

Using Embedded Dynamic Random Access Memory to Reduce Energy Consumption of Magnetic Recording Read Channel

Ningde Xie¹, Tong Zhang¹, and Erich F. Haratsch²

¹ECSE Department, Rensselaer Polytechnic Institute, Troy, NY 12180 USA

²LSI Corporation, Allentown, PA 18109 USA

Although the performance of a magnetic recording read channel can be improved by employing advanced iterative signal detection and coding techniques, the method nevertheless tends to incur significant silicon area and energy consumption overhead. Motivated by recent significant improvement of high-density embedded dynamic random access memory (eDRAM) towards high manufacturability at low cost, we explored the potential of integrating eDRAM in read channel integrated circuits (IC) to minimize the silicon area and energy consumption cost incurred by iterative signal detection and coding. As a result of the memory-intensive nature of iterative signal detection and coding algorithms, the silicon cost can be reduced in a straightforward manner by directly replacing conventional SRAM with eDRAM. However, reducing the energy consumption may not be trivial. In this paper, we present two techniques that trade eDRAM storage capacity to reduce the energy consumption of iterative signal detection and coding datapath. We have demonstrated eDRAM's energy saving potential by designing a representative iterative read channel at the 65 nm technology node. Simulation shows that we can eliminate over 99.99% of post-processing computation for dominant error events detection, and achieve up to a 67% reduction of decoding energy consumption.

Index Terms—Embedded dynamic random access memory (DRAM), energy consumption, low-density parity check (LDPC).

I. INTRODUCTION

IT is almost evident that future magnetic recording read channels will employ iterative signal detection and coding techniques to sustain the continuous scaling of hard disk storage density. However, those advanced iterative signal detection and coding techniques will inevitably incur significant silicon area and energy consumption overhead. Motivated by recent significant improvement of high-density embedded DRAM (eDRAM) [1]–[4], this paper attempts to explore the potential of using eDRAM instead of conventional SRAM as on-chip memory in read channel integrated circuits (IC) to reduce the silicon area and energy consumption induced by those advanced iterative signal detection and coding techniques.

As reported by IBM [3], compared with conventional SRAM, eDRAM can achieve $3 \times$ higher storage density and $0.8 \times$ lower energy consumption while maintaining a sufficiently high-speed performance for most applications. Therefore, due to the memory-intensive nature of iterative signal detection and coding, we can directly use eDRAM as a drop-in replacement of SRAM to largely reduce the silicon area overhead and modestly reduce energy consumption in a very straightforward manner. This work concerns how to further improve the energy efficiency through read channel architecture design innovations when eDRAM is being used as on-chip memory. It is intuitive that the high storage density of eDRAM could make it feasible or economic to apply certain unconventional design approaches that essentially trade memory storage capacity for energy efficiency. Following this intuition, we propose two design approaches, including 1) conditional execution of dominant error event detection and 2) iterative decoder voltage overscaling. The first approach tends

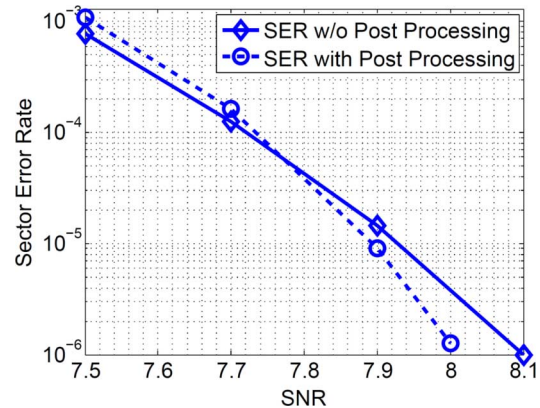


Fig. 1. SER with and without post processing.

to obviate a large percentage of explicit executions of dominant error event detection, while the second approach leverages the run-time variations of decoding iteration numbers to aggressively reduce the iterative decoder supply voltage. Both design approaches can effectively reduce the energy consumption but demand extra memory storage capacity.

To demonstrate the proposed design approaches, we use an iterative read channel as a test vehicle, which employs low-density parity-check (LDPC) code, soft output Viterbi algorithm (SOVA) signal detection, and dominant error event detection. Targeting at 1.5 Gb/s channel throughput with the 512-byte sector format, we designed the entire iterative read channel at 65 nm CMOS technology node. We show that the first design approach (i.e., conditional execution dominant error even detection) can eliminate over 99.99% of post-processing computation for detecting dominant error events, and the second approach (i.e., LDPC decoder voltage overscaling) can achieve up to 67% reduction of LDPC decoding energy consumption.

II. BASELINE ITERATIVE READ CHANNEL

The baseline iterative read channel being considered in this work uses LDPC code and SOVA signal detection. Each sector

Manuscript received March 01, 2009; revised May 15, 2009 and June 15, 2009. Current version published December 23, 2009. Corresponding author: N. Xie (e-mail: xien@rpi.edu).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TMAG.2009.2026898

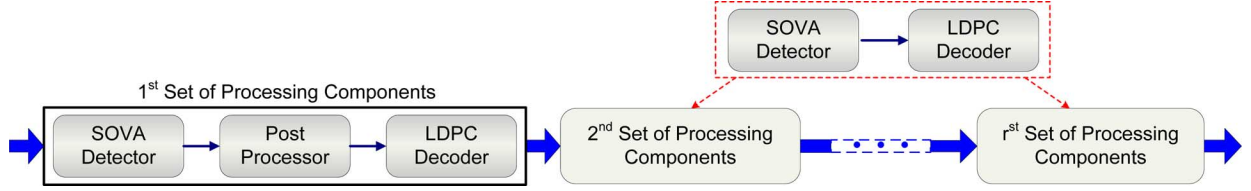


Fig. 2. Unrolled baseline magnetic recording read channel architecture.

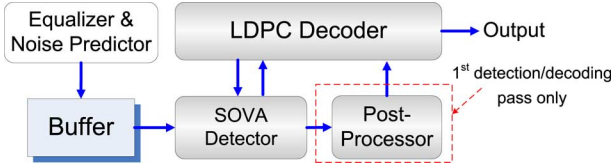


Fig. 3. Recursive baseline magnetic recording read channel architecture.

contains 512-byte user data, and the equalizer contains a 10-tap FIR filter with the target of $1 + 0.75D$ followed by a 3-tap whitening filter. A rate-8/9 regular quasi-cyclic (QC) LDPC code with the column weight of 4 is being used. To further improve the performance, a post-processor is also used to realize dominant error event detection [5]–[7]. We interleave two 64-bit single parity check codes for the purpose of dominant error event detection. In this context, the post-processor operates on the hard-decision of the SOVA detector output and, once it detects a dominant error event, it simply sets the corresponding soft-output magnitude to zero. Based on our simulations, using the post-processing in the first round of channel detection/decoding can noticeably improve the overall system performance, while it does not help if the post-processing is further used in the succeeding detection/decoding iterations. With the maximum allowable channel detection/decoding iteration number of 4, Fig. 1 shows the simulated sector error rate (SER) results with and without post processing in the first round of channel detection/decoding, respectively. It clearly indicates at least 0.1 dB gain by using post-processor to perform the dominant error event detection.

Given a target read channel throughput S_{ch} and the maximum allowable channel iteration number r , such iterative read channel may be implemented with two different options: 1) Unrolled architecture as illustrated in Fig. 2: All the components including SOVA detector, post-processor, and LDPC decoder are designed to achieve the throughput S_{ch} , and simply duplicated by r times along the datapath; 2) Because the number of channel iterations in the run time varies from one sector to the next, we can use a recursive architecture, as illustrated in Fig. 3. We implement only one set of components that must achieve a throughput, denoted as S_{comp} , which is higher than the target channel throughput, and insert a buffer between equalizer and SOVA detector to prevent data loss. This work assumes a baseline read channel with the recursive architecture because of its obvious advantage of silicon area.

A. Estimation of Buffer Size

One critical issue in this baseline recursive read channel architecture design is to determine the size of the buffer memory that is used to prevent data loss. The buffer should be just big

enough to ensure that the buffer overflow rate is lower than the target sector error rate (SER). We assume that the datapath is pipelined and its controller is designed in such a way that all the components are almost always busy (i.e., processing data). Let d denote the number of sectors that the buffer can hold, n_i denote the number of channel iterations required for each sector, and F denote the sector length. The latency of processing K sectors can be approximated as $T_K = \sum_K (n_i + 1) \cdot F / S_{comp}$, during which $T_K \cdot S_{ch} / F$ sectors arrives and K sectors leaves. Therefore, to avoid buffer overflow, we should have

$$\sum_{i=1}^K (n_i + 1) \cdot \frac{S_{ch}}{S_{comp}} - K < d. \quad (1)$$

Hence, let $P_c^{(n_i)}$ denote the probability that n_i channel iterations are required for processing each sector, the upper bound for buffer overflow probability P_f can be estimated as

$$\sum_{i=1}^K (n_i + 1) \cdot \frac{S_{ch}}{S_{comp}} - K > d \left(\prod_{i=1}^K P_c^{(n_i)} \right) \quad (2)$$

which must be lower than the target SER. Due to the lack of analytical methods, we can carry out extensive computer simulations to estimate the values of $P_c^{(n_i)}$. Because of the very low target SER (e.g., 10^{-10} and below) in practice, we may have to use conservative trajectory extrapolations to approximately estimate $P_c^{(n_i)}$ and overflow probability upper bound P_f . Moreover, it is clear that the overflow probability upper bound also depends on the value of K . As K increments from 1 to infinity, the overflow probability upper bound will first increase and then decrease and eventually approach to zero. In this work, we rely on extensive numerical calculations to search for the K that leads to the maximal overflow probability upper bound.

B. Baseline Read Channel ASIC Design

We assume the target channel throughput S_{ch} is 1.5 Gb/s, the component throughput S_{comp} is 2 Gb/s, and the maximum allowable number of channel iterations is 4. We estimate the buffer size as follows. Under two different SERs, we carry out simulations and obtain the channel iteration number statistics as listed in Table I, based on which we conservatively estimate the buffer overflow probability P_f under different buffer size d as shown in Fig. 4. Assuming a target SER of 10^{-10} , we set $d = 5$ in this baseline read channel.

With the target 2 Gb/s component throughput, we designed the SOVA detector, post-processor, and LDPC decoder using Synopsys tools and TSMC 65 nm CMOS standard cell and SRAM libraries, where the LDPC decoder can achieve 2 Gb/s in case of carrying out 24 decoding iterations. The SOVA detector uses the modified register-exchange design approach [8], and

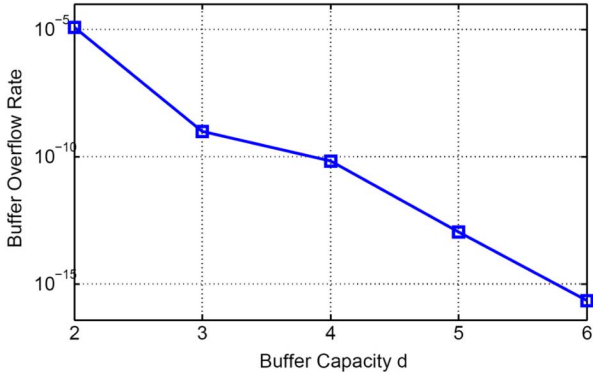


Fig. 4. Estimated sector buffer overflow rate.

TABLE I

STATISTICS OF THE CHANNEL ITERATION NUMBER UNDER TWO DIFFERENT SERS

Iteration Number n_i	SER= 1.4×10^{-4}	SER= 9.0×10^{-6}
	Occurrence \Rightarrow est. $P_c^{n_i}$	Occurrence \Rightarrow est. $P_c^{n_i}$
0	71363 \Rightarrow 0.999076	999874 \Rightarrow 0.999884
1	53 \Rightarrow 7.42e-4	108 \Rightarrow 1.08e-4
2	10 \Rightarrow 1.40e-4	7 \Rightarrow 7.00e-6
3	3 \Rightarrow 4.20e-5	1 \Rightarrow 1.00e-6

TABLE II

DATAPATH ASIC DESIGN SYNTHESIS RESULTS

	Silicon Area (mm ²)		Throughput
	Logic	SRAM	
SOVA Detector	0.10	0.06	2Gbps
Buffer	-	1.07	
Post Processor	0.10	0.11	
LDPC Decoder	1.45	2.39	
Total	1.65	3.63	

the LDPC decoder uses sum-product algorithm and its architecture follows the one presented in [9]. Readers are referred to [6] for the description of the computations involved in dominant error even detection, and sufficient computation parallelism is used to meet the 2 Gb/s throughput. In terms of finite wordlength configuration, the output of the equalizer uses 6 bits, the path metric and soft output of the SOVA detector use 9 bits and 6 bits, respectively, the FIR coefficients and dominant error event weight metric in post processor use 6 bits and 10 bits, and the internal LDPC decoding messages use 6 bits. Table II summarizes the synthesis results including the area of logic circuit and SRAM.

III. DESIGN EXPLORATION USING EMBEDDED DRAM

This section discusses the potential of exploiting the higher storage density enabled by eDRAM to improve the above baseline read channel silicon area and energy efficiency. The above design results of the baseline read channel show that the on-chip SRAM occupies more than 68% of the total silicon area, which clearly suggests a great area reduction potential if we simply replace the on-chip SRAM with eDRAM. This will lead to a 45% saving of the total silicon area assuming eDRAM achieves $3 \times$ higher density than its SRAM counterpart [3]. Besides such straightforward drop-in replacement to reduce silicon area, this section presents two approaches that further leverage eDRAM to reduce read channel energy consumption. It should be pointed out that the process of eDRAM may introduce up to 10% extra

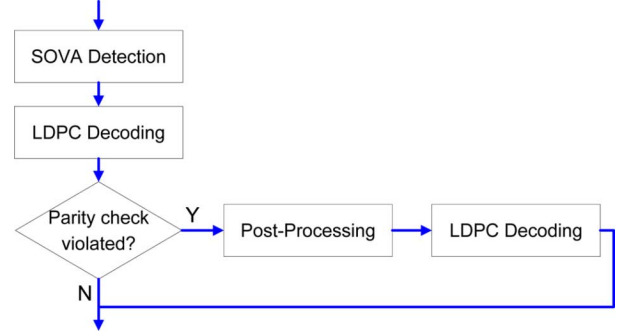


Fig. 5. Modified data processing flow for conditional execution of post-processing in the first round of read channel processing.

fabrication cost, leading to a subtle tradeoff between potential performance gain and cost penalty. Such a tradeoff should be carefully considered and evaluated in practice.

A. Conditional Execution of Post-Processing

As illustrated in Fig. 3, like in current design practice, the post-processor in the first detection/decoding pass carries out dominant error even detection for all the sectors. In this work, we propose to modify the data processing flow as illustrated in Fig. 5: Instead of blindly performing post-processing on each sector, we first carry out LDPC decoding immediately after signal detection, and the post-processing is invoked only if the decoding fails. This is motivated by the observation that, under the target very low sector error rate, most sectors can be successfully decoded during the first pass even without using post-processing, which suggests that most post-processing during the first pass is unnecessary and simply wastes energy.

Clearly, to support such conditional execution of post-processing, we must add a buffer that can hold two data frames in case LDPC decoding fails and we need to invoke post-processing. One of the data frames is 6-bit channel output data and the other one is 1-bit detector hard decision. At 65 nm technology node, such a buffer will occupy 0.31 mm² if SRAM is being use, which can be reduced to 0.1 mm² when eDRAM is being used. Hence, the use of eDRAM can better justify and support this proposed conditional execution of post-processing. To demonstrate its energy saving potential, we carried out the following simulations and analysis. It is clear that, when we use the above data processing flow, the overall decoding iteration number of the LDPC decoder may increase, i.e., the LDPC decoder may consume more energy. Let P_L^u and P_L^c denote the average power consumption of the LDPC decoder with unconditional and conditional post-processing, respectively. Let P_P and P_e represent the power consumption of the post processor and eDRAM respectively. If the post processor is invoked with the probability of α , the average power saving can be estimated as follows:

$$(P_L^u + P_P) - (\alpha \cdot P_P + (1 - \alpha) \cdot P_L^c + P_e). \quad (3)$$

Based on the simulation results as shown in Fig. 1, we assume the system will operate under the SNR of 8.6 dB in order to reach sufficiently low sector error rate. Following the results in [3] (i.e., energy consumption of eDRAM tends to be $0.8 \times$ lower than its SRAM counterpart) and using Synopsis tools (TSMC 65 nm CMOS standard cell with 1.2 V power supply), we estimate the power consumption for every component as in Table III.

TABLE III
POWER CONSUMPTION RESULTS

	Power Consumption (mW)	
	P_L^u	P_L^c
LDPC Decoder @ 8.6dB	64	68
SOVA Detector	81	
Embedded DRAM	12	
Post Processor @ 8.6dB	51	

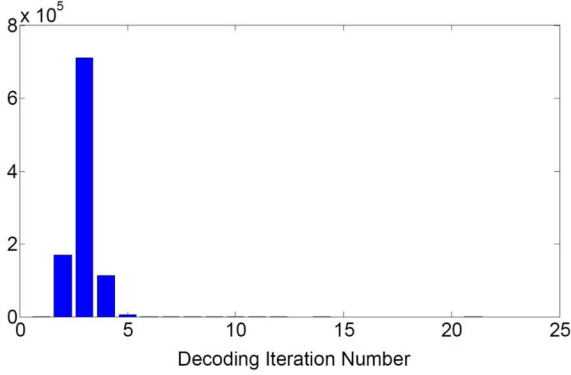


Fig. 6. Histogram of LDPC decoding iteration numbers.

Meanwhile, targeting at an SER below 10^{-10} , we carry out simulation to estimate α . With the estimated $\alpha = 2.1 \times 10^{-5}$, based on (3) and the results listed in Table III, we have that 35 mW can be saved at the expense of extra 0.1 mm² silicon area.

B. LDPC Decoder Voltage Scaling

We further develop a method that leverages the large storage capacity provided by eDRAM to enable the well known voltage scaling technique to reduce LDPC decoder energy consumption. Let N_{\max} denote the maximum allowable number of LDPC decoding iterations. Due to the on-the-fly decoding convergence check inherent in LDPC decoding, the run-time number of decoding iterations may vary from one sector to the next and the average iteration number can be much less than N_{\max} . For example, we simulated 10^6 sectors at 8.6 dB under the above presented read channel configuration and obtained the LDPC decoding iteration number histogram as shown in Fig. 6.

Let L denote the target read channel sector processing rate, and $V_{dd-\text{crit}}$ denote the supply voltage under which the LDPC decoder carry out N_{\max} iterations within $1/L$. When operating under the supply voltage $V_{dd-\text{crit}}$, due to the significant run-time decoding iteration number variation as shown in the above, the LDPC decoder may simply be idle most time during the run time, leading to a potential for applying voltage scaling to reduce energy consumption. Ideally, we may want to dynamically scale the supply voltage so that it is *just enough* for the LDPC decoder to carry out the exact number of iterations for decoding each sector. However, since the exact number of decoding iterations cannot be known until the decoding is finished, it is impossible to realize such ideal voltage scaling *a priori*. Furthermore, such fine-grain dynamic voltage scaling tends to incur non-negligible silicon and energy overhead.

Leveraging the large storage capacity provided by eDRAM, we propose to insert a certain amount of buffer memories between the detector and decoder, as illustrated in Fig. 7, to enable a fixed voltage scaling on LDPC decoder. Under a scaled supply voltage, the LDPC decoder may not always be able to

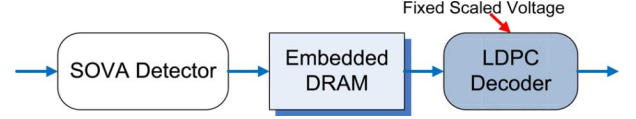


Fig. 7. Embedded DRAM buffer stacking to enable LDPC decoder voltage scaling.

finish the decoding of present sector within $1/L$, which is referred to as *decoding overflow*. The buffer memories are used to prevent the sector loss in presence of LDPC decoding overflow. Notice that, in order to ensure iterative detection and decoding, this LDPC decoder buffer should store both the input and output of the SOVA detector. As we reduce the voltage scaling factor, the LDPC decoder energy consumption will accordingly reduce, but the probability of decoding overflow will increase, which will demand a larger amount of buffer memories to prevent buffer overflow. This work studies this design tradeoff described below.

Given voltage scaling factor $K_v < 1$, the buffer memories should be sufficiently large so that the buffer overflow probability is (much) less than the target sector error rate. Let m denote the number of sectors that can be stored in the buffer memories and N_r represent the maximum number of decoding iterations that the LDPC decoder can carry out within $1/L$. We assume that the decoding of all the sectors is statistically independent and let $P^{(n)}$ represent the probability that n iterations are required in one LDPC decoding. Therefore, during the time period of K/L , the upper bound for the buffer overflow probability P_r can be estimated as

$$P_r \leq \sum_{i=1}^K \sum_{n_i > (m+K) \cdot N_r} \left(\prod_{i=1}^K P^{(n_i)} \right). \quad (4)$$

In spite of the above simple formulation, there are no existing accurate analytical methods that can estimate the values of $P^{(n)}$ for LDPC decoding. Hence, we have to empirically estimate $P^{(n)}$ through simulations. Given target buffer overflow probability P_r and m , we can accordingly determine the minimal allowable value of N_r . On the first order of approximation, we have that the circuit delay is proportional to $V_{dd}/(V_{dd} - V_t)^\alpha$, where $\alpha \in [1, 2]$ is the velocity saturation index. Therefore, we can estimate the allowable voltage scaling factor K_v by solving the following equation:

$$\frac{N_r}{N_{\max}} = \frac{(K_v \cdot V_{dd-\text{crit}} - V_t)^\alpha}{K_v \cdot (V_{dd-\text{crit}} - V_t)^\alpha}. \quad (5)$$

After we obtain the allowable voltage scaling factor K_v , the LDPC decoder energy saving percentage can be approximated as $(1 - K_v^2 - P_e^m/P_L^c)$, where P_e^m here is the power consumption of the eDRAM that can hold m sectors. To demonstrate the LDPC decoding energy saving potential, we carried out a case study as follows. First, based on the LDPC decoding iteration number statistics simulation results shown in Fig. 6, we can estimate the buffer overflow probability according to (4), as illustrated in Fig. 8.

Because the computer simulations could not empirically reveal the values of $P^{(n)}$ for $n > 14$ within a reasonable amount of simulation time, we conservatively estimate the values of $P^{(n)}$ for $n > 14$ on the order of 10^{-6} based on the above simulations. Accordingly, we can estimate the minimal allowable value

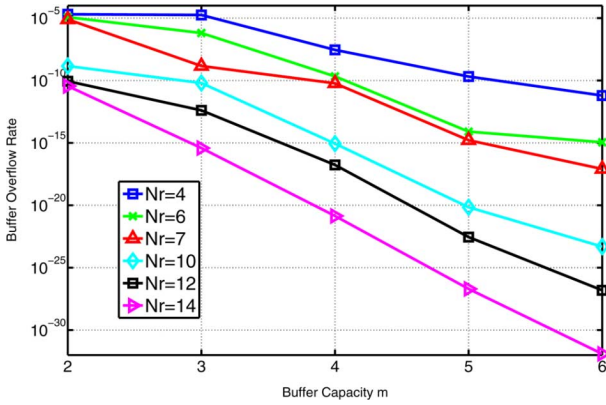


Fig. 8. Buffer overflow probability P_f vs. buffer capacity m , where m is the number of sectors that can be stored.

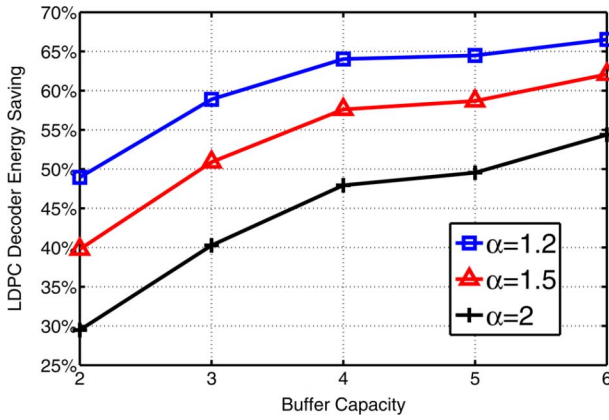


Fig. 9. Estimated LDPC decoder energy saving under different values of buffer capacity m and velocity saturation index α .

of N_r under $P_f < 10^{-10}$ and different value of m , and we have N_r equals to 14 ($m = 2$), 10 ($m = 3$), 7 ($m = 4$), 6 ($m = 5$), and 4 ($m = 6$), respectively. In our ASIC design at 65 nm node described above, the $V_{dd-crit}$ is 1.2 V and the threshold voltage is about 0.5 V. The value of α is not readily available and we consider three different values of α , i.e., 1.2, 1.5, and 2. Therefore, with $N_{max} = 24$ as in Section II.B, we can estimate the voltage scaling factor K_v (as listed in Table IV), LDPC decoder energy saving (as shown in Fig. 9) and total energy saving while taking into account of the buffer energy consumption overhead (as shown in Fig. 10) under different values of buffer capacity m and velocity saturation index α . The results clearly show a great energy saving potential for the read channel chip design, and similar potentials can be expected for many other communication systems where iterative coding and signal detection are being used. Finally, we note that the energy saving curve tends to become flat for $m > 4$, which is because the buffer energy consumption becomes more significant and offsets the energy saving gained by LDPC decoder voltage scaling.

IV. CONCLUSION

It is evident that the emerging eDRAM may shift the signal processing integrated circuit design to a new paradigm with a

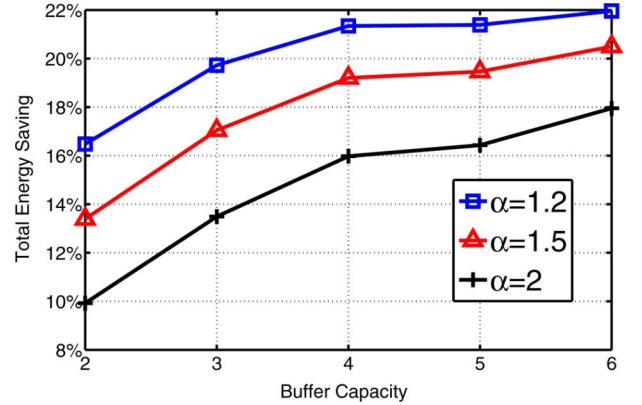


Fig. 10. Estimated total energy saving, while taking into account of the buffer energy consumption overhead, under different values of buffer capacity m and velocity saturation index α .

TABLE IV
ESTIMATED VOLTAGE SCALING FACTOR K_v

Index α	Buffer Capacity m				
	2	3	4	5	6
1.2	0.69	0.60	0.54	0.52	0.49
1.5	0.75	0.66	0.60	0.58	0.53
2	0.82	0.74	0.68	0.65	0.60

much greater design space available to explore. Particularly concerning magnetic recording read channel with advanced iterative signal processing and coding, this paper presents simple yet effective approaches that trade the memory storage capacity provided by eDRAM for energy saving. Their effectiveness has been well demonstrated using ASIC design at 65 nm CMOS technology node.

REFERENCES

- [1] Iida *et al.*, "A 322 MHz random-cycle embedded DRAM with high-accuracy sensing and tuning," *IEEE J. Solid-State Circuits*, vol. 40, pp. 2296–2304, Nov. 2005.
- [2] D. Anand *et al.*, "A 1.0 GHz multi-banked embedded DRAM in 65 nm CMOS featuring concurrent refresh and hierarchical BIST," in *Proc. IEEE Custom Integrated Circuits Conf.*, Sept. 2007, pp. 795–798.
- [3] J. Barth *et al.*, "A 500 MHz random cycle, 1.5 ns latency, SOI embedded DRAM macro featuring a three-transistor micro sense amplifier," *IEEE J. Solid-State Circuits*, vol. 43, pp. 86–95, Jan. 2008.
- [4] S. Romanovsky *et al.*, "A 500 MHz random-access embedded 1 Mb DRAM macro in bulk CMOS," in *Dig. Tech. Papers. IEEE Int. Solid-State Circuits*, Feb. 2008, p. 270.
- [5] J. Caroselli *et al.*, "Improved detection for magnetic recording systems with media noise," *IEEE Trans. Magn.*, vol. 33, no. 5, pp. 2779–2781, Sep. 1997.
- [6] W. Feng, A. Vityaev, G. Burd, and N. Nazari, "On the performance of parity codes in magnetic recording systems," in *Proc. IEEE GLOBECOM*, 2000, pp. 1877–1881.
- [7] Z. A. Keirn, V. Y. Krachkovsky, E. F. Haratsch, and H. Burger, "Use of redundant bits for magnetic recording: Single-parity codes and Reed-Solomon error-correcting code," *IEEE Trans. Magn.*, vol. 40, no. 1, pp. 225–230, Jan. 2004.
- [8] O. J. Joeressen and H. Meyr, "A 40-Mb/s soft-output Viterbi decoder," *IEEE J. Solid-State Circuits*, vol. 30, pp. 812–818, Jul. 1995.
- [9] H. Zhong, T. Zhang, and E. F. Haratsch, "Quasi-cyclic LDPC codes for the magnetic recording channel: Code design and VLSI implementation," *IEEE Trans. Magn.*, vol. 43, no. 3, pp. 1118–1123, Mar. 2007.