

行政院國家科學委員會補助專題研究計畫成果報告

※※※

※ 應用於無線通訊之低功耗基頻處理器 ※

※※※

計畫類別：個別型計畫      整合型計畫

計畫編號：NSC97-2220-E-009-166

執行期間： 98 年 8 月 1 日 至 99 年 7 月 31 日

計畫主持人：李 鎮 宜 教授

計畫參與人員：陳志龍、林義閔、賴義澤、李欣儒、林佳龍、歐陽亦桓、  
許智翔、郭明諭、朱娟瑤

執行單位：國立交通大學 電子工程學系

中 華 民 國      99 年 5 月 31 日

## 應用於無線通訊之低功耗基頻處理器(2/3)

A Low-Power Baseband Processor for Wireless Communication System (2/3)

計畫編號: NSC97-2220-E-009-166

執行期間: 98 年 8 月 1 日 至 99 年 7 月 31 日

主持人: 李鎮宜 交通大學電子工程系教授

參與學生: 陳志龍、林義閔、賴義澤、李欣儒、林佳龍、歐陽亦桓、許智翔、  
郭明諭、朱娟瑤

### 一、 中文摘要

基頻訊號處理在無線通訊系統上扮演關鍵性的角色，不僅可有效提升傳輸的效能，更能提供多模式和多標準的系統實現方案。然而要達成低成本和低功率設計方法，不僅對於個別模組的演算法需深入瞭解外，也必須融入系統層級的行為，方能提供一個具有技術競爭力的解決方案。因此在這一年的研究計畫，我們針對 OFDM 主流無線通訊系統所需求的關鍵模組 LDPC decoder 跟 BCH decoder 進行相關議題的研究，探討高速、低功率 LDPC decoder 以及低複雜度的軟性 BCH decoder 的設計方式，並研究在不同設計規範下達成多模和多標準的作業模式，之後將會把此關鍵模組設計整合，並配合基頻訊號的同步模組電路，完成一個符合多模、多標準的低功率基頻處理器。

### 關鍵字

基頻處理器；多模式；多標準；低成本；低功率；LDPC Decoder；BCH Decoder

## 英文摘要

Signal processing in baseband processor designs plays a key role in wireless communication system designs—in not only improving overall system transmission performance, but also providing the capability of multi-mode and multi-standard for cost-effective system realization. To reach better performance indices in terms of low-cost and low-power, it is necessary to investigate system design methodologies, covering in-depth exploration of algorithms of key modules and exploitation of unique features/behaviors of a complete system. As a result, a more competitive solution can be delivered. In this year (2009/8~2010/7), we have concentrated on the key modules (LDPC decoder and BCH decoder) of the main-stream OFDM wireless communication systems. The first issue is the high-speed solution for LDPC decoder. The second issue is the low-cost solution for soft BCH decoder. In the end, these design techniques and key modules will be integrated on a design platform, together with synchronization modules, to come up with a multi-mode, multi-standard, and low-power baseband processor.

## Keywords

Baseband Processor, Multi-mode, Multi-Standard, Low-Cost, Low-Power, LDPC Decoder, BCH Decoder

## 二、計畫的緣由與目的

### A. *A 11.5-Gbps LDPC Decoder Based on CP-PEG Code Construction*

Low-density parity-check (LDPC) code is a famous error control code with near Shannon limit performance [A-1] and can be described by its parity-check matrix  $H$ . The message exchanging order between nodes is called scheduling, which will influence the convergence speed of the decoding algorithm. In standard BP algorithm, simultaneous update of all check node messages or variable node messages is named as flooding scheduling. Alternatively, the layered BP algorithm [A-3] performing message update along different check node groups is a method of check-node-centric sequential scheduling (CSS). Researchers have revealed that CSS could reduce maximum iteration to approximate half of the standard BP with similar performance.

Recently, LDPC codes adopted in high-throughput systems have high code-rate property to increase channel efficiency. However, the introduced large check node degree  $d_c$  will cause implementation full of difficulties. Even though the CSS could reduce the iteration number, the throughput is still degraded due to long critical path of check node unit (CNU).

In this project, the proposed decoder aims at providing a high-throughput and hardware-efficient solution to the high code-rate LDPC with large check node degrees. In order to reduce the iteration number, the decoding scheduling is based on the variable-node-centric sequential scheduling (VSS; also known as shuffled decoding [A-6]), where the messages are updated along different variable node groups. Since the inputs of CNU operation are also divided into several subgroups, the complexity and critical path delay of CNU are reduced. Furthermore, single pipelined approach and modified CNU are proposed to minimize the message storage memory. Using a(2048,1920) LDPC code constructed by circulant permutation progressive edge-growth (CP-PEG) algorithm [A-7] as a design example, the overall decoder chip implemented in 90nm technology will show its advantages in terms of throughput, energy efficiency, and hardware efficiency.

## B. An Improved Soft BCH Decoder with One Extra Error Compensation

The Bose-Chaudhuri-Hocquenghen (BCH) [B-1] codes are popular in storage and communication systems, such as flash device, DMB-T [B-2] and DVB-S2 [B-3] broadcasting systems. Recently, soft decoding of BCH codes has aroused many research interests. Forney developed the generalized-minimum-distance (GMD) [B-4] to generate a list of candidate codewords and choose a most likely codeword from the list. Other algorithms with similar concept, such as Chase [B-5] and SEW [B-6], are also widely used in many applications. Moreover, Therattil and Thangaraj provided a sub-optimum MAP BCH decoding method with Hamming SISO decoder in 2005 [B-7].

In general, the complexity of a soft BCH decoder is much higher than a hard BCH decoder for decoding an entire codeword. Nevertheless, soft BCH decoders with lower complexity can be revealed by focusing on the least reliable bits instead of the whole codeword. A soft BCH decoding algorithm using error magnitudes to deal with the least reliable bits was developed in 1997 [B-8]. However, Fig. 1 shows that there is about 0.25 db performance loss at  $BER = 10^{-5}$  in AWGN channel as compared to hard decision BCH decoder for BCH (255,239) code. For the existing soft decision algorithms, the soft BCH decoder provides either better error correcting performance or lower hardware complexity than a traditional hard BCH decoder. In this project, a soft BCH decoder which has similar concept as [B-8] and enhances the correcting performance by compensating one extra error while maintaining the low hardware complexity is presented.

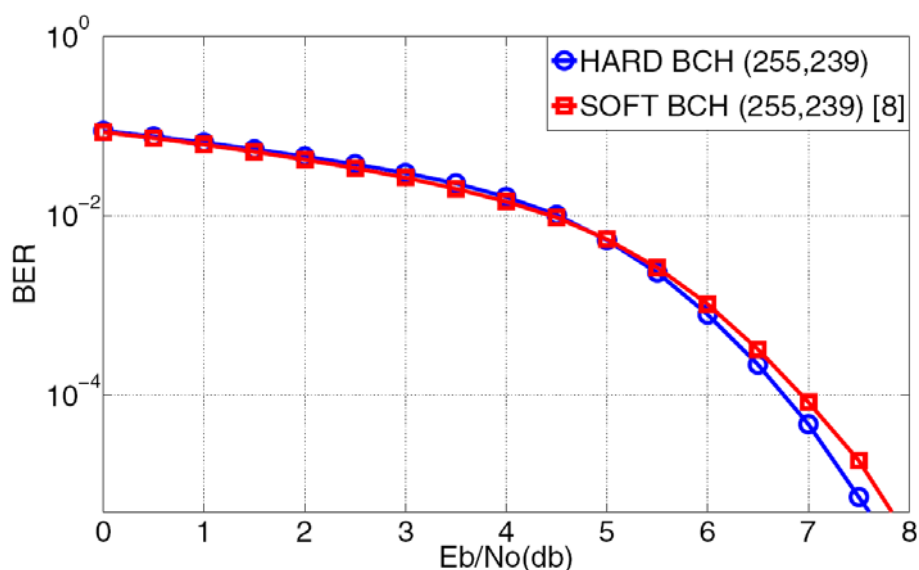


Fig1. Simulation Result for BCH (255,239)

### 三、 研究方法及成果

#### A. *A 11.5-Gbps LDPC Decoder Based on CP-PEG Code Construction*

##### 1. **CODE STRUCTURE AND DECODING ALGORITHM**

###### I. **CP-PEG LDPC Code Construction**

The (2048, 1920) irregular LDPC code, rate-15/6, used in this project was constructed by CP-PEG algorithm and shown in Fig. 2(a). The constructed parity-check matrix  $H$  consists of  $p \times p$  circulant permutation (CP) and all-zero matrices. A CP matrix is a cyclic square matrix with constant row and column weight of one. The number of each CP matrix indicates the cyclic shift amount and -1 means all zero matrixes. By setting  $p=32$ , there are  $4 \times p$  check nodes and  $64 \times p$  variable nodes in bipartite graph, where each check node has uniform degree 46, and  $16 \times p$ ,  $24 \times p$ ,  $24 \times p$  variable nodes have degrees of 4, 3, 2 respectively. The performance of this code was proven to have better performance than other PEG-based LDPC codes [A-7]; nevertheless, the high check node degree required suitable decoder architecture to overcome implementation difficulties.

###### II. **Variable-node centric Sequential Scheduling**

In VSS approach, the initialization, stopping criterion test, and output steps remain the same as the standard BP algorithm. The only difference between two algorithms lies in the updating procedure. The normalized min-sum (NMS) algorithm which compensates the approximation error in check node In VSS approach, the initialization, stopping criterion test, and output steps remain the same as the standard BP algorithm. The only difference between two algorithms lies in the updating procedure. The normalized min-sum (NMS) algorithm which compensates the approximation error in check node is shown and described in the next page.

In this work, the codeword is divided into  $G=4$  groups, therefore the parity-check matrix  $H$  is divided into 4 sub-matrices ( $H_1$  to  $H_4$ ). As shown in Fig. 2(b), each sub-matrix consists of equal number of variable nodes with the same degree to reduce the hardware cost of variable node unit (VNU). Moreover, the sub-matrices with the same shift amounts (shaded blue CP matrices) are arranged in the same position thus the routing and control could be further simplified.

- 1) **Initialization:**  $z_{mn}^{(0)} = P_n$ ,
- 2) **Iterative Decoding:** For  $0 \leq g \leq G - 1$ , perform the following two steps.
  - a) Check node to variable node update step, for  $g \cdot N_G \leq n \leq (g + 1) \cdot N_G - 1$  and each  $m \in M(n)$ , process
$$\varepsilon_{mn}^{(i)} \approx \prod_{\substack{n' \in N(m) \setminus n \\ n' \leq g \cdot N_G - 1}} \text{sign}(z_{mn'}^{(i)}) \times \prod_{\substack{n' \in N(m) \setminus n \\ n' \geq g \cdot N_G}} \text{sign}(z_{mn'}^{(i-1)})$$

$$\times \min \left\{ \min_{\substack{n' \in N(m) \setminus n \\ n' \leq g \cdot N_G - 1}} \left\{ |z_{mn'}^{(i)}| \right\}, \min_{\substack{n' \in N(m) \setminus n \\ n' \geq g \cdot N_G}} \left\{ |z_{mn'}^{(i-1)}| \right\} \right\} \times \beta \quad (1)$$
  - b) variable node to check node update step, for  $g \cdot N_G \leq n \leq (g + 1) \cdot N_G - 1$ , process
$$z_{mn}^{(i)} = P_n + \sum_{m' \in M(n) \setminus m} \varepsilon_{m'n}^{(i-1)} \quad (2)$$

$$z_n^{(i)} = P_n + \sum_{m \in M(n)} \varepsilon_{mn}^{(i-1)} \quad (3)$$
- 3) **Hard Decision:** Let  $X_n$  be the  $n$ -th bit of decoded codeword. If  $z_n^{(i)} \geq 0$ ,  $X_n = 0$ , else if  $z_n^{(i)} < 0$ ,  $X_n = 1$ .

$H =$

H1				H2				H3				H4				(p)			
31	31	...	31	0	31	31	...	31	0	31	31	...	31	-1	31	31	...	31	-1
16	29	...	-1	-1	6	27	...	22	31	29	21	...	-1	0	16	3	...	14	-1
28	19	...	-1	-1	16	7	...	-1	-1	11	9	...	-1	31	30	4	...	-1	0
1	1	...	16	0	23	1	...	-1	-1	26	22	...	27	-1	3	12	...	-1	31
64p																			

(a) Original matrix

	Deg=4				Deg=3				Deg=2							
H1	31	31	31	31	-1	-1	31	17	31	31	31	-1	31	-1	31	0
	16	29	3	25	15	31	-1	31	24	18	-1	-1	11	31	-1	-1
	28	19	29	14	31	4	1	-1	-1	26	12	31	-1	16	-1	-1
	1	1	7	29	11	10	30	14	15	-1	-1	16	-1	-1	16	0
H2	31	31	31	31	-1	-1	12	31	31	26	31	-1	31	-1	31	0
	6	27	29	15	25	30	-1	23	26	31	-1	-1	28	-1	22	31
	16	7	10	15	31	1	31	-1	-1	8	22	31	-1	31	-1	-1
	23	1	2	29	17	28	1	9	19	-1	-1	15	-1	21	-1	-1
H3	31	31	31	31	-1	17	6	31	31	7	-1	-1	31	-1	31	-1
	29	21	7	20	31	-1	-1	5	9	16	31	-1	12	31	-1	0
	11	9	0	31	6	3	31	-1	5	31	-1	31	-1	28	-1	31
	26	22	11	24	26	31	0	4	-1	-1	11	20	-1	-1	27	-1
H4	31	31	31	31	-1	31	28	20	31	31	-1	-1	31	-1	31	-1
	16	3	29	1	29	-1	-1	31	2	31	31	-1	0	-1	14	-1
	30	4	21	27	31	8	31	-1	20	3	-1	31	-1	31	-1	0
	3	12	0	14	29	17	2	7	-1	-1	9	31	-1	22	-1	31
	16p															

(b) Permuted and divided into four groups

Fig2. Parity-Check Matrix of(2048,1920) LDPC Code

## 2. PROPOSED DECODER ARCHITECTURE

In this section, the complete decoder architecture will be presented, including data path, scheduling, and VLSI structure of CNU and modified CNU.

### I. Single Pipelined Architecture

The entire decoder depicted in Fig. 3(a) is composed of fully-parallel CNUs and partial-parallel VNUs, where the VNU2, VNU3, and VNU4 will handle variable node operations with degree 2, 3, and 4 respectively. Let  $\alpha_g^{(i)}$  denote the sorted messages sent from variable nodes in the  $g$ -th group to one specific check node at  $i$ -th iteration, which is:

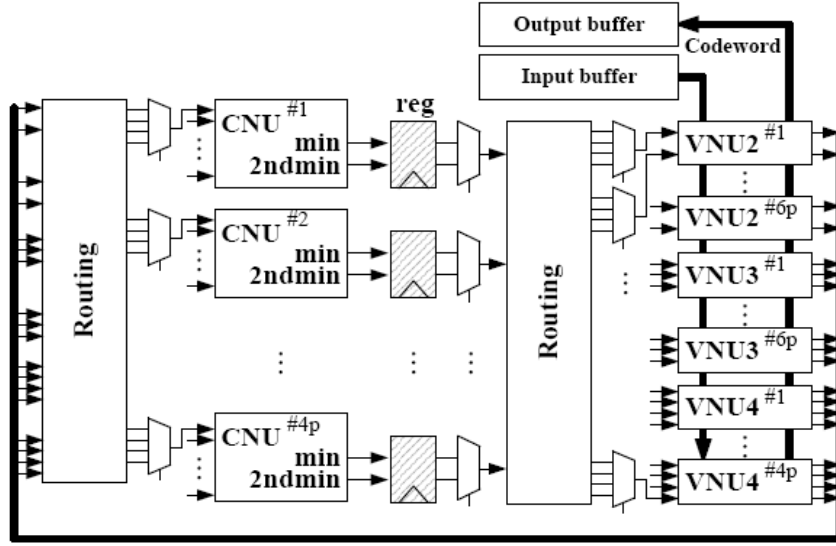
$$\alpha_g^{(i)} = \min_{\substack{n' \in N(m) \setminus n \\ g \cdot N_G \leq n' \leq (g+1) \cdot N_G - 1}} \left\{ \left| z_{mn'}^{(i)} \right| \right\} \quad (4)$$

Then the magnitude part of check node to variable node message in (1) could be computed by the following equation:

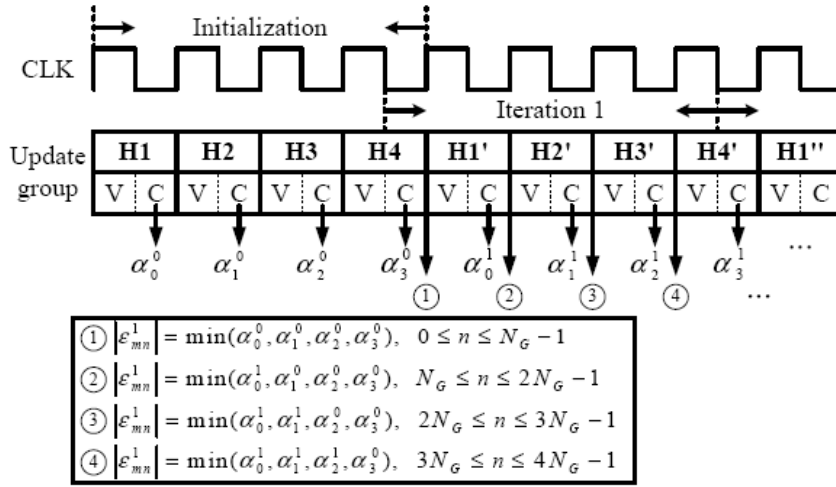
$$\left| \varepsilon_{mn}^{(i)} \right| = \min \left\{ \left\{ \alpha_j^{(i)} \right\}_{j < g}, \alpha_g^{(i)}, \left\{ \alpha_k^{(i-1)} \right\}_{k > g} \right\} \quad (5)$$

Fig. 3(b) demonstrates the timing diagram of proposed decoder. There are  $G$  initialization cycles required to calculate  $\alpha_g^{(i)}$  for  $0 \leq g \leq G - 1$ . Since only one subgroup of the message  $z_{mn}^{(i)}$  is updated in  $g$ -th cycle of one iteration, the main operation of CNU could be simplified to calculate  $\alpha_g^{(i)}$  (local sorting) in each cycle and then perform global sorting like equation (5). From the propose single pipelined architecture, only messages  $\alpha_g^{(i)}$  and  $\varepsilon_{mn}^{(i)}$  are stored. The sorted results could be represented by min value, second min value, and the index of min value in NMS algorithm. Therefore, the proposed decoder only latches two values, one index, and sign part of messages in each subgroup, while the variable node to check node message  $z_{mn}^{(i)}$  is on-the-fly calculated. The single pipelined architecture is feasible because the CNU could be updated immediately after VNU's operations in VSS approach.





(a) Single pipelined LDPC decoder architecture



(b) variable-node-centric sequential scheduling

Fig3. Proposed Architecture and Scheduling

## II. Modified CUN

The operation of check node to variable node update could be divided into magnitude part and sign part. Fig. 4(a) illustrates the magnitude part of CNU, which is an accumulative sorter composed of a local sorter and a global sorter. The local sorter is used to find the local min and second min values in each subgroups, and global min and second min values of a check node will be found by a global sorter. Similarly, the sign operation can be computed in an accumulative way like the accumulative sorter.

For our proposed CP-PEG LDPC codes with  $dc = 46$ , The VSS approach with  $G = 4$  could divide 46 inputs of the sorter into only 12 inputs. More subgroup number  $G$  will result in fewer inputs of sorter, but increase the storage for min, second min, and index values of each subgroup. In order to further reduce the storage overhead of each subgroup, we propose a reduced storage

accumulative sorter as shown in Fig. 4(b). The basic idea is to simplify the local min and local second min from  $G - 1$  subgroups into one group. Some extra control circuits are needed to open or close the feedback loop in Fig. 4(b). This sorter architecture is beneficial since the complexity reduction of storage registers and global sorters is higher than the overhead of control circuits. Section IV will show the performance of this modified CNU is similar to original CNU.

### III. Summary

In traditional two-stage pipelined architecture, both  $z_{mn}^{(i)}$  and  $\varepsilon_{mn}^{(i)}$  messages are kept in registers or memory. Assume the bit-width of messages is  $w$  ( $= 6$ ) and variable node degree is  $dv$ , then the required memory size (or registers) is as follows:

$$\begin{aligned}
& MEM_{VNU} + MEM_{CNU} \\
&= (\sum dv) \cdot w + \\
&\quad (N - K) \cdot (Min + 2ndMin + Index + Sign) \\
&= (\sum dv) \cdot w + (N - K) \cdot (2 \cdot (w - 1) + \log_2[dc] + dc) \\
&= 5888 \cdot 6 + 128 \cdot (2 \cdot 5 + \log_2[46] + 46) = 43264 \quad (6)
\end{aligned}$$

For the proposed single pipelined decoder and modified CNU in Fig.4 (b), the memory size is reduced to

$$\begin{aligned}
& MEM_{CNU} \\
&= (N - K) \cdot \\
&\quad (2 \cdot (Min + 2ndMin + Index + 2ndIndex) + Sign) \\
&= (N - K) \cdot (4 \cdot (w - 1) + 4 \cdot \log_2[dc] + dc) \\
&= 128 \cdot (4 \cdot 5 + 4 \cdot \log_2[46] + 46) = 11520 \quad (7)
\end{aligned}$$

Therefore the overall register reduction of proposed architecture is 73%, leading to the following advantages: fewer registers, higher utilization of functional units, and reduced complexity. Since high-rate LDPC codes usually have more VNUs than CNUs (in our case: 512 VNUs and 128 CNUs), the elimination of registers from VNU to CNU not only reduces hardware cost but also lowers power consumption of clock tree.

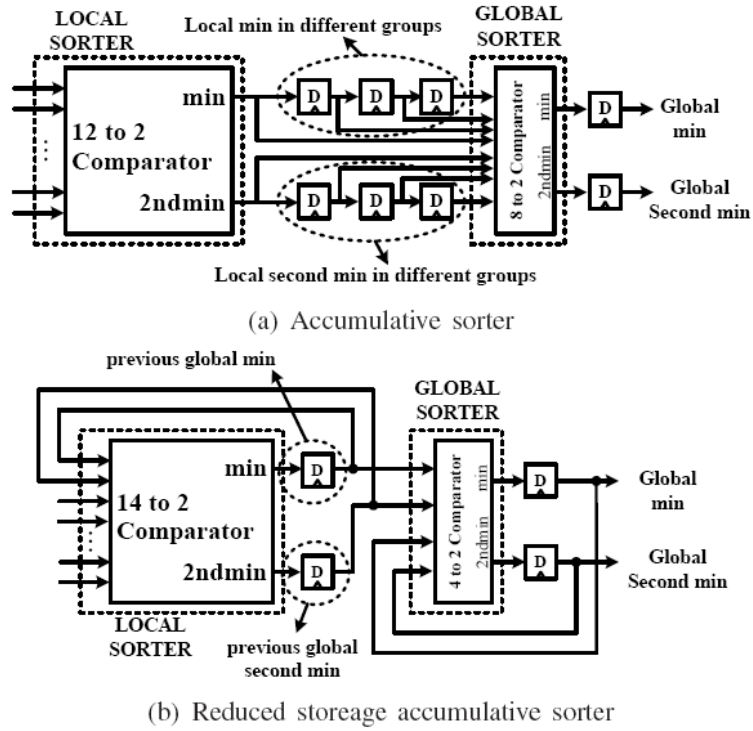


Fig4. CNU Architecture

### 3. PERFORMANCE AND IMPLEMENTATION RESULTS

Under AWGN channel with BPSK modulation, the performance curves are simulated to determine the required bit-width and maximum iteration number. The simulation parameters of proposed algorithm are 6-bit input quantization (5-bit integer and 1-bit decimal fraction), scaling factor 0.75 for NMS algorithm, and 4 or 5 iterations. In Fig. 5, the bit-error rate (BER) curves indicate that 4 iterations for the proposed algorithm are sufficient to achieve similar performance of standard BP algorithm with 7 iterations. Furthermore, in the aspect of almost the same code-rate and better error-correcting capability, our CP-PEG LDPC codes outperforms 1.2 dB better than the (255, 239) RS code at  $BER=10^{-5}$ , which reveals the potential of CP-PEG LDPC codes for storage applications and fiber optical communication systems. The overall SNR loss between this work and Shannon limit is only 1.6dB. The proposed LDPC decoder is implemented by standard-cell design flow and fabricated in 90-nm 1P9M CMOS technology. The core occupied 3.84 mm<sup>2</sup> of area with 68% utilization. The die photo is shown in Fig. 6, where the distribution of CNUs and VNUs is auto-determined by APR tool. Since required decoding cycles of one LDPC codeword are 4 initialization cycles plus 4 iterations, the throughput is  $(1920\text{bit}/20\text{cycles}) \times \text{frequency}$ . Fig. 7 shows the measured maximum throughput and power

dissipation under different SNR conditions and supply voltages. The measurement result indicates that the test chip with FF corner can achieve 11.5 Gbps throughput under 1.4V supply voltage. The throughput could be scaled down to 5.77Gbps with 0.8V supply voltage to meet the throughput requirement of IEEE 802.15.3c standard and the energy efficiency will be 0.033 nJ/bit. Compared with the state-of-the-art in Table I, the proposed LDPC decoder outperforms others in the aspects of throughput, hardware efficiency, and power efficiency. Since the LDPC code specifications of these designs are different, the SNR loss between each work to their Shannon limit is also listed for reference.

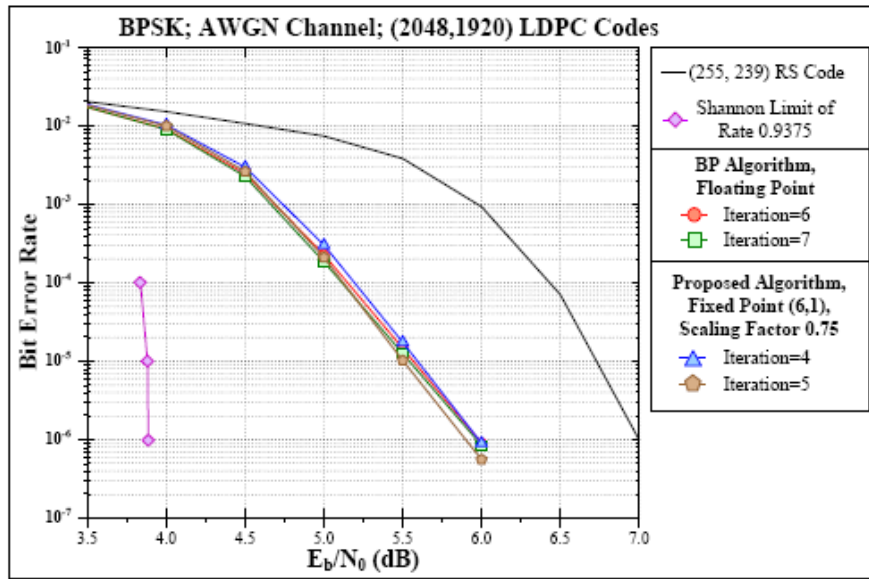


Fig5. Performance

#### 4. CONCLUSION

A high-throughput and power-efficient LDPC decoder is presented. Utilizing the characteristic of variable-node-centric sequential scheduling, the proposed decoding algorithm could reduce the maximum iteration number without performance loss. In addition, the single pipelined architecture and modified CNU can save 73% message storage memory and decrease the sorter size, resulting in a low-complexity design. After implementation in 90nm technology, the test chip occupies 3.84 mm<sup>2</sup> of area and supports maximum 11.5 Gbps data rate under 1.4V supply voltage.

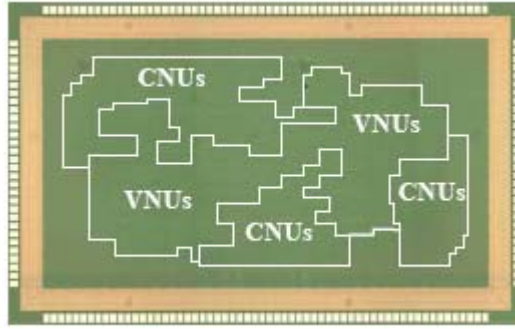


Fig6. Chip Micrograph

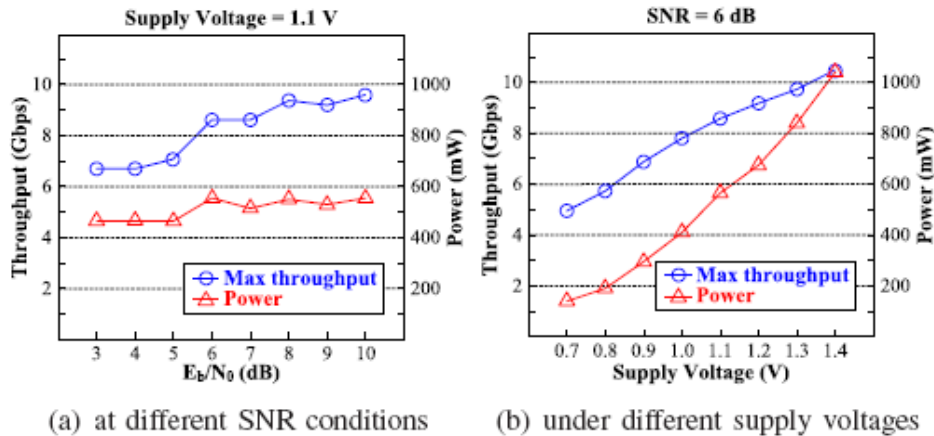


Fig7. Measured Maximum Throughput and Power Consumption

Table1. Comparison with the State-of-The-Art

	This work	CICC'07 [8]	SOVC'07 [9]
Process	90-nm	0.13- $\mu$ m	0.13- $\mu$ m
Code Spec	(2048, 1920)	(660,480)	Wimax
Code Rate	0.9375	0.73	0.5
Core Area	3.84 $mm^2$	7.3 $mm^2$	4.45 $mm^2$
Gate Count	708k	690k	420k
Iteration	4	15	2 - 8
Input Quantization	6 bits	4 bits	8 bits
Clock Frequency	120 MHz	300 MHz	83.3 MHz
Max. Throughput	11.5 Gbps	2.44 Gbps	222 Mbps
Power	191.2 mW <sup>1</sup>	1383 mW <sup>2</sup>	52 mW
Energy Efficiency	0.033 nJ/bit <sup>1</sup>	0.566 nJ/bit <sup>2</sup>	0.23 nJ/bit
Hardware Efficiency <sup>3</sup>	12.1	3.5	0.53
SNR loss to Shannon limit <sup>4</sup>	1.6 dB	2.8 dB	2.9 dB

<sup>1</sup> when frequency is scaled to 5.77 Gbps throughput with 0.8V supply voltage where BER= $10^{-6}$

<sup>2</sup> when  $E_b/N_0=5.5$ dB indicating BER <  $10^{-8}$

<sup>3</sup> Throughput/Gate count (Mbps/K-gate)

<sup>4</sup> when BER= $10^{-5}$

B. An Improved Soft BCH Decoder with One Extra Error Compensation

1. PROPOSED COMPENSATION SOFT BCH DECODING

The proposed soft BCH decoder shown in Fig. 8 includes three major steps: syndrome calculator, error locator evaluator, and compensation error magnitude solver. From the received polynomial  $R(x)$ , the syndrome polynomial  $S(x) = S_1 + S_2x^1 + \dots + S_{2t}x^{2t-1}$  is expressed as

$$S_j = R(\alpha^j) = \sum_{i=1}^v (\alpha^j)^{e_i} = \sum_{i=1}^v (\beta_{e_i})^j \quad (8)$$

for  $j = 1, 2, \dots, 2t$ , where  $\alpha$  is the primitive element over  $GF(2^m)$ . Notice that  $e_i$  is the  $i$ -th actual error location and  $\beta_{e_i} = \alpha^{e_i}$  indicates the corresponding error locator.

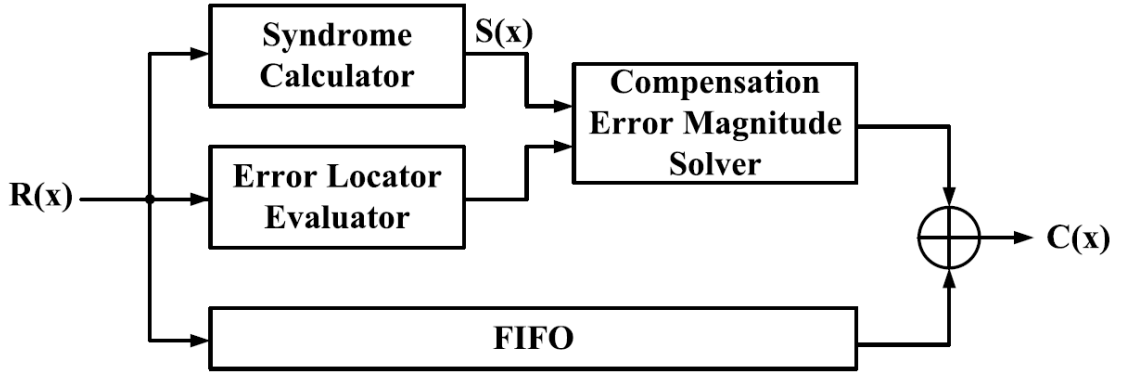


Fig8. Soft Decision BCH Decoding Block Diagram

With soft inputs, error locator evaluator can choose  $2t$  least reliable inputs and evaluate their corresponding error locators to form the error locator set  $B = [\beta_{l_1}, \beta_{l_2}, \dots, \beta_{l_{2t}}]^T$ . Also, the error location set,  $L = [l_1, l_2, \dots, l_{2t}]^T$ , can be calculated with  $B$  because the error locator of the  $l_i$ -th location is  $\beta_{l_i} = \alpha^{l_i}$ . The relation between  $B$  and the syndrome vector,  $S = [S_1, S_2, \dots, S_{2t}]^T$ , can be formulated as

$$\begin{bmatrix} \beta_{l_1} & \beta_{l_2} & \dots & \beta_{l_{2t}} \\ \beta_{l_1}^2 & \beta_{l_2}^2 & \dots & \beta_{l_{2t}}^2 \\ \vdots & \vdots & \dots & \vdots \\ \beta_{l_1}^{2t} & \beta_{l_2}^{2t} & \dots & \beta_{l_{2t}}^{2t} \end{bmatrix} \begin{bmatrix} \gamma_1 \\ \gamma_2 \\ \vdots \\ \gamma_{2t} \end{bmatrix} = \begin{bmatrix} S_1 \\ S_2 \\ \vdots \\ S_{2t} \end{bmatrix} \quad (9)$$

where  $\Gamma = [\gamma_1, \gamma_2, \dots, \gamma_{2t}]^T$  is the error magnitude set corresponding to  $B$ , and the  $2t \times 2t$  matrix in (9) is defined as  $\beta$ -matrix  $B$ . Let  $\Delta = [\delta_1, \delta_2, \dots, \delta_{2t}]^T$  be defined as

$$\underline{\Delta} = \mathbf{B} \times \underline{\Gamma} + \underline{S} \quad (10)$$

From (8) and (9), it is evident that if all the errors are in the error location set, the exact  $\gamma_i$  value can be determined and  $\Delta$  will be all zero; otherwise, this decoding approach fails to correct errors. There are at most  $2t$  error locations can be determined. However, it is very likely that only one error outside  $L$  but the decoder can't solve any error. To improve the error correcting ability, we additionally check whether  $\Delta$  is a geometrical sequence or not to make a compensation for an error location outside  $L$ . A geometrical sequence  $\Delta = [\beta_{lloss}, \beta_{lloss2}, \dots, \beta_{lloss2t}]$  means an error location  $lloss$  can be found, where  $\beta_{lloss} = \alpha_{lloss}$ . For example, if there are four errors in 1st, 3rd, 5th and 9th locations for a BCH (255,239) decoder which can correct 2 errors,  $S$  is expressed as

$$\underline{S} = \begin{bmatrix} \beta_1 + \beta_3 + \beta_5 + \beta_9 \\ \beta_1^2 + \beta_3^2 + \beta_5^2 + \beta_9^2 \\ \beta_1^3 + \beta_3^3 + \beta_5^3 + \beta_9^3 \\ \beta_1^4 + \beta_3^4 + \beta_5^4 + \beta_9^4 \end{bmatrix}^T \quad (11)$$

In the case that the decoder collects  $B = [\beta_1, \beta_3, \beta_6, \beta_9]$ , and  $\Gamma = [1, 1, 0, 1]$ ,  $\Delta$  becomes

$$\begin{aligned} \underline{\Delta} &= \begin{bmatrix} \beta_1 & \beta_3 & \beta_6 & \beta_9 \\ \beta_1^2 & \beta_3^2 & \beta_6^2 & \beta_9^2 \\ \beta_1^3 & \beta_3^3 & \beta_6^3 & \beta_9^3 \\ \beta_1^4 & \beta_3^4 & \beta_6^4 & \beta_9^4 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 0 \\ 1 \end{bmatrix} + \begin{bmatrix} \beta_1 + \beta_3 + \beta_5 + \beta_9 \\ \beta_1^2 + \beta_3^2 + \beta_5^2 + \beta_9^2 \\ \beta_1^3 + \beta_3^3 + \beta_5^3 + \beta_9^3 \\ \beta_1^4 + \beta_3^4 + \beta_5^4 + \beta_9^4 \end{bmatrix} \\ &= [\beta_5, \beta_5^2, \beta_5^3, \beta_5^4] \end{aligned} \quad (12)$$

Then not only errors at 1-st, 3-rd and 9-th locations but also an error at 5-th location can be corrected. Therefore, the proposed compensation soft BCH decoder can correct at most  $2t+1$  error. The compensation error magnitude solver (CEMS) shown in Fig. 8 is used to solve (9) and (10) to get  $\Gamma$  and  $\Delta$ . For those  $\gamma_i$  equal to 1, the corresponding  $l_i$  and  $l_{loss}$  are the exact error locations. The codeword polynomial  $C(x)$  can be obtained by inverting values at error locations in the received polynomial  $R(x)$ .

To obtain the  $\gamma_i$  value in (9), the Gauss Elimination method is the most intuitive way but the complexity is  $O(n^3)$ . In BCH codes, the valid error magnitude in (9) is either 0 or 1, so the problem can be formulated into checking all combinations of  $\gamma_i$  over  $GF(2)$  instead of calculating real error magnitudes. A  $2t$ -bit counter is used to do a heuristic search for all binary combinations. Since  $S_{12} = S_2, S_{22} = S_4, \dots, S_{2t} = S_{2t}$  in BCH codes, the even part of syndromes check can be

eliminated to simplify (9) as :

$$\begin{bmatrix} \beta_{l_1} & \beta_{l_2} & \cdots & \beta_{l_{2t}} \\ \beta_{l_1}^3 & \beta_{l_2}^3 & \cdots & \beta_{l_{2t}}^3 \\ \vdots & \vdots & \cdots & \vdots \\ \beta_{l_1}^{2t-1} & \beta_{l_2}^{2t-1} & \cdots & \beta_{l_{2t}}^{2t-1} \end{bmatrix} \begin{bmatrix} \gamma_1 \\ \gamma_2 \\ \vdots \\ \gamma_{2t-1} \\ \gamma_{2t} \end{bmatrix} = \begin{bmatrix} S_1 \\ S_3 \\ \vdots \\ S_{2t-1} \end{bmatrix} \quad (13)$$

The complexity can be significantly reduced for only half size matrix, Bodd and Sodd, used in (13).

Following steps illustrate the details of the efficient Implementation of CEMS.

Input :  $\underline{\mathcal{B}}$  ,  $\underline{S}_{odd}$  and  $\underline{\Gamma} = 0$

1) Construct the  $\beta$ -matrix  $\mathbf{B}_{odd}$  with  $\underline{\mathcal{B}}$

2)  $\underline{\Delta}_{odd} = \mathbf{B}_{odd} \times \underline{\Gamma} + \underline{S}_{odd}$

3) if  $\underline{\Delta}_{odd}$  is a geometrical sequence

Go to 4)

else

if  $\underline{\Gamma} == 2^{2t} - 1$

Failed Decoding

else

$\underline{\Gamma} = \underline{\Gamma} + 1$

Go to 2)

4) Find  $l_{loss}$  with  $\underline{\Delta}_{odd}$

Output :  $\underline{\Gamma}$  and  $L_{loss}$

By iteratively counting  $\Gamma$  value, a heuristic search for all binary combinations can be completed. At each iteration, the solver can verify whether the geometrical sequence check stands or not.

## 2. VLSI ARCHITECTURE FOR THE COMPENSATION SOFT BCH DECODER

### I. Error Locator Evaluator

As shown in Fig. 9, error locator evaluator architecture includes the reliability part, the error locator part and the error location part. The upper part is the reliability part which stores the reliabilities of  $2t$  least reliable candidates  $R_{l1}, R_{l2}, \dots, R_{l2t}$ . The medium part is the error locator part to construct the error locator set  $B$ . Because the error locator of the  $i$ -th location is  $\alpha_i$ , the error locator of  $(i+1)$ -th locations is  $\alpha$  times the error locator of  $i$ -th location. The error locator can



be computed by multiplying  $\alpha^{-1}$  with register REG if the input is serial in from the highest degree coefficient of  $R(x)$ .

Thus, the error locator part can use a constant multiplier to calculate the error locator of each input. Notice that register REG initially contains the error locator of the first input. The bottom part is the error location part. The decoding method focuses on the least reliable bits instead of the whole codeword, so the error location part uses a counter to compute the error location  $l_i$  corresponding to each  $R_{li}$  for serial input. Hence, the Chien search procedure is no longer required and a lot of redundant decoding latencies can be eliminated.

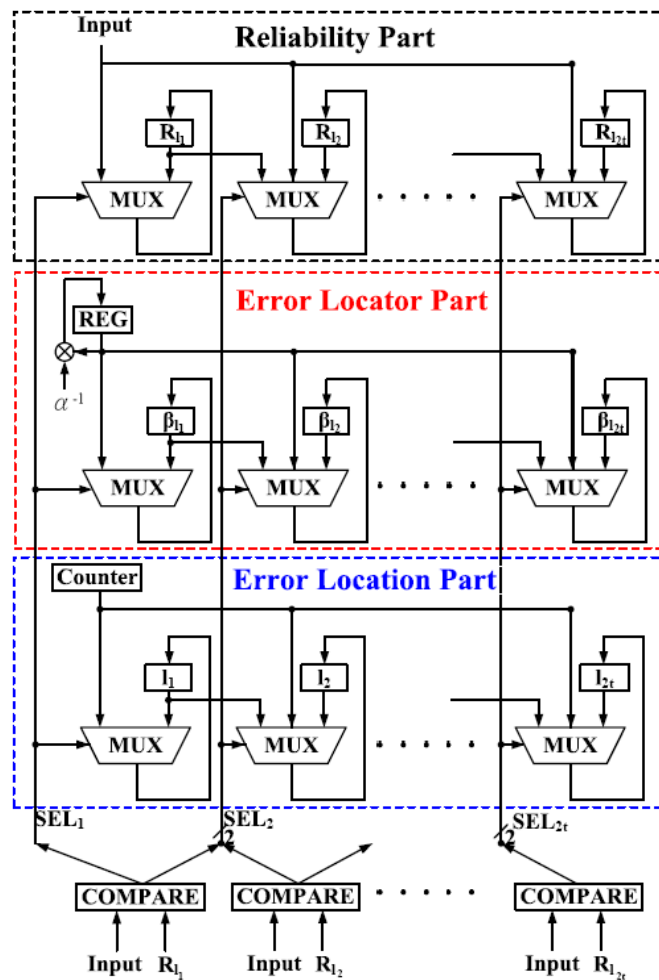


Fig9. Error Locator Evaluator Architecture for Serial Input

Error locator evaluator classifies the soft inputs to choose  $2t$  least reliable inputs as the candidate reliabilities  $R_{11}, R_{12}, \dots, R_{12t}$ . Their corresponding error locators  $\beta_{li}$  and error locations  $l_i$  are also calculated and stored in registers. Error locator evaluator compares the soft inputs with  $R_{li}$ , and then generates the select signals  $SEL_i$  to control the multiplexers. In the  $i$ -th stage, if the input is smaller than  $R_{li-1}$ , the  $i$ -th stage value is updated with  $(i-1)$ -th stage value.

If the input is greater than  $R_{li-1}$  and smaller than  $R_{li}$ , the  $i$ -th stage value is updated with the input value. Otherwise, the  $i$ -th stage holds its current value.

## II. Compensation Error Magnitude Solver (CEMS)

The compensation error magnitude solver (CEMS) in Fig. 10 is employed to evaluate (13) while given Sodd and B. Totally  $2t^2$  registers are used to store each entry in the Bodd matrix. The initial value of registers in each row is set as B so that the output of the SQUARE will always be  $\beta_{li}^2$  for first  $t-1$  cycles. Iteratively multiplied by  $\beta_{li}^2$ , the bottom registers generate  $\beta_{li}^{2j+1}$  for  $i = 1 \sim 2t$  and  $j = 0 \sim t-1$ . Thus, totally only  $2t$  multipliers are used for Bodd calculation. After  $t-1$  cycles, Bodd is constructed and the registers will stop update. Matrix multiplication is evaluated in the following  $22t$  cycles. By counting  $\Gamma$  value, a heuristic search for all binary combinations can be completed. At each iteration, each  $\beta_{lij}$  value will be calculated with  $\gamma_i$ , and the solver can verify whether the geometrical sequence check stands or not. If  $\Delta_{odd}$  is a geometrical sequence, then  $\delta_i \times \delta_{i+2} = \delta_{i+1}^2$ . CEMS uses  $t$  multipliers to check the relation and uses a look up table (LUT) for looking for  $l_{loss}$  from  $\delta_1$ .

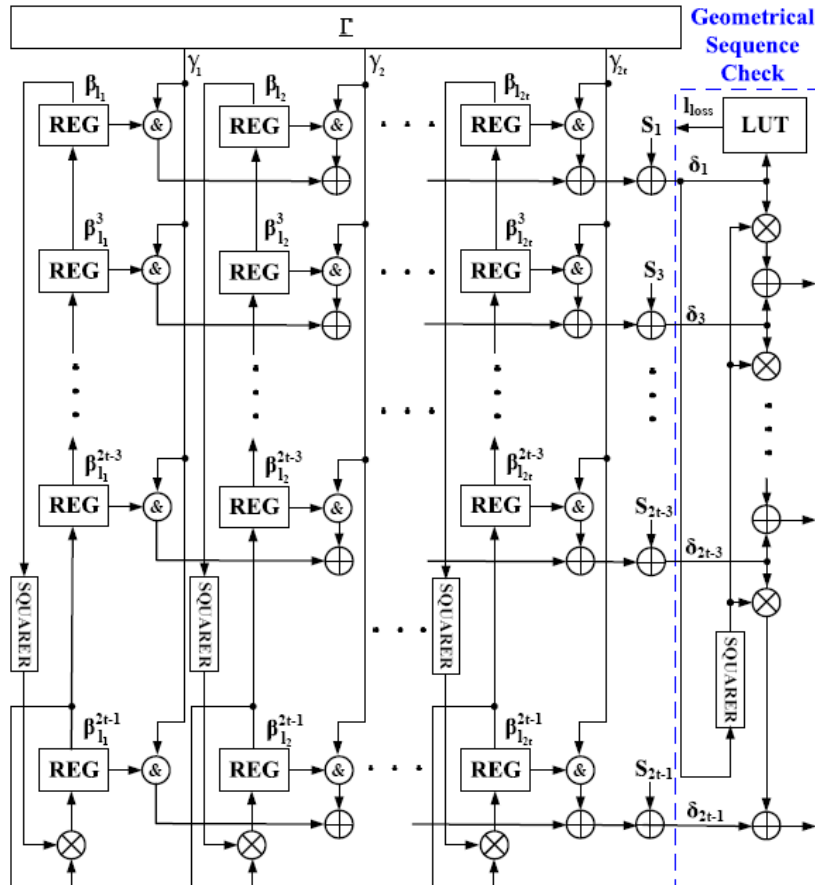


Fig10. Compensation Error Magnitude Solver Architecture

### III. Architecture Comparison

The architectures of a hard BCH decoder and the proposed soft BCH decoder are compared in TABLE II. In finite field operation, the complexity of a multiplier is much higher than a register. Because of fewer multipliers, the proposed soft BCH decoder with more registers and additional LUT has similar hardware complexity as the hard BCH decoder with inversionless Berlekamp-Massey (iBM) algorithm [B-11]. Moreover, searching error locations at error locator evaluator procedure leads to a lot of latency saving. Therefore, the proposed soft BCH decoder can provide higher throughput with almost the same hardware complexity as compared to the traditional hard BCH decoder. For example, for BCH (255,239) code, the proposed soft BCH decoder has 20 registers, 1 LUT and 5 multipliers while the hard BCH decoder has 12 registers and 9 multipliers. Furthermore, the proposed decoder also has only 53% latency as compared with traditional hard BCH decoder.

Table II. Comparison Table for A (n, k, t) BCH Code

	(n,k,t) Hard BCH with iBM	(n,k,t) Soft BCH Proposed	(255,239,2) Hard BCH with iBM	(255,239,2) Soft BCH Proposed
register	$5t + 2$	$2t^2 + 6t$	12	20
multiplier	$3t + 3$	$3t - 1$	9	5
constant multiplier	$3t$	$t + 1$	6	3
square	0	$2t + 1$	0	5
LUT	0	1	0	1
latency	$2n + 2t$	$n + 2^{2t} + t - 1$	514	272

### 3. SIMULATION AND IMPLEMENTATION RESULTS

Simulation and implementation results for our proposed soft BCH decoder are presented in this section. Fig. 11 shows the performance comparison for 2-error-correcting (255,239) BCH code under BPSK modulation in AWGN channel. The achieved coding gain is about 0.75dB over the hard BCH decoder at  $BER = 10^{-5}$ . Our proposed decoder can outperform 0.35dB and 0.2dB coding gain as compared with GMD [B-4] and sub-optimum MAP [B-7] respectively.

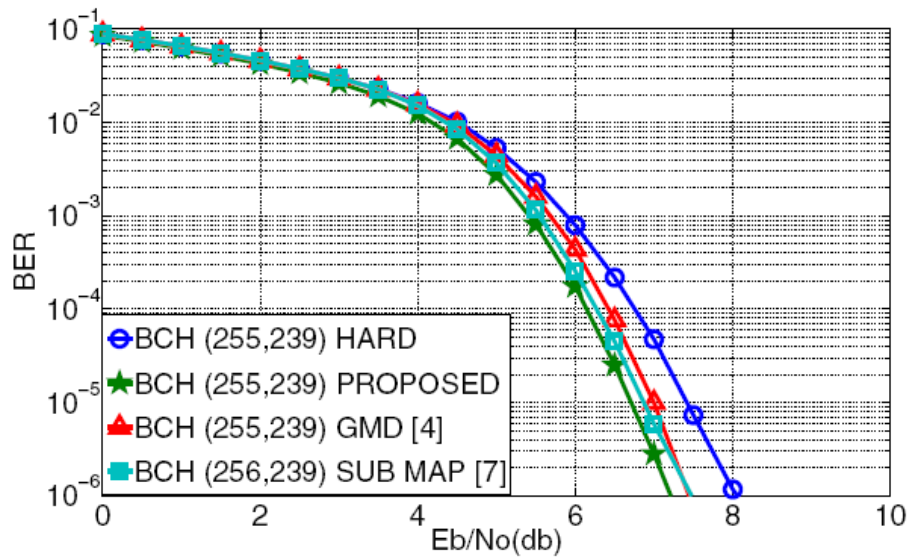


Fig11. Simulation results for BCH (255,239) code

The BCH (255,239) decoder is implemented with hard decision and soft decision methods and demonstrated in TABLE III. The hard BCH decoder uses iBM algorithm to solve key equation and needs Chien search to get error locations. Computing error locations without Chien search, the soft BCH decoder has almost half latency of the hard BCH decoder. Hence, the soft BCH decoder has much better throughputs than the hard BCH decoder. According to the post-layout simulations, the soft BCH decoder saves 47.1% clock cycle latency with similar gate count and operation frequency as compared with the hard BCH decoder in standard CMOS 90nm technology.

Table III. Summary of Implementation Results

	Hard BCH (255,239) t = 2	Soft BCH (255,239), t = 2
Technology	90nm	90nm
Architecture	iBM + Chien Search	CEMS w/o Chien Search
Operation Frequency	360MHz (Post Layout)	360MHz (Post Layout)
Core Area	14400 $\mu m^2$	13225 $\mu m^2$
Gate Count	4.38K	4.06K
Latency	514	272
Throughput	167.4Mb/s	316.3Mb/s

## 結論與討論

Among this year, we propose two modified FEC decoders for this low-power base-band processor and describe those contributions as follows:

A high-throughput and power-efficient LDPC decoder is presented. Utilizing the characteristic of variable-node-centric sequential scheduling, the proposed decoding algorithm could reduce the maximum iteration number without performance loss. In addition, the single pipelined architecture and modified CNU can save 73% message storage memory and decrease the sorter size, resulting in a low-complexity design. After implementation in 90nm technology, the test chip occupies 3.84 mm<sup>2</sup> area and supports maximum 11.5 Gbps data rate under 1.4V supply voltage.

We also provide an improved soft BCH decoder which performs better performance and comparable hardware complexity as compared to the conventional hard BCH decoder. The complexity is reduced by dealing with the least reliable bits, and the error correcting ability is enhanced by compensating an extra error outside the least reliable set. In addition, Chien search can be eliminated with a counter that evaluates error locations in the proposed error locator evaluator procedure. Thus, a lot of redundant decoding latencies can be eliminated and higher throughputs can be achieved without parallelism. From the experimental results of BCH (255,239) code, the proposed soft decoder can give 0.75dB coding gain over the hard BCH decoder at BER = 10<sup>-5</sup>. Also, it can achieve 316.3 Mb/s throughputs while reducing 7% gate-count as be compared with the 167.4Mb/s traditional hard BCH decoder in CMOS 90nm technology.

#### 四、 參考文獻

- [A-1] R. G. Gallager, *Low-Density Parity-Check Codes*. Cambridge, MA: MIT Press, 1963.
- [A-2] D. J. C. MacKay and R. M. Neal, "Near Shannon limit performance of low density parity check codes," *Electron. Lett.*, vol. 33, no. 6, pp.457–458, Mar. 1997.
- [A-3] M. Mansour and N. Shanbhag, "High-throughput LDPC decoders," *IEEE Trans. on VLSI Systems*, vol. 11, no. 6, pp. 976–996, Dec. 2003.
- [A-4] Part 3: carrier sense multiple access with collision detection (CSMA/CD) access method and physical layer specifications, *IEEE Std. P802.3an-2006*, Sept. 2006.
- [A-5] Part 15.3: wireless medium access control (MAC) and physical layer (PHY) specifications for high rate wireless personal area networks (WPANs), *IEEE Std. P802.15.3c-DF8*, 2009.
- [A-6] J. Zhang and M. Fossorier, "Shuffled iterative decoding," *IEEE Transactions on Communications*, vol. 53, no. 2, pp. 209–213, Feb. 2005.
- [A-7] Y. K. Lin, C. L. Chen, Y. C. Liao, and H. C. Chang, "Structured LDPC codes with low error floor based on peg tanner graphs," in *IEEE Int. Sympo. Circuits and Systems (ISCAS'08)*, May 2008, pp. 1846–1849.
- [A-8] A. Darabiha, A. C. Carusone, and F. R. Kschischang, "A 3.3-Gbps bitserial block-interlaced min-sum LDPC decoder in 0.13- $\mu$ m CMOS," in *Proc. IEEE CICC'07*, Sept. 2007, pp. 459–462.
- [A-9] X. Y. Shih, C. Z. Zhan, C. H. Lin, and A. Y. Wu, "A 19-mode 8.29mm<sup>2</sup> 52-mW LDPC decoder chip for IEEE 802.16e system," in *Proc. Int. Sympo. VLSI Circuits (SOVC'07)*, June 2007, pp. 16–17.
- [B-1] C. R. Baugh and B. A. Wooley, "Theory and Practice of Error Control Codes", Addison-Wesley, 1983.
- [B-2] Framing Structure, Channel Coding and Modulation for Digital Television Terrestrial Broadcasting System, *NSPRC Std. GB 20 600-2006*, 2007.
- [B-3] Digital Video Broadcasting (DVB) Second Generation System for Broadcasting, Interactive Services, News Gathering and Other Broadband Satellite Applications, *ETSI Std. En 302 307*, 2005.
- [B-4] G. D. Forney, "Generalized Minimum Distance Decoding," *IEEE Trans. Inform. Theory*, vol. 12, p. 125V131, Apr. 1966.
- [B-5] D. Chase, "A Class of Algorithms for Decoding Block Codes with Channel Measurement Information," *IEEE Trans. Inform. Theory*, vol. IT-18, p. 170V182, Jan. 1972.
- [B-6] M. Lalam, K. .Amis, D. Lerous, D. Feng, and J. Yuan, "An Improved Iterative Decoding Algorithm for Block Turbo Codes," *IEEE Int. Sympo. on Info. Theory*, pp. 2403–2407, July 2006.

- [B-7] F. Therattil and A. Thangaraj, "A Low-complexity Soft-decision Decoder for Extended BCH and RS-Like Codes," *IEEE Trans. Inform. Theory*, p. 1320V132, Sept. 2005.
- [B-8] W. J. ReidIII, L. L. Joiner, and J. J. Komo, "Soft Decision Decoding of BCH Codes Using Error Magnitudes," *IEEE Int. Symp. on Info. Theory*, p. 303, June 1997.
- [B-9] Y. Chen and K. Parhi, "Small Area Parallel Chien Search Architectures for Long BCH Codes," *IEEE Trans. on VLSI*, vol. 12, no. 5, pp. 545–549, May 2004.
- [B-10] J. Cho and W. Sung, "Strength-Reduced Parallel Chien Search Architecture for Strong BCH Codes," *IEEE Trans. on Circuits and Systems II*, vol. 55, no. 5, pp. 427–431, May 2008.
- [B-11] I. S. Reed, M. T. Shih, and T. K. Truong, "VLSI Design of Inverse-Free Berlekamp-Massey Algorithm," *Proc. Inst. Elect. Eng*, vol. 138, pp. 295–298, Sept. 1991.

## 五、計畫成果自評

在此計畫執行第二年中，我們提供兩個可應用於無線通訊之低功耗基頻處理器元件，其中分別為：

- 1) A 11.5-Gbps LDPC Decoder Based on CP-PEG Code Construction
- 2) An Improved Soft BCH Decoder with One Extra Error Compensation

下表四與五是為本研究團隊對於今年執行的計畫進度與各別研究規劃，在表中，我們完成了此兩研究的架構設計、下線與陸續完成量測；在此計畫第二年的研究成果，在未來，希望能為產業界、學術界盡一份心力。

表四 本年度(2009-2010)子計畫一之研究規劃

月份 (2009)	01	02	03	04	05	06	07	08	09	10	11	12
<b>LDPC</b>												
LDPC Paper Survey												
CP-PEG Code Construction												
Sequential Scheduling Design												
Algorithm Simulation												
Architecture Design												
Cell-Based Design												
Chip Layout												

表五 本年度(2009-2010)子計畫二之研究規劃

月份 (2009)	01	02	03	04	05	06	07	08	09	10	11	12
<b>An Improved Soft BCH Decoder with One Extra Error Compensation</b>												
Soft BCH Paper Survey												
Soft BCH Algorithm Development												
Soft BCH Algorithm Simulation												
Soft BCH Architecture Design												
Soft BCH Cell-Based Design												

感謝本計畫對我們的鼓勵，讓我們每年有穩定成長的貢獻，下者我們列掛有本計畫的論文：

1. Yi-Min Lin, Chih-Lung Chen, Hsie-Chia Chang, and **Chen-Yi Lee**, "A 26.9K 314.5Mbps Soft (32400,32208) BCH Decoder Chip for DVB-S2 System," has been revised by *IEEE Journal of Solid-State Circuits*



2. Yi-Min Lin, Hsie-Chia Chang, and **Chen-Yi Lee**, “An Improved Soft BCH Decoder with One Extra Error Compensation,” in *IEEE Int. Symposium on Circuits and Systems (ISCAS)*, France Paris, May 2010.
3. Yi-Min Lin, Chih-Lung Chen, Hsie-Chia Chang, and **Chen-Yi Lee**, “A 26.9K 314.5Mbps Soft (32400,32208) BCH Decoder Chip for DVB-S2 System,” in *IEEE Asia Solid State Circuits Conf. (ASSCC)*, Taiwan Taipei, Nov. 2009
4. Chic-Lung Chen, Kao-Shou Lin, Hsie-Chia Chang, Wai-Chi Fang, and **Chen-Yi Lee**, “A 11.5-Gbps LDPC Decoder Based on CP-PEG Code Construction,” in *IEEE ESSCIRC Proceedings*, Greece Athens, Sep. 2009

表六簡列出此計畫支持本研究室的相關研究成果，其中，有 9 件國內、外專利申請中、11 篇國際期刊與會議論文已發表於 IEEE（相關研究統計列出如附件所示），本研究室亦十分感謝國家科學委員會今年暨未來的持續支持與鼓勵。

表六 本計畫相關近年(2007-2010)研究貢獻

成果項目			96.01.01- 99.05.31	成果項目			96.01.01- 99.05.31
專利	申請	國內 (件數)	4	論文	期刊	國內 (件數)	0
		國外 (件數)	5			國外 (件數)	1+3*
		國內外合計件數	9			國內外合計件數	1+3*
	獲得	國內 (件數)	1		研討會	國內 (件數)	0
		國外 (件數)	0			國外 (件數)	3+4*
		國內外合計件數	1			國內外合計件數	3+4*

\*：今年研究量產的論文數

## 附件 - 2007-2010 本計畫相關之研究成果

### Patents:

1. 翁政吉、唐正浩、張錫嘉、李鎮宜，”應用於迭代解碼之多層級網路架構及其傳輸方法，”中華民國專利申請第 96149409 號，96 年 12 月 21 日。(公告號：200929892) (經濟部科專 95-EC-17-A-01-S1-048)
2. 陸志豪、林建青、李鎮宜、張錫嘉、許雅三，”多模多平行度資料交換方法及其裝置，”中華民國專利申請第 96146695 號，96 年 12 月 7 日。(公告號：200926612) (經濟部科專 93-EC-17-A-03-S1-0005)。
3. 陸志豪、廖彥欽、李鎮宜、許雅三、張錫嘉，”應用於低密度對稱檢查碼 (LDPC) 解碼器之運算方法及其電路，”中華民國專利申請第 096128039 號，96 年 7 月 31 日。(公告號：200906073) (經濟部科專 93-EC-17-A-03-S1-0005)
4. 陸志豪、林建青、李鎮宜、許雅三、張錫嘉，”用於通訊系統的資料交換裝置及方法，”中華民國專利申請第 096114252 號，96 年 4 月 23 日。(公告號：200832913) (國科會 NSC94-2220-E-009-027)
5. 李鎮宜、林建青、林凱立、張錫嘉，”用於更新低密度配類核對 (LDPC) 碼解碼器之核對節點的方法及其裝置，”中華民國專利發明 I291290 號，96 年 12 月 11 日。(國科會 NSC 93-2220-E-009-033)
6. 陸志豪、廖彥欽、李鎮宜、許雅三、張錫嘉，”應用於低密度對稱檢查碼 (LDPC) 解碼器之運算方法及其電路，”日本專利申請特願 2008-082997 號，97 年 3 月 27 日。(經濟部科專 93-EC-17-A-03-S1-0005)
7. Cheng-Chi Wong, Cheng-Hao Tang, Hsie-Chia Chang, and Chen-Yi Lee, ”Method and apparatus of multi-stage network for iterative decoding,” has been filed as U.S. Patent pending, 12/178987, July 24, 2008. ( pub. no. 20090160686 ) ( 經濟部科專 95-EC-17-A-01-S1-048 )
8. Chih-Hao Liu, Chien-Ching Lin, Chen-Yi Lee, Hsie-Chia Chang, and Yar-Sun Hsu, ”Multi-mode multi-parallelism data exchange method and thereof,” has been filed as U.S. Patent pending, 12/048101, March 13, 2008. ( pub. no. 20090146849 ) ( 經濟部科專 93-EC-17-A-03-S1-0005 )
9. Chih-Hao Liu, Yen-Chin Liao, Chen-Yi Lee, Hsie-Chia Chang, and Yar-Sun Hsu, ”Operating Method Applied to Low Density Parity Check (LDPC) Decoder and Circuit Thereof”, has been filed as U.S. Patent pending, 11/939119, November 13, 2007. ( pub. no. 20090037799 ) ( 經濟部科專 93-EC-17-A-03-S1-0005 )

10. Chih-Hao Liu, Chih-lung Chen, Chen-Yi Lee, Yar-Sun Hsu, and Hsie-Chia Chang, "Method and Apparatus for Switching Data in Communication System," has been filed as U.S. Patent pending, 802028, May 18, 2007. (pub. no. 20080198843) (國科會 NSC94-2220-E-009-027)

## **Journals:**

1. Cheng-Chi Wong, Ming-Wei Lai, Chien-Ching Lin, Hsie-Chia Chang, and Chen-Yi Lee, "Turbo Decoder Using Contention-Free Interleaver and Parallel Architecture," has been accepted by *IEEE J. Solid-State Circuits*.
2. Chih-Hao Liu, Chien-Ching Lin, Shau-Wei Yen, Chih-Lung Chen, Hsie-Chia Chang, Chen-Yi Lee, Yar-Sun Hsu and Shyh-Jye Jou, "Design of a Multimode QC-LDPC Decoder Based on Shift-Routing Network," *IEEE Trans. Circuits Syst. II*, , vol.56, no.9, pp.734-738, September 2009. (SCI/EE)
3. Hsie-Chia Chang, Chien-Ching Lin, Fu-Ku Chang, and Chen-Yi Lee, "A Universal VLSI Architecture for Reed-Solomon Error-and-Erasure Decoders," *IEEE Trans. Circuits Syst. I*, vol.56, no.9, pp.1960-1967, September 2009.
4. Chih-Hao Liu, Shau-Wei Yen, Chih-Lung Chen, Hsie-Chia Chang, Chen-Yi Lee, Yar-Sun Hsu, and Shyh-Jye Jou, "An LDPC Decoder Chip Based on Self-Routing Network for IEEE 802.16e Applications," *IEEE J. Solid-State Circuits*, vol. 43, no. 3, pp. 684~694, March 2008. (SCI/EE)

## **Conference:**

1. Yi-Min Lin, Hsie-Chia Chang, and Chen-Yi Lee, "An Improved Soft BCH Decoder with One Extra Error Compensation," *has been accepted by IEEE Int. Symposium on Circuits and Systems (ISCAS) 2010.* (EI)
2. Yi-Min Lin, Chih-Lung Chen, Hsie-Chia Chang, and Chen-Yi Lee, "A 26.9K 314.5Mbps Soft (32400, 32208) BCH Decoder Chip for DVB-S2 System," *IEEE Asian Solid-State Circuits Conference (A-SSCC)*, Hsinchu, Taiwan, November 2009, pp.373-376, (EI)
3. Shao-Wei Yen, Ming-Chih Hu, Chih-Lung Chen, Hsie-Chia Chang, Shyh-Jye Jou, and Chen-Yi Lee, "A 0.92mm<sup>2</sup> 23.4mW Fully-Compliant CTC Decoder for WiMAX 802.16e Application," *IEEE Custom Integrated Circuits Conference (CICC)*, , San Jose, California, October 2009, pp.191-194. (EI).
4. Chih-Lung Chen, Kao-Shou Lin, Hsie-Chia Chang, Wai-Chi Fang, and Chen-Yi Lee, "A 11.5-Gbps LDPC Decoder Based on CP-PEG Code Construction," *European Solid-State*

*Circuits Conf. (ESSCIRC)*, Athens, Greece, September 2009.

5. Chih-Hao Liu, Chien-Ching Lin, Hsie-Chia Chang, Chen-Yi Lee, and Yar-Sun Hsu, "Multi-mode Message Passing Switch Networks Applied for QC-LDPC Decoder," in *IEEE Int. Symposium on Circuits and Systems (ISCAS)*, Seattle, U.S., May 2008, pp.752-755. (EI)
6. Cheng-Chi Wong, Cheng-Hao Tang, Ming-Wei Lai, Yan-Xiu Zheng, Chien-Ching Lin, Hsie-Chia Chang, Chen-Yi Lee, and Yu-T. Su, "A 0.22nJ/b/iter 0.13 $\mu$ m Turbo Decoder Chip Using Inter-Block Permutation Interleaver," in *IEEE Custom Integrated Circuits Conference (CICC)*, San Jose, California, October 2007, pp.273-276.
7. Dah-Jia Lin, Chien-Ching Lin, Chih-Lung Chen, Hsie-Chia Chang, and Chen-Yi Lee, "A Low-Power Viterbi Decoder Based on Scarce State Transition and Variable Truncation Length," in *IEEE Int. Symposium on VLSI Design, automation and Test (VLSI-DAT)*, Hsinchu, Taiwan, April 2007, pp.99~102. (EI)