# A 4.68Gb/s Belief Propagation Polar Decoder with Bit-Splitting Register File

Youn Sung Park, Yaoyu Tao, Shuanghong Sun, Zhengya Zhang
Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor

## Abstract

A 1.48mm$^2$ 1024-bit belief propagation polar decoder is designed in 65nm CMOS. A unidirectional processing reduces the memory size to 45Kb, and simplifies the processing element. A double-column 1024-parallel architecture enables a 4.68Gb/s throughput. A bit-splitting latch-based register file accommodates logic in memory for an 85% density. The architecture and circuit techniques reduce the power to 478mW for an efficiency of 15.5pJ/b/iteration at 1.0V. At 475mV, the efficiency is improved to 3.6pJ/b/iteration for a throughput of 780Mb/s.

## Introduction

The recently invented polar codes are provably capacity-achieving [1], and their regular structure promises energy-efficient codec designs. The successive cancellation (SC) decoding of polar codes is sequential in nature [1]. The latest SC decoder [2] provides a throughput up to 1Gb/s, but requires at least 500Kb memory. We design a parallel belief propagation (BP) polar decoder to achieve 4.68Gb/s using only 45Kb memory. An optimal code selection enables a competitive error correcting performance compared to LDPC codes, with no error floor observed at low error rates (Fig. 6).

## Highly Parallel Belief Propagation Polar Decoder

The encoding and decoding of polar codes are done by processing elements (PE) connected in a factor graph (Fig. 1). What is unique about polar codes is that the source bits either carry information or are frozen to 0. An ($N$, $K$) polar code has $K$ information bits and $N - K$ frozen bits. Variable code rate is supported with no overhead. An $N$-bit encoding is done by passing $N$ bits through $\log_2 N$ stages of encoding PEs (Fig. 1). Each encoding PE consists of an XOR and a pass-through. The encoded bits are sent over a communication channel, and the soft information for each bit, in log-likelihood ratio (LLR), is received from the channel. In BP decoding [3], LLRs are passed iteratively from left to right and then from right to left through decoding PEs to compute the likelihood of the information bits (Fig. 1). The bidirectional PE performs 3 compare-selects and 3 sums to compute a pair of left-bound messages (L messages) and a pair of right-bound messages (R messages).

The regular wiring structure between stages permits a highly parallel decoder implementation without the complex wiring seen in LDPC decoders. A single-column decoder architecture for the 1024-bit polar code (Fig. 2(a)) consists of 512 bidirectional PEs to compute one stage in parallel, with a 45Kb left-bound message memory (L memory) and a 45Kb right-bound message memory (R memory). Each memory stores 9 rows (one row for each stage, and the last stage is not stored) of 512 pairs of 5-bit LLR messages. Based on synthesis in 65nm CMOS, the single-column decoder occupies 2.02mm$^2$ at an 85% density, providing a decoding throughput of 945Mb/s at 250MHz using 15 iterations at the worst-case 0.9V and 125°C. We apply architecture and circuit techniques to further improve the throughput and reduce the area of the decoder.

## Memory Reduction and Throughput Enhancement

The BP decoding of the 1024-bit polar code specifies one right-bound message propagation (R propagation) from stage 0 to 8, followed by one left-bound message propagation (L propagation) from stage 9 to 0 for a total of 19 stages to complete one iteration. In each stage, one row (5Kb) of messages is read from each of L and R memory, and one row is written to each memory (Fig. 2(a)). The PE is bidirectional: it updates both L and R messages, whereas the message propagation is unidirectional to allow only one of L or R messages to be propagated. Therefore, we design a unidirectional PE to match the unidirectional propagation to update only L messages in L propagation, and only R messages in R propagation, thereby reducing the complexity of the PE to 2 compare-selects and 2 sums. The unidirectional processing allows L messages and R messages to share only one 45Kb memory (Fig. 2(b)), reducing the memory size by 50% and logic complexity by 33% without sacrificing throughput. Synthesis results show that the area is reduced by 35%, and the critical path is shortened to 3.5ns.

The critical path of the unidirectional decoder architecture runs through the pipeline registers, router, PE, and returns to the registers (Fig. 3(a)). Within the critical path, the processing and routing delays are relatively short, benefiting from the compact unidirectional PE design and the regular wiring in polar codes. We design a double-column architecture (Fig. 3(b)) for a better utilization of a clock period. This design increases the clock period from 3.5ns to 4ns – a rather small increase for shortening the iteration latency from 19 to 10 cycles, and improving the throughput by 66%. The message memory is split to a 25Kb bank0 and a 20Kb bank1 to support double-column processing, but the total memory size remains constant. Twice as many PEs and routers are used, but the compact PE and the regular wiring increase the area of the decoder by only 28%.

## High-Density Bit-Splitting Register File

The double-column architecture accesses 20Kb from memory every cycle. We design a dual-port latch-based register file for the wide and shallow memory (Fig. 4). Replacing register with latch and sharing of read and write word lines reduce the area and power compared to a distributed register memory. A simple sequential addressing and the elimination of column multiplexing improve the array efficiency and performance compared to SRAM. The latch-based register files for bank0 and bank1 occupy 1.7mm×0.12mm and 1.7mm×0.1mm, respectively, each providing 5Kb read ports and 5Kb write ports along one 1.7mm side. The ultra-dense port placement prevents a successful integration due to an insufficient number of routing tracks (Fig. 4). To relieve the congestion, we split the register file to bit rows to provide more tracks, and allocate PEs between bit rows to take advantage of locality and compression. The compare-select and add are done at bit level, right next to the bit memory, and the number of output wires over to the next bit row is substantially reduced. The new bit-splitting register file supports a denser integration of memory and logic than a standard register file, resulting in a final decoder area of 1.48mm$^2$ with a high density of 85%.

## Chip Implementation and Measurement Results

A double-column BP polar decoder test chip incorporating bit-splitting register file was fabricated in a TSMC 65nm CMOS process (Fig. 5). The 2.16mm×1.6mm test chip includes a 1.8mm×0.82mm polar decoder core, and additive white Gaussian noise generators and error checkers for measuring the error correcting performance. The test chip is fully functional, and its frame error rate and power measurements are graphed in Fig. 6. At room temperature and a nominal 1.0V supply, the chip operates at a maximum frequency of 300MHz for a throughput of 2.05Gb/s using 15 iterations. With a simple early termination scheme based on agreement of 3 consecutive hard decisions, the average iteration count is lowered to 6.57 (including convergence detection latency) at a 4.0dB SNR with no loss in error correcting performance (Fig. 6). Early termination enables a higher throughput of 4.68Gb/s at 478mW, or 15.5pJ/b/iteration. As the first BP polar decoder in silicon, the chip demonstrates a 34×, 2.8×, and 5.2× improvement in throughput, energy efficiency, and area efficiency, respectively, over the latest SC polar decoder ASIC [4] (normalized to 65nm and 1.0V, Fig. 7). Scaling the supply voltage to 475mV reduces the throughput to 780Mb/s for an improved energy efficiency of 3.6pJ/b/iteration. The results demonstrate the potential of polar codes for future communication and storage systems.

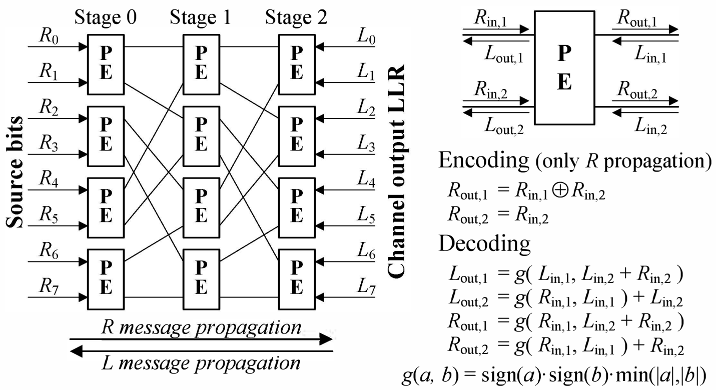
## Acknowledgements

This work is funded by NSF CCF-1054270 and Broadcom Foundation. We thank Andrew Blanksby, Matthias Korb, Medhi Hatamian, Pascal Urard, David Blaauw and Dennis Sylvester for advice.

## References

[1] Arikan, *Trans. Inf. Theory*, Jul. 2009.
[2] Sarkis *et al.*, *JSAC*, submitted, 2013.
[3] Arikan, *Commun. Lett.*, Jun. 2008.
[4] Mishra *et al.*, *ASSCC*, Nov. 2012.
[5] Leroux *et al.*, *TSP*, Jan. 2013.
[6] Park *et al.*, *VLSI Symp.*, Jun. 2012.

978-1-4799-3328-0/14/$31.00 ©2014 IEEE


Stage 0
Stage 1
Stage 2
$R_0$ $R_1$ $R_2$ $R_3$ $R_4$ $R_5$ $R_6$ $R_7$
$L_0$ $L_1$ $L_2$ $L_3$ $L_4$ $L_5$ $L_6$ $L_7$
PE
Source bits
Channel output LLR
R message propagation
L message propagation
$R_{\text{in},1}$ $L_{\text{out},1}$ $R_{\text{in},2}$ $L_{\text{out},2}$
$R_{\text{out},1}$ $L_{\text{in},1}$ $R_{\text{out},2}$ $L_{\text{in},2}$

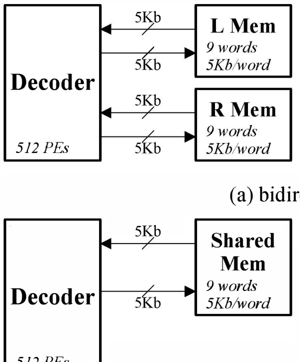
Encoding (only $R$ propagation)

$$R_{\text{out},1} = R_{\text{in},1} \oplus R_{\text{in},2}$$
$$R_{\text{out},2} = R_{\text{in},2}$$

Decoding

$$L_{\text{out},1} = g(\, L_{\text{in},1},\, L_{\text{in},2} + R_{\text{in},2}\,)$$
$$L_{\text{out},2} = g(\, R_{\text{in},1},\, L_{\text{in},1}\,) + L_{\text{in},2}$$
$$R_{\text{out},1} = g(\, R_{\text{in},1},\, L_{\text{in},2} + R_{\text{in},2}\,)$$
$$R_{\text{out},2} = g(\, R_{\text{in},1},\, L_{\text{in},1}\,) + R_{\text{in},2}$$
$$g(a,\, b) = \text{sign}(a)\cdot\text{sign}(b)\cdot\min(|a|,|b|)$$

Fig. 1. Factor graph representation of an 8-bit polar code and processing elements (PE) for encoding and decoding.



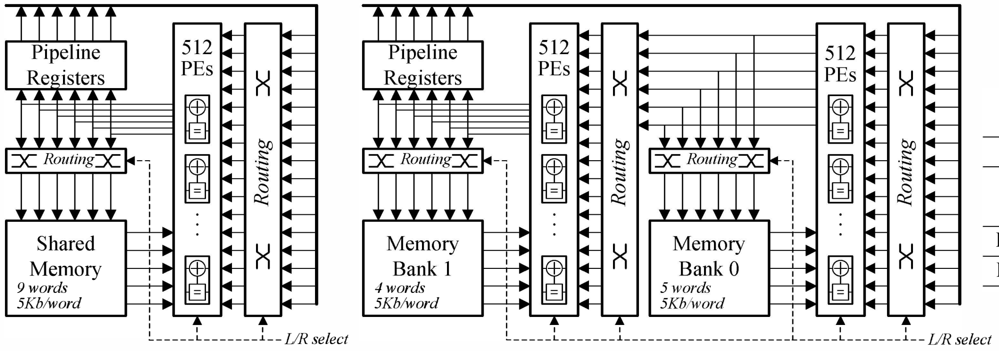
| Schedule | R propagation | L propagation |
|---|---|---|
| Cycle | 0 1 ··· 8 | 9 10 ··· 18 |
| Stage | 0 1 ··· 8 | 9 8 ··· 0 |
| Read $R_{\text{mem}}$ | - R0 ··· R7 | R8 R7 ··· - |
| Read $L_{\text{mem}}$ | L0 L1 ··· L8 | - L8 ··· L0 |
| Write $R_{\text{mem}}$ | R0 R1 ··· R8 | - R8 ··· - |
| Write $L_{\text{mem}}$ | - L0 ··· L7 | L8 L7 ··· - |

(a) bidirectional architecture

| Schedule | R propagation | L propagation |
|---|---|---|
| Cycle | 0 1 ··· 8 | 9 10 ··· 18 |
| Stage | 0 1 ··· 8 | 9 8 ··· 0 |
| Read Mem | L0 L1 ··· L8 | R8 R7 ··· - |
| Write Mem | R0 R1 ··· R8 | L8 L7 ··· - |

(b) unidirectional architecture

Fig. 2. Comparison of single-column (a) bidirectional and (b) unidirectional architecture.



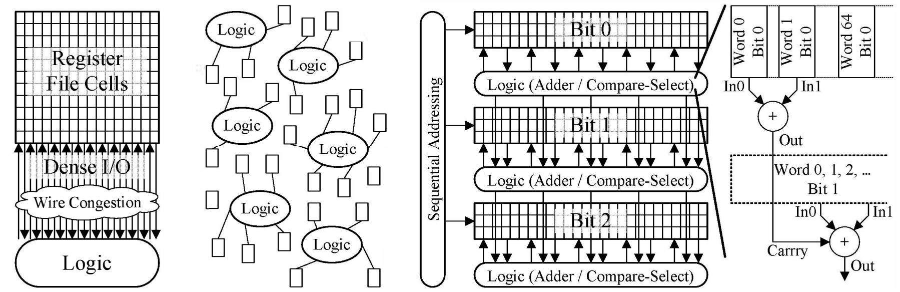
(a) single-column architecture

(b) double-column architecture

| | Single-Column | Double-Column |
|---|---|---|
| # of PEs | 512 | 1024 |
| Memory | 45Kb<br>(1 bank) | 45Kb<br>(2 banks) |
| Latency - R prop. | 9 cycles | 5 cycles |
| Latency - L prop. | 10 cycles | 5 cycles |
| Iteration Latency | 19 cycles | 10 cycles |

(c) architecture comparison

Fig. 3. (a) single-column and (b) double-column unidirectional architecture and (c) their comparison.



Standard register file
*Routing congestion*

Distributed registers
*Low efficiency*

Bit-splitting register file
*Logic in memory enables locality and compression*

Fig. 4. Bit-splitting register file compared to standard register file and distributed registers.

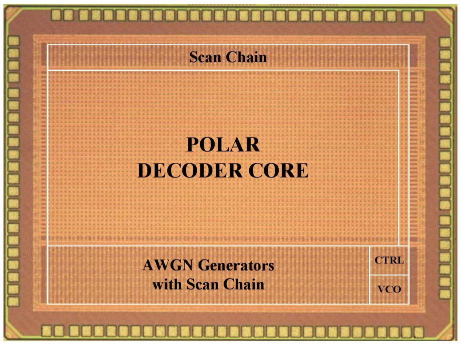
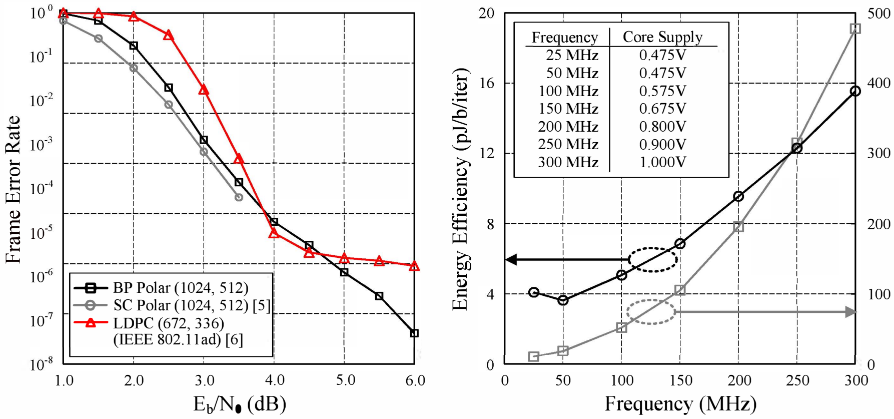


Fig. 5. Chip microphotograph.



| Frequency | Core Supply |
|---|---|
| 25 MHz | 0.475V |
| 50 MHz | 0.475V |
| 100 MHz | 0.575V |
| 150 MHz | 0.675V |
| 200 MHz | 0.800V |
| 250 MHz | 0.900V |
| 300 MHz | 1.000V |

Fig. 6. Frame error rate (FER) performance comparison, and measured power consumption and energy efficiency of the BP polar decoder at the minimum supply voltage for each clock frequency. (BP polar decoding using maximum 15 iterations with early termination enabled.)

| | This Work | | Mishra [4] |
|---|---|---|---|
| Code | BP Polar | | SC Polar |
| Block Length | 1024 | | 1024 |
| Process [nm] | 65 | | 180 |
| Core Area [mm$^2$] | 1.476 | | 1.71 |
| Utilization | 85% | | |
| Supply [V] | 1.0 | 0.475 | 1.3 |
| Frequency [MHz] | 300 | 50 | 150 |
| Power [mW] | 477.5 | 18.6 | 67 |
| Iteration | 6.57[a] | 6.57[a] | |
| Throughput [Mb/s] | 4676 | 779.3 | 49 |
| Energy Eff. [pJ/b] | 102.1 | 23.8 | 1367 |
| [pJ/b/iter] | 15.54 | 3.63 | |
| Area Eff. [Mb/s/mm$^2$] | 3168 | 528.0 | 28.65 |
| *Normalized to 65nm, 1.0V* | | | |
| Throughput [Mb/s] | 4676 | 779.3 | 135.7 |
| Energy Eff. [pJ/b] | 102.1 | 23.8 | 292.2 |
| Area Eff. [Mb/s/mm$^2$] | 3168 | 528.0 | 608.5 |

[a] Average decoding iteration at 4.0dB with early termination enabled

Fig. 7. Chip summary and comparison.