size for the two-iteration concatenated BCH decoder is approximately 155kbytes including all FIFOs, 2 frame converters, 1 interleaver and 2 deinterleavers. From post-layout simulation, the proposed two-iteration concatenated BCH decoder architecture can operate at a clock frequency of 430MHz and has a data processing rate of 110 Gb/s in 90-nm CMOS technology.

Compared to the conventional three-iteration concatenated BCH decoder, the proposed decoder architecture requires 853,000 lower gates, as well as 97 kbytes lower memory, which results in 34% lower actual area usage after placement and routing. The cost of the low hardware complexity is 0.08dB degradation in the NCG performance and 0.12% more parity ratio, but not losing the compatibility with the OTU4 frame. Therefore, the proposed high-speed, low-complexity decoder architecture features a very high data processing rate as well as high error correction capabilities, and makes it a suitable choice for next-generation 100 Gb/s optical communication systems.

5. CONCLUSION

This paper presents the design and implementation of the two-iteration concatenated BCH code and its high-speed low-complexity two-parallel decoder architecture for 100 Gb/s optical communications. A low-complexity syndrome computation architecture and a high-speed Dual-pSiBM KES architecture are applied to the proposed concatenated BCH decoder with the aim of implementing a high-speed low-complexity decoder architecture. Two-parallel processing by converting the frame format allows the decoder to achieve the high data processing rate required for 100 Gb/s optical communication systems. Also, the two-iteration concatenated BCH code with block interleaving methods allows the decoder to achieve high correction ability to compensate for serious transmission quality degradation. As a result, the proposed decoder architecture features a very high data processing rate as well as excellent error correction capability. Thus, it has potential applications in the next-generation FEC schemes for 100 Gb/s optical communications.

6. ACKNOWLEDGEMENT

This work was supported by the IT R&D program of MKE/KEIT. [KI002145, High Speed Digital Signal Processing based CMOS Circuit Design for Next-generation Optical Communication]

7. REFERENCES


