Novel Energy-Efficient and Latency-Improved PVT Tolerant Read Scheme for SRAM Design in Video Processing and Machine Learning Applications

Soumitra Pal*, Jiyue Yang, Stephen Bauer, Puneet Gupta, Sudhakar Pamarti

Department of Electrical and Computer Engineering
University of California, Los Angeles, CA, USA
*Corresponding author (e-mail: spal@connect.ust.hk)

***Abstract* – SRAMs consume a significant area and, thereby, a substantial portion of the total energy in modern memory-hungry processors. We propose a novel scheme for reducing the read energy consumption and latency for video or image processing and machine learning applications in which the data stored in neighboring SRAM rows is mostly similar. We read two rows at a time using a new PVT tolerant sense amplifier that decodes 3-levels – "00", "11", or "01"/"10", with the last case triggering a single row re-read. This reduces the overall number of bitline charge/discharge and, hence, overall read energy consumption and latency. A 128×256 SRAM block is simulated in 14-nm FinFET technology, and it is observed that the proposed scheme consumes up to a 49.4% lower average energy and shows a 50% shorter latency than a conventional read when the data in both rows are identical.**

Index Terms – SRAM, read energy, latency, switching activity factor, reference column, PVT tracking.

I. INTRODUCTION

Static random access memory (SRAM) remains the preferred choice for cache memory owing to its low read/write energy and latency. However, the rapidly increasing demands for the amount of SRAM in big data and machine learning applications, especially in the face of scaling challenges, prompts innovations to further reduce SRAM energy consumption and latency. In this article, we propose a read scheme based on a novel PVT tolerant 3-level read circuit to reduce energy consumption and latency where we exploit the fact that most of the neighboring SRAM cells store similar data. The proposed read scheme can operate at the same frequency as that of a conventional SRAM array.

The rest of the paper is organized as follows. The background of this work is presented in Section II. Section III describes the proposed dual-read scheme. The simulation results and comparison with the conventional read are presented in Section IV. Finally, Section V concludes the paper.

II. BACKGROUND

As SRAMs and processors are fabricated on the same die, and the embedded SRAMs consume a sizable portion of the total chip area, SRAMs consume a significant amount (~40%) of total energy consumption [1]. The energy consumption during read operations of SRAMs contributes to a significant portion of the total energy consumption [2]. Most of the dynamic energy dissipated during read operations is consumed due to the switching of the highly capacitive bitline.

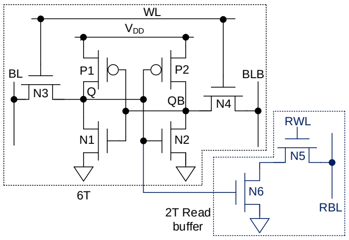


Fig. 1. Conventional 8T SRAM cell.

The energy, denoted as E, is given as follows:

$$E = \alpha_{0\to 1} \times C_{\mathrm{BL}} \times V_{\mathrm{DD}} \times (\Delta V) \quad (1)$$

where $\alpha_{0\to 1}$ is the switching activity factor, C_{BL} is effective bitline capacitance, V_{DD} is the supply voltage, and ΔV is the differential voltage on bitline. Therefore, by reducing the V_{DD}, read energy can be reduced. However, V_{DD} scaling has a negative impact on write margin, read stability, sense margin, etc., and the cell also becomes very sensitive to PVT variations [3]. In addition, low-power SRAM systems often use assist methods to lower the V_{DD} gradually [4]. However, the effectiveness of these assist methods is limited at aggressively scaled technologies, particularly at low voltages, because such technologies have quite a high variability. Considering all these, it is even preferred that an SRAM's supply voltage be higher than that of surrounding digital blocks in order to ensure reliable operation even under the worst PVT variations [2] and, hence, consume higher energy.

Other methods to reduce energy consumption are decreasing the contribution of ΔV and C_{BL}, which include employing offset-compensated sense amplifiers [5], [6] and designing hierarchical bitlines [7]. However, these techniques render large area overheads, and hence, they are of limited use.

Therefore, reducing the switching activity factor is a favorable option. For the 6T SRAM cell, the bitline activity factor is 1 because of its differential structure. On the other hand, for an 8T cell (Fig. 1), the activity factor is between 0 and 1 based on the input data (Q) because of its single-ended nature. Therefore, researchers are exploiting the single-ended nature of the 8T cell and similar data storage characteristics at neighboring SRAMs to reduce the dynamic energy by reducing the switching activity factor for specific applications [1], [8], [9], [10]. For example, the authors in [8] have 6T cells for LSBs but 8T cells for MSBs for V_{DD} scaling without increasing the error rate. The SRAM in [9] uses the high similarity in neighboring cell data to forecast the read output, which lowers the read energy consumption. [1] and [10] uses the single-ended nature of the 8T cell and applies data inverting and column-based data encoding, respectively, for converting the majority of the data in the entire array to either ‘1’ or ‘0’ to save read energy. However, the majority bit counting and inverting in [1] and column-based data encoding in [10], and after that, decoding the data during read operations in both the schemes limit the operating frequency. Hence, they cannot be operated at the same frequency as that of conventional SRAM arrays. Therefore, they operate only at a few kHz frequencies [1], [10].

Spatial and value locality in data is well known in computer vision and machine learning applications and can be exploited to reduce energy consumption [11]. For example, background pixels in images and videos have highly correlated values [12]. Such locality is also true for neural network applications where both weights and outputs are usually small valued following a bell-shaped distribution around 0 [13], [14].

Locality has been leveraged for memory compression [15] and error correction [16], [17]. For example, [16] shows that for common general-purpose software benchmarks, as much as 99% of stored words have a leading prefix of six 0s. Similarly, [17] shows that the Hamming distance between neighboring words is usually small. In this work, the strong correlation between nearby bits is used to reduce the number of bitline charge/discharge during read operations, and thereby reducing read energy and latency.

III. Proposed Dual Read Scheme

A. Basic Understanding of the Proposed Scheme

We propose a scheme that attempts to read two cells in a column at a time, i.e., reading 3-levels (“00”, “11”, and “01”/ “10”), rather than only 2-levels (‘0’ and ‘1’) in a conventional read. The proposed scheme uses the same 8T cell as shown in Fig. 1. The basics of the proposed scheme can be understood from Fig. 2. During a read operation, two consecutive RWLs are activated at the same time and RBLs are precharged to V_{DD}. Therefore, if at least one cell in a column stores logic ‘1’, the RBL of that column is discharged and remains discharged until precharged again in the next cycle, whereas if both the cells store logic ‘0’, the RBL remains precharged and doesn’t consume any energy (ideally) while precharging in the next cycle for read operation.

As can be seen from Fig. 2, we have used two inverters for sensing, and their outputs are O1 and O2. Both the outputs will be high if both the cells which are selected for reading store logic ‘1’ (i.e., “11”), and both the outputs will be low if both the cells store ‘0’ (i.e., “00”). Only O1 will be high if only one cell stores ‘1’ (i.e., for “01” or “10”, and how this distinction is made is explained in Section III-B). Therefore, it can be clearly understood that in the case of both the cells storing the same value (“00” and “11”) with charging/ discharging the RBL only once, we can read the data of two cells. However, in the case of only one cell storing ‘1’ (i.e., “01” or “10”), we have to read that column once again by selecting only one row.

For a basic understanding, let’s say the data has a 10-bit width (see Table I, which is in line with [10]). Let’s say we want to read R1 and R2 rows. As both the rows in C1-C7 columns store the same data (“11” or “00”), the corresponding sense amp (both inverters) output(s) either ‘1’ or ‘0’ based on the input data, i.e., both the inverters corresponding to column C5 will output ‘0’ (indicating both the cells are storing ‘0’), whereas both the inverters in other columns will output ‘1’ (indicating both the cells in the corresponding columns are storing ‘1’).

Table I. Considered Data Sample.

	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10
R1	1	1	1	1	0	1	1	0	0	1
R2	1	1	1	1	0	1	1	1	1	0
R3	1	1	1	1	0	1	1	0	0	0

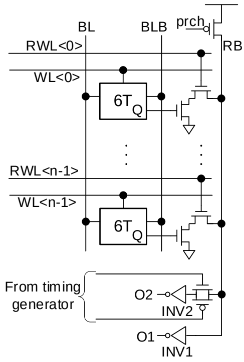


Fig. 2. Dual-row read concept: 2 consecutive RWLs are activated at the same time, and a 2b output is produced.

However, as C8-C10 columns store different data (“01” or “10”) in R1 and R2 rows, the O1 and O2 of the corresponding column will be ‘1’ and ‘0’, respectively. Therefore, it is not clear which cell in the corresponding column is storing ‘1’. Hence, these three columns have to be read again. For that, the RBLs of only these three columns are precharged again by turning ON the corresponding precharge transistors, whereas the same for all other columns are kept OFF (see Fig. 2 and Fig. 3). Now, let’s say we select to read R1. Since the SRAM cell at C10 (in Table I) stores ‘1’, its O1 will be ‘1’, whereas the O1 of C8 and C9 will be ‘0’ (meaning the cells in R2 store ‘1’).

If a normal read operation were performed, then while reading the rows R1 and R2, RBL would charge/discharge 15 times. On the other hand, in the proposed read scheme, the RBL is charged/discharged only 10 times. Furthermore, since we can select each column independently and C1-C7 store the same data in both rows, with reading only once, we can read both rows, thereby reducing 50% latency for these columns. However, since C8-C10 store different data, the latency is the same as that of the conventional scheme for these columns.

Therefore, we can exploit the characteristics of stored data in video or image processing and machine learning applications to save energy due to the charging/discharging of RBL and latency during read operations.

B. Working Mechanism of the Proposed Dual Read Scheme

The working mechanism of the proposed dual read scheme is based on the slope of RBL discharging (Fig. 4). As can be seen from the figure, based on the number of cells storing ‘1’, the slope of RBL discharging is different. INV1 in Fig. 2 should always give a high output if RBL is discharged by at least one cell (i.e., in the case of “01”, “10”, or “11”) as it is directly

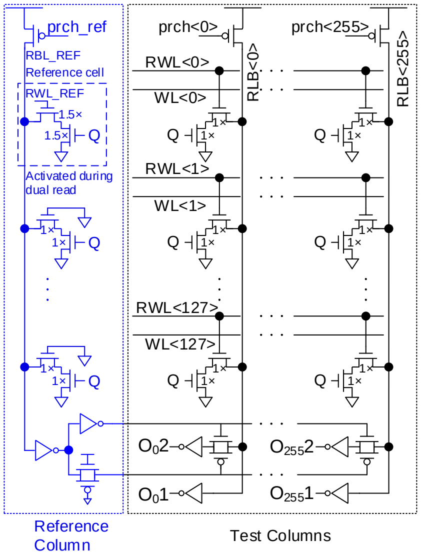


Fig. 3. PVT tracking sampling time generator.

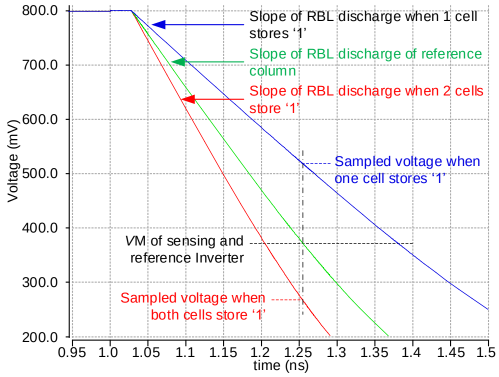


Fig. 4. RBL discharge slope and the sampled input voltage of INV2 in Fig. 2.

connected to RBL. However, INV2 should output high only when both the cells in a column store '1' (i.e., in the case of "11"). Hence, instead of directly applying the RBL, a sampled voltage of RBL is applied to the input of INV2 (Fig. 2).

The sampling time is generated from a reference column (Fig. 3). The reference column is similar to that of a test column. Hence, the reference column tracks the PVT variations in the test columns. Even though the reference column tracks the process variations, there are still local mismatches between transistors. Hence, there is a possibility of misread, i.e., the stored data '1'/'0' can be misread as '0'/'1'.

To understand misread '0', let's consider the "11" read case. As both the cells in a column are storing '1', the RBL discharges with the slope as shown in "red" in Fig. 4. Therefore, O1 in Fig. 2 will output '1', and O2 should also output '1' to read "11" correctly. However, if we make the reference column the same as the test column by activating two 2T read ports in the reference column, the reference column also discharges with the same slope as that of the test column, shown in "red" in Fig. 4. Now, if, due to transistor mismatch between the reference column and test column, the RBL discharge rate of the test column becomes slower than that of the reference column, the sampled voltage at the input of INV2 becomes higher than the switching threshold (V_M) of INV2 (see Fig. 3 and Fig. 4), and hence, the INV2 misreads the stored data as '0'.

TABLE II: READ ENERGY CONSUMPTION @ V_{DD} = 0.8V.

	Conventional (Single-read)		Proposed (Dual-read)			
Data	0	1	00	01	10	11
Energy/Column (fJ)	0.18	78.6	0.7	79.2	157.9	79.5
Avg. Energy/Access (fJ) (equal data distribution)	39.3		39.6			

The "11" can be read correctly if we slow down the discharge of the reference column by activating only one 2T read port. In that case, the slope of discharge of the reference column will be as shown in "blue" in Fig. 4. Hence, the sampled voltage for the "11" case will always be lower than the V_M of INV2, and it will output correctly. However, it can cause a misread in the case of "01" or "10". To read "01" or "10" correctly, only INV1 should output '1', whereas INV2 should output '0'. However, if, due to transistor mismatch between the reference column and test column, the discharge rate of the test column becomes faster than that of the reference column, the sampled voltage becomes lower than the V_M of the INV2, and the INV2 also outputs '1'.

Therefore, even though the reference column tracks the PVT, to consider the local mismatches between transistors and to avoid misread '0' or '1', we place the slope of the RBL discharge of the reference column ("green" slope in Fig. 4) in between the slopes of the RBL discharged by only one cell storing '1' ("blue" slope in Fig. 4) and both cells storing '1' ("red" slope in Fig. 4). Hence, even if a test column discharges slower (faster) than its nominal rate due to transistor mismatch in the case of "11" ("01" or "10"), its slope ("red" ("blue") slope in Fig. 4) still remains below (above) the reference column ("green" slope in Fig. 4), and the sampled voltage at the input of INV2 remains lower (higher) than the V_M of INV2. This helps distinguish whether an RBL in a test column is discharged by one or two cells, even in the presence of PVT variations and mismatches; based on that, either one or both the inverters produce high output. To place the reference column's slope like that, the transistor size of the 2T read buffer in the reference cell, which is active during a dual read operation, is set to 1.5×, whereas for all other cells that are kept OFF, it is 1× (Fig. 3).

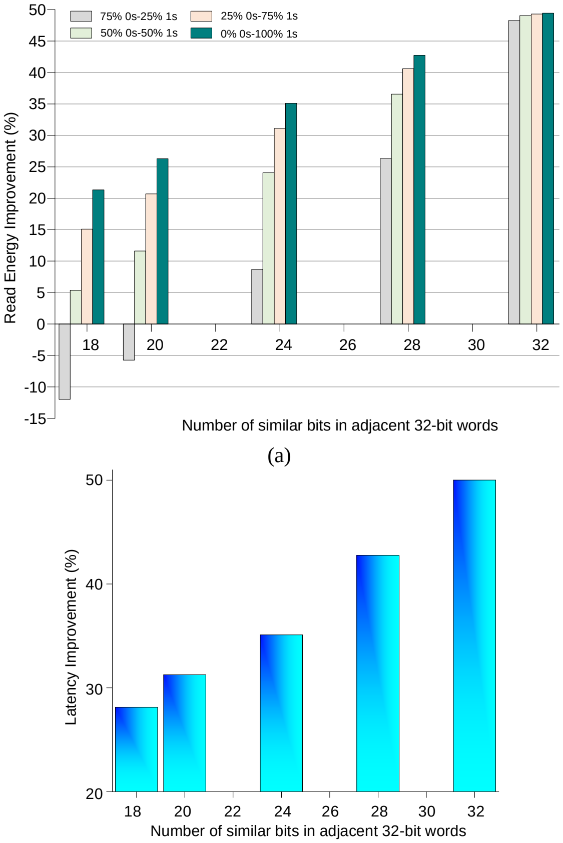
IV. SIMULATION RESULTS

We performed post-layout simulations, including the parasitics, of an SRAM array of 128×256 for the proposed dual-read scheme in 14-nm FinFET technology at 0.8 V. The operating frequency was set to 1 GHz. For a fair comparison, the conventional single-read scheme was also simulated in the same environment. The read energy consumption of both conventional and proposed schemes for different stored data is reported in Table II. On average, a conventional read operation consumes 39.3 fJ energy per read operation when we consider '0' and '1' are equally distributed.

On the other hand, if we assume that all the four cases for the proposed scheme (i.e., "00", "01", "10", and "11") are equally distributed, then it consumes 39.6 fJ per read access. This slight penalty (0.9%) is observed due to having the additional reference column for tracking the PVT variation and generating the time for sampling. The additional reference column also

TABLE III. FIRST LAYER WEIGHTS OF AN INT8 VERSION OF ALEXNET.

Bit position	0	1	2	3	4	5	6	7
Similarity (%)	71	100	100	95	84	68	55	50
% of 1s	41	0	0	2	9	21	36	51



(a)

(b)

Fig. 5. (a) Read energy and (b) read latency improvement in the proposed scheme w.r.t. similarity between data in adjacent 32b rows.

causes a 0.4% area penalty at the chip level.

Furthermore, we estimated the read energy consumption of a 32-bit word for different similarities and with different ‘0’ and ‘1’ combinations. The improvement of the proposed scheme is presented in Fig. 5 (a). As can be seen, the proposed scheme shows up to 49.4% improvement over the conventional read. Furthermore, since the proposed scheme reads two rows at a time, it also has an improvement in latency (see Fig. 5 (b)).

For a realistic example, we considered the similarity statistics of the first layer weights of an INT8 version of AlexNet (see Table III): about 18% energy saving is achievable in reading this layer with our proposed technique. Other layers also have similar weight distributions. Another realistic example we considered is the ResNet, where the probability of “00” and “11” are 21.6% and 29.4%, and the rest, 49%, are divided equally between “01” and “10”. In this example, the proposed scheme shows a 3.7% improvement in energy and a 25.5% improvement in latency.

V. CONCLUSION

In this paper, we proposed a novel dual-read scheme for reducing energy consumption and latency during read operations in video or image processing and machine learning applications. Basically, we utilized the high similarity of the data stored in SRAMs to reduce the switching activity factor for improving the read energy consumption and latency in these applications. The proposed novel read scheme can detect 3-levels rather than conventional 2-levels. The proposed scheme uses a reference column that tracks the PVT variations. As the proposed dual-read scheme is energy efficient, shows improvement in latency, and is also PVT tolerant, it can be a good option for designing SRAM in aggressively scaled technologies for video or image processing and machine learning applications.

REFERENCES

[1] H. Fujiwara *et al.*, “Novel video memory reduces 45% of bitline power using majority logic and data-bit reordering,” *IEEE Trans VLSI Syst*, vol. 16, no. 6, pp. 620–627, Jun. 2008.

[2] C. Duan, *et al.*, “Energy-Efficient Reconfigurable SRAM: Reducing Read Power Through Data Statistics,” *IEEE J Solid-State Circuits*, vol. 52, no. 10, pp. 2703–2711, Oct. 2017.

[3] M. S. M. Siddiqui, *et al.*, “A 16-kb 9T Ultralow-Voltage SRAM with Column-Based Split Cell-VSS, Data-Aware Write-Assist, and Enhanced Read Sensing Margin in 28-nm FDSOI,” *IEEE Trans VLSI Syst*, vol. 29, no. 10, pp. 1707–1719, Oct. 2021.

[4] B. Zimmer *et al.*, “SRAM assist techniques for operation in a wide voltage range in 28-nm CMOS,” *IEEE Trans. Circuits Syst. II: Exp. Briefs*, vol. 59, no. 12, pp. 853–857, 2012.

[5] B. Giridhar, *et al.*, “A reconfigurable sense amplifier with auto-zero calibration and pre-amplification in 28nm CMOS,” *IEEE Int Solid State Circuits Conf*, pp. 242–243, 2014.

[6] Y. Sinangil and A. P. Chandrakasan, “A 128 Kbit SRAM with an embedded energy monitoring circuit and sense-amplifier offset compensation using body biasing,” *IEEE J Solid-State Circuits*, vol. 49, no. 11, pp. 2730–2739, Nov. 2014.

[7] B. Do Yang and L. S. Kim, “A low-power SRAM using hierarchical bit line and local sense amplifiers,” *IEEE J Solid-State Circuits*, vol. 40, no. 6, pp. 1366–1376, Jun. 2005.

[8] I. J. Chang, *et al.*, “A Priority-based 6T/8T hybrid SRAM architecture for aggressive voltage scaling in video applications,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 21, no. 2, pp. 101–112, Feb. 2011.

[9] M. E. Sinangil and A. P. Chandrakasan, “Application-specific sram design using output prediction to reduce bit-line switching activity and statistically gated sense amplifiers for up to 1.9×lower energy/access,” *IEEE J Solid-State Circuits*, vol. 49, no. 1, pp. 107–117, Jan. 2014.

[10] A. T. Do, *et al.*, “Energy-Efficient Data-Aware SRAM Design Utilizing Column-Based Data Encoding,” *IEEE Trans. Circuits Syst. II: Exp. Briefs*, vol. 67, no. 10, pp. 2154–2158, Oct. 2020.

[11] H. Xu *et al.*, “Reducing SRAM Reading Power with Column Data Segment and Weights Correlation Enhancement for CNN Processing,” *IEEE Trans. Comput-Aided Des. Integr. Circuits Syst*, vol. 40, no. 11, pp. 2237–2250, Nov. 2021.

[12] G. Gallego *et al.*, “Event-Based Vision: A Survey,” *IEEE Trans Pattern Anal Mach Intell*, vol. 44, no. 1, pp. 154–180, Jan. 2022.

[13] M. Kurtz *et al.*, “Inducing and Exploiting Activation Sparsity for Fast Inference on Deep Neural Networks,” *Proc.-37th Int. Conf. Machine Learning. PMLR*, pp. 5533–5543, Nov. 21, 2020.

[14] M. A. Raihan and T. M. Aamodt, “Sparse Weight Activation Training,” *Int. Conf. Learning Representations (ICLR)* . 2020.

[15] A. Alameldeen and D. Wood, “Frequent Pattern Compression: A Significance-Based Compression Scheme for L2 Caches,” 2004.

[16] I. Alam, C. Schoeny, L. Dolecek, and P. Gupta, “Parity++: Lightweight Error Correction for Last Level Caches,” *Proc. 48th Annu. IEEE/IFIP Int. Conf. Dependable Syst. Netw. Workshops*, pp. 114–120, Jul. 2018.

[17] M. Gottscho, I. Alam, C. Schoeny, L. Dolecek, and P. Gupta, “Low-Cost Memory Fault Tolerance for IoT Devices,” *ACM Trans Embed Comput Syst*, vol. 16, no. 5s, Sep. 2017.