2006 DAC/ISSCC Student-Design-Contest

CONCEPTUAL CATEGORY
1st Place

2006


Kazunori Shimizu


Tatsuyuki Ishikawa


Nozomu Togawa


Takeshi Ikenaga


Satoshi Goto

ASIC Implementation of LDPC Decoder Accelerating Message-Passing Schedule

Waseda University, Kita-Kyushu, Japan

ASIC Implementation of LDPC Decoder Accelerating Message-Passing Schedule

Kazunori Shimizu†

Tatsuyuki Ishikawa†, Nozomu Togawa‡,

Takeshi Ikenaga†, and Satoshi Goto†.

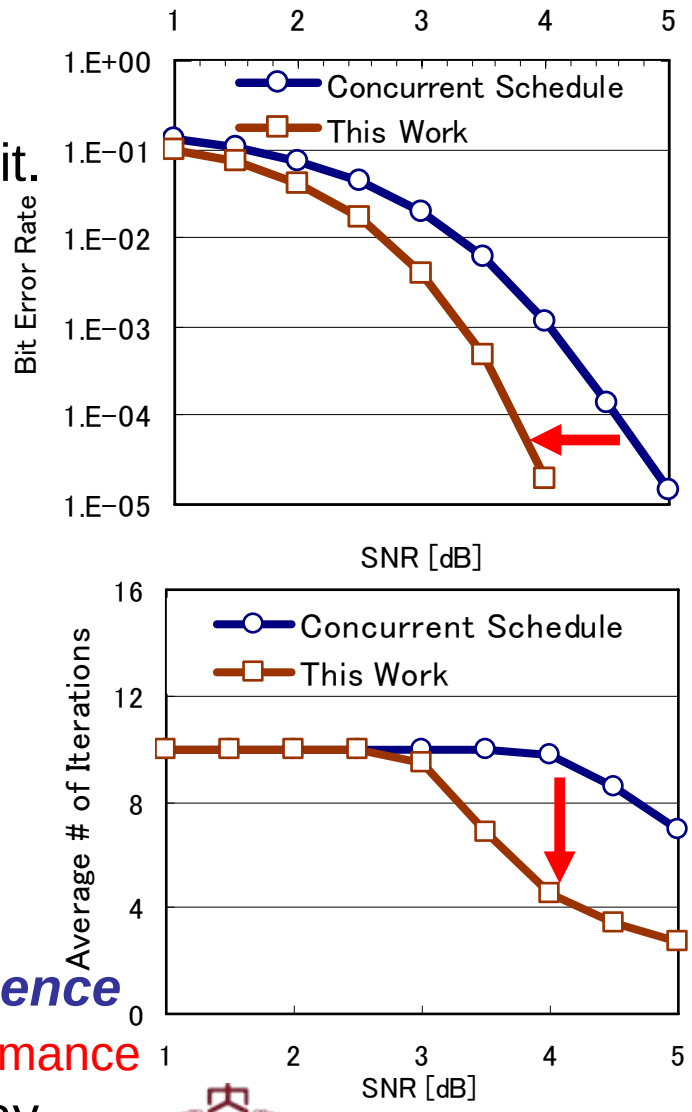
† Graduate School of IPS, Waseda University
‡ Department of Computer Science, Waseda University

Research Background

Low-Density Parity-Check Code:
is an error correcting code which achieves information rates very close to the Shanon limit.

Message-Passing Algorithm:
is an iterative algorithm for decoding LDPC codes, is composed of row operation and column operation.
Inherent parallelism of the algorithm makes it suitable for hardware design.

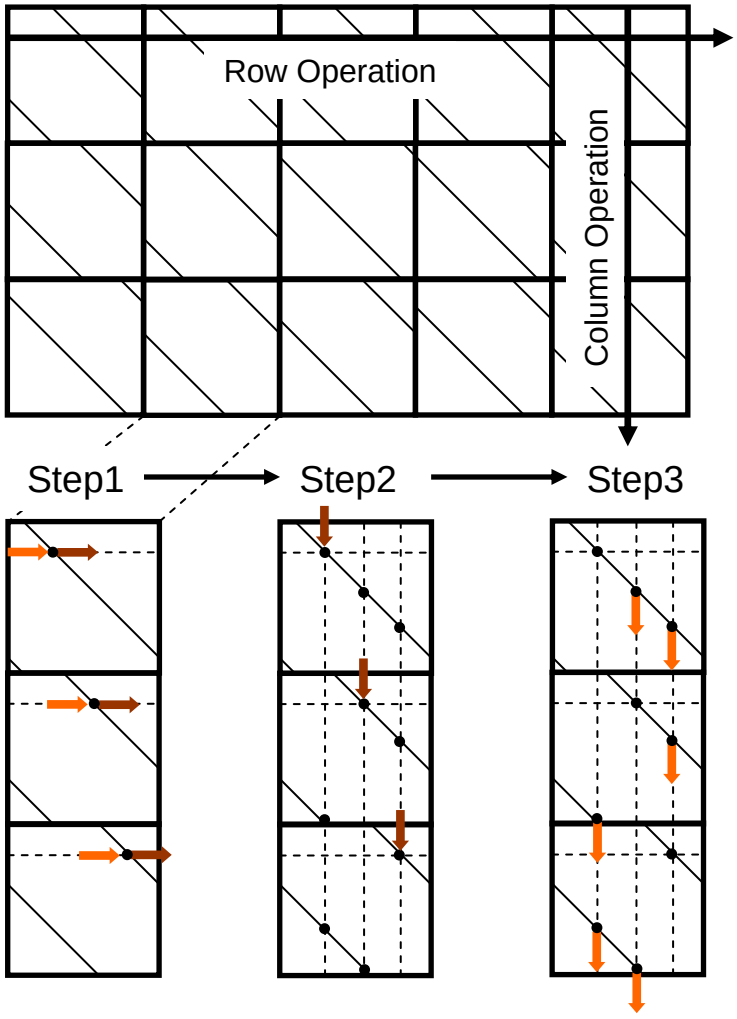
Motivation:
Requirement is performance gain
(bit error performance and data rate)
with a small hardware overhead.



This Work: Accelerating decoding convergence
enables the decoder to improve Bit Error Performance and Decoding Throughput within a limited delay.

Accelerating Message-Passing Schedule

Parity check matrix for hardware sharing

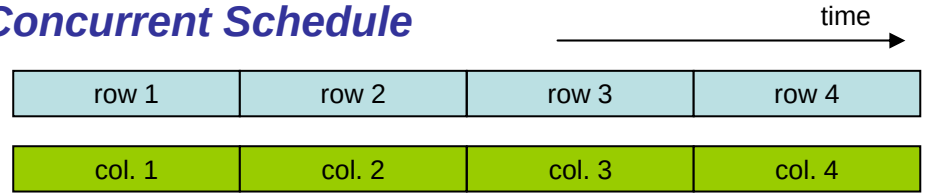


message alpha

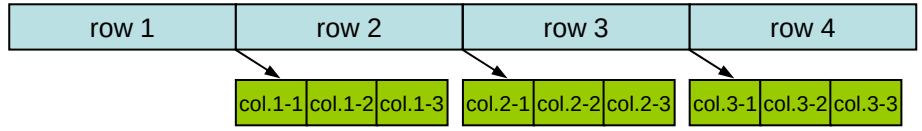
Row Operation ⇄ Column Operation

message beta

Concurrent Schedule



This Work



- ✓ The updated message by the row operations are fed into the column operation immediately after the row operations.
- ✓ The number of column operations increases three times compared to the concurrent schedule.

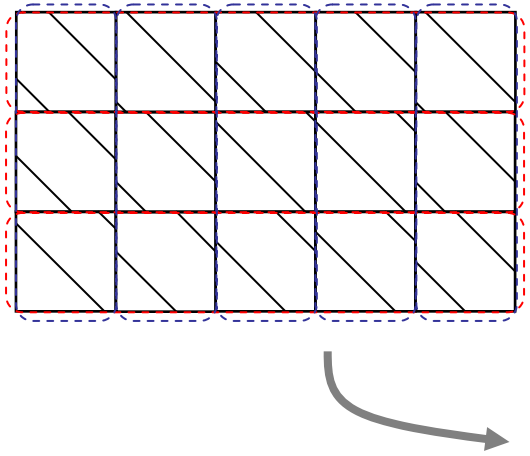
➡ message alpha updated by row operation

➡ message beta updated by column operation.

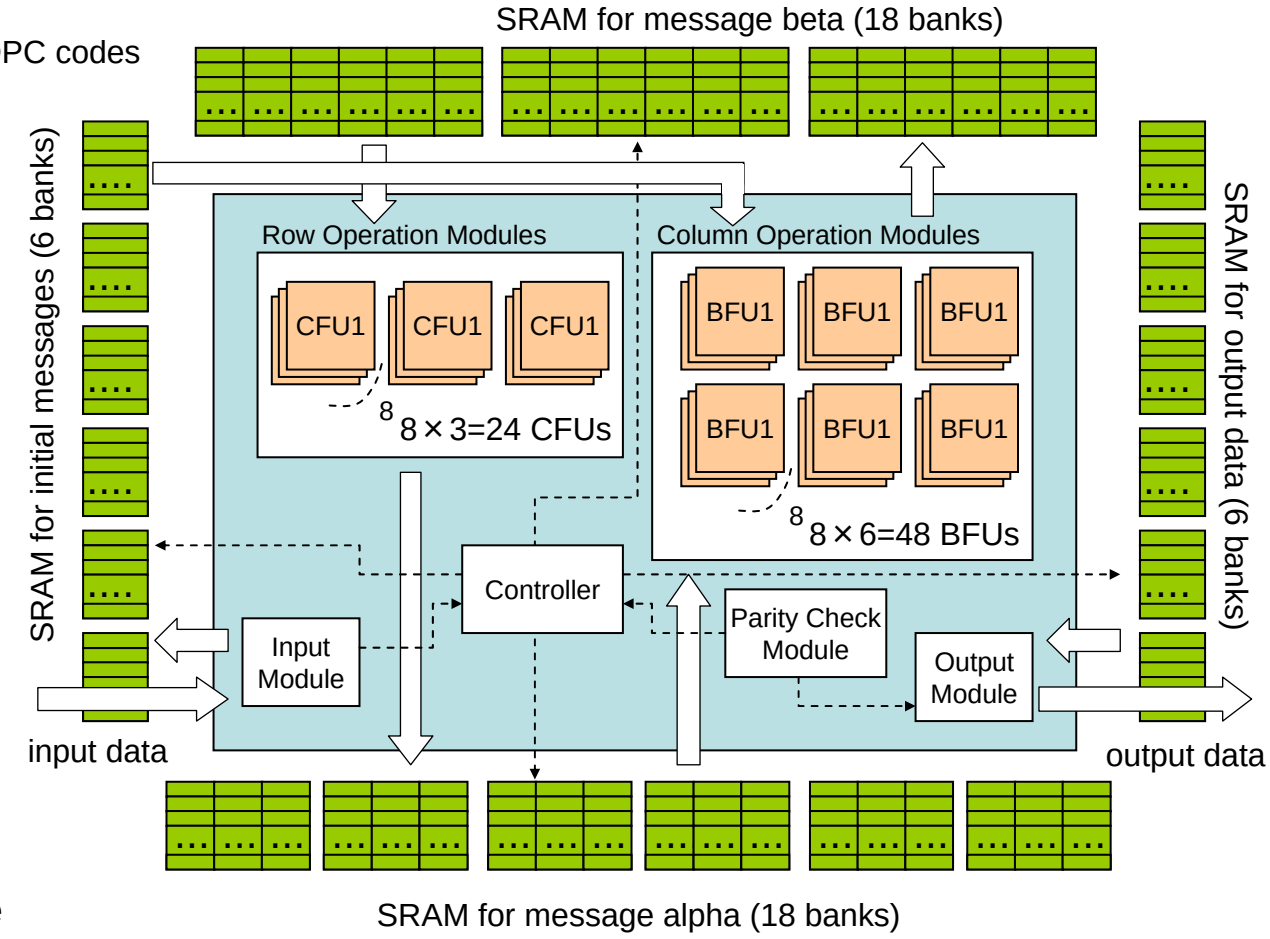
Block Diagram of LDPC Decoder

LDPC Decoder

Parity Check Matrix for (3,6)-regular LDPC codes



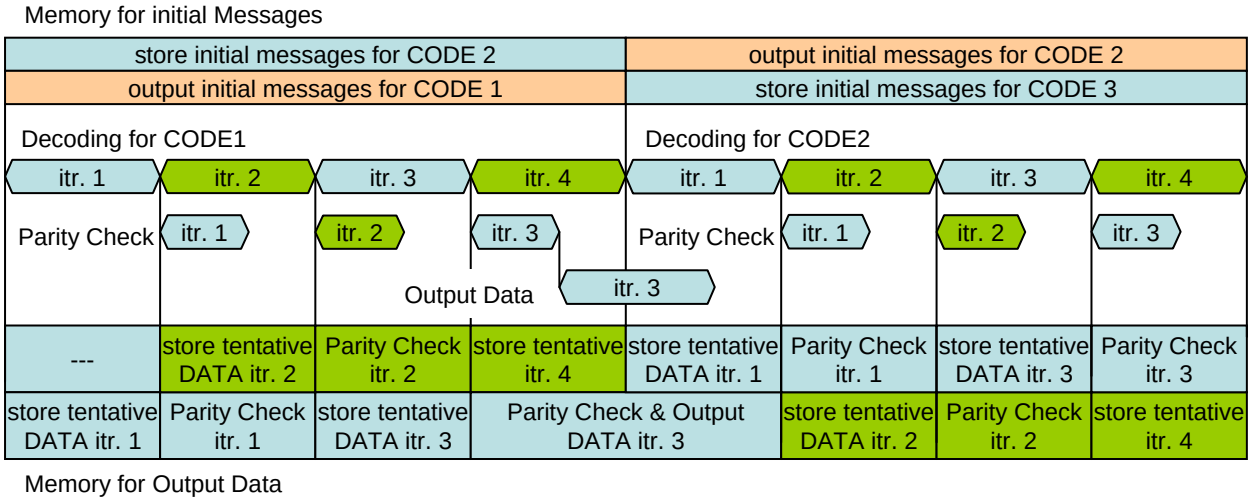
- ✓ The decoder for the (3,6)-regular LDPC code is composed of a three row operation module and a six column operation module.
- ✓ Within each module, there are eight functional units to operate eight row and eight column operations in parallel.



BFU : Bit Functional Unit
CFU : Check Functional Unit

Experimental Result

Timing diagram of the LDPC decoder



- ✓ SRAM for initial message and output data stores two LDPC codewords so that the decoding hardware can be fully used.
- ✓ Parity check in each iteration.
- ✓ Switches the decoding codeword when all bit errors in each codeword are decoded correctly.

- Synthesis Results (TSMC 0.18 μ m CMOS)

	Logic[μ m^2]	SRAM[μ m^2] (# of Banks)	Total[μ m^2]
Concurrent Schedule	3,448,747	7,113,932 (48)	10,562,680
Proposed Schedule	4,124,134	7,113,932 (48)	11,238,066

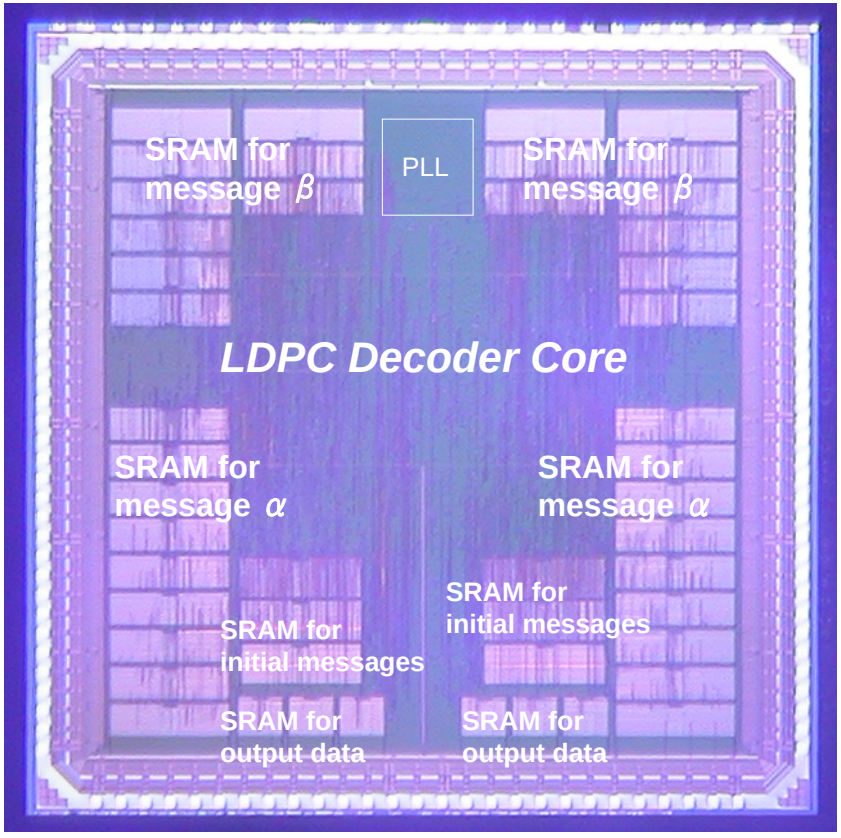
- Experimental Results (SNR=4.5, Iteration limit=10, @120[MHz])

	Power [mW]	Average #of Iterations	Throughput [Mbps]	Power/Thr. [mW/Mbps]	Bit Error Rate
Concurrent Schedule	327.4	8.536	48	6.8	0.000141
Proposed Schedule	528.9	3.391	122	4.3	0.000000

Synthesis & Experimental Results

- ✓ Decoder core based on the proposed schedule increases about 6% in the total area.
- ✓ Decoding throughput is about 2.54 times faster compared to the concurrent schedule.
- ✓ Power efficiency [mW/Mbps] can be improved up to 37%.

LDPC Decoder Chip



Summary of the LDPC Decoder Chip

Code length	3,072 [bits]
Code rate	0.5, (3,6)-regular
Design process	0.18um, 6Metal,CMOS
Chip size	5.0mm* 5.0mm
Gate count	96,945 gates (Decoder Core)
# of CFU	3 * 8 = 24
# of BFU	6 * 8 = 48
Total SRAM Area	7,113,932 [μ m^2]
PLL Area	266,136 [μ m^2]
Chip Density	49%
Clock Frequency	120 [MHz] (Max)
Throughput	122 [Mbps] (@120MHz, SNR=4.5)
Power Consumption	529 [mW] (@120MHz, 1.62V)

Chip Micrograph