REGULAR PAPERS

Fabrication of a magnetic-tunnel-junction-based nonvolatile logic-in-memory LSI with content-aware write error masking scheme achieving 92% storage capacity and 79% power reduction

To cite this article: Masanori Natsui et al 2017 Jpn. J. Appl. Phys. 56 04CN01

View the article online for updates and enhancements.

Related content

- Fine-Grained Power-Gating Scheme of a Metal–Oxide–Semiconductor and Magnetic-Tunnel-Junction-Hybrid Bit-Serial Ternary Content-Addressable Memory
  Shoun Matsunaga, Masanori Natsui, Kimiyuki Hiyama et al.

  Shoun Matsunaga, Masanori Natsui, Shoji Ikeda et al.

- Design of a low-power nonvolatile flip-flop using three-terminal magnetic-tunnel-junction-based self-terminated mechanism
  Daisuke Suzuki and Takahiro Hanyu
Fabrication of a magnetic-tunnel-junction-based nonvolatile logic-in-memory LSI with content-aware write error masking scheme achieving 92% storage capacity and 79% power reduction

Masanori Natsui*, Akira Tamakoshi, Tetsuo Endoh, Hideo Ohno, and Takahiro Hanyu

Tohoku University, Sendai 980-8577, Japan

E-mail: natsui@nlec.tohoku.ac.jp

Received October 11, 2016; accepted November 19, 2016; published online February 16, 2017

A magnetic-tunnel-junction (MTJ)-based video coding hardware with an MTJ-write-error-rate relaxation scheme as well as a nonvolatile storage capacity reduction technique is designed and fabricated in a 90 nm MOS and 75 nm perpendicular MTJ process. The proposed MTJ-oriented dynamic error masking scheme suppresses the effect of write operation errors on the operation result of LSI, which results in the increase in an acceptable MTJ write error rate up to 7.8 times with less than 6% area overhead, while achieving 79% power reduction compared with that of the static-random-access-memory-based one. © 2017 The Japan Society of Applied Physics

1. Introduction

A magnetic-tunnel-junction (MTJ)1–9-based nonvolatile logic-in-memory (NV-LIM) architecture is one of the key technologies to solve a serious power-wall problem of VLSI processors in the emerging dark silicon era.6,9) Nonvolatility and three-dimensional stackability of MTJ devices allow us to apply a power-gating technique with fine granularity with a small area overhead. The impact of embedding nonvolatile memories into a logic circuit has been demonstrated through the design and fabrication of various NV-LIM LSIs.10–19)

In order to maximally utilize the leading-edge performance of MTJ devices and broaden the scope of their applications to practical-scale MTJ/MOS-hybrid logic LSIs, it is important to manage both hard errors due to process variation and soft errors due to stochastic behavior of MTJ devices.20–23) One possible way to manage the hard errors is to embed an error checking and correction (ECC) mechanism.24,25) However, traditional ECC techniques such as adding parity bits are not suitable for NV-LIM LSIs, which have a distributed nonvolatile memory over the logic plane. As for soft errors, nearly 100% switching probability might be guaranteed by using large pulse widths with high switching voltages for write operation, but this pessimistic approach sacrifices the performance of LSI since the energy dissipation for MTJ write operation, which is equal to write current × write voltage,26,27) and the clock frequency of LSI strongly depend on these parameters. To design NV-LIM LSI that ensures both high reliability and performance, an error checking scheme that detects a faulty condition of an internal nonvolatile memory and corrects it appropriately is essential.

In this paper, we present an MTJ/MOS-hybrid video coding hardware with an MTJ-write-error-rate relaxation scheme as well as a nonvolatile storage capacity reduction technique. The proposed processor checks whether a write operation to the MTJ device is done correctly or not, and dynamically excludes the effect of a write error with considering the logic operation. Moreover, the processor adopts a circuit architecture that can minimize the number of MTJ devices for nonvolatile storage, which leads to both reduced power consumption and compact circuit implementation while maintaining the same degree of the throughput. The efficiency of the proposed processor fabricated using a 90 nm MOS and 75 nm perpendicular MTJ process produced on a 300 mm wafer fabrication line is demonstrated by evaluating its performance including tolerance against MTJ write failures.

2. Dynamic MTJ write error masking scheme

Figure 1 shows the proposed error masking scheme. The basic concept is to determine whether a write error is negligible or non-negligible, and to mask errors if and only if they affect logic operations of LSI. From the perspective of the measured MTJ device behavior, we assume that the write errors are caused by both hard errors due to MTJ device defects (i.e., stack-at-X errors) and soft errors due to the stochastic behavior of the MTJ device. Whenever a write operation occurs, the proposed scheme checks if erroneous memory data that will cause invalid logic operations exist and masks them to be stored owing to write errors as shown in Fig. 1(a), this

![Diagram](image-url)

Fig. 1. (Color online) Basic concept of dynamic MTJ write error masking scheme. (a) Memory data is masked since error bits alter the stored value to an invalid one, (b) memory data are used since write errors occur but have no impact on the stored value.
Motion-vector prediction is a computationally extensive but crucial algorithm in video compression.28,29) There are two types of architectures, namely, motion-vector (MV)-oriented and pixel-oriented30) architectures. Figure 2 shows the differences in the data and computation allocation flows between both types of architectures. In the case of the motion-vector-oriented architecture, since all reference window data must be stored into each PE, the total number of MTJs for the reference window is \( NW^2(R_0 + R_1 + R_2 + R_3)^2 \). In the case of the pixel-oriented architecture, on the other hand, since only one pixel datum is stored into each PE, the total number of MTJs for the reference window is \( NW_R \). Note that \( N \) is bits per pixel of image data, \( W_R \) is the reference window size, and \( W \) is the search window size.

memory is considered to have non-negligible errors and the stored data are masked to prevent invalid logic operations. On the other hand, if write errors occur but have no impact on the stored value, as shown in Fig. 1(b), they are considered as negligible errors and normal logic operation is performed without data masking. This logic-operation-dependent error masking scheme can prevent the overmasking, which often occurs in conventional error masking techniques that statically mask all of the erroneous memories. It suppresses the impact of MTJ write failure on logic operation and substantially relaxes the yield requirement of the MTJ device.

3. MTJ/MOS-hybrid motion-vector prediction unit

Motion-vector prediction is a computationally extensive but crucial algorithm in video compression.28,29) There are two types of architectures, namely, motion-vector (MV)-oriented and pixel-oriented30) architectures. Figure 2 shows the differences in the data and computation allocation flows between both architectures in a 3 × 3-pixel search window and a 2 × 2-pixel reference window. Since each processing element (PE) in the pixel-oriented architecture performs an element-by-element computation of one reference window pixel, only one pixel datum is stored in each PE, in contrast to the entire-reference-window pixel data in the MV-oriented architecture. As a result, the total storage capacity can be markedly reduced, and the power dissipation due to write operation is also minimized. Since the total amount of internal memories for reference windows depends on the size of the reference window and bits per pixel of image data, the pixel-oriented architecture is preferable for MTJ-based MV prediction units.

Figure 3 shows the circuit architecture of an MTJ/MOS-hybrid MV prediction unit for an 8 × 8 search window and a 4 × 4 reference window. Unlike the MV-oriented architecture, which contains several processing elements for parallel sum-of-absolute (SAD) operations, the absolute-difference function and accumulation function are separated into different circuit blocks, which reduces the total amount of MTJ-based registers embedded in the logic circuit and the total write energy of MTJ-based nonvolatile registers per operation.

Let the coordinates of the search window be defined as (0, 0) to (7, 7) and that of the reference window as (2, 2) to (5, 5), where PE(a, b) has an 8-bit nonvolatile memory for storing a pixel value located at \((a + 2, b + 2)\) and calculates the absolute difference between the stored pixel value and the input value. The result from this calculation is transferred to one of the 25 accumulators (ACCs) via a connection block and accumulated onto a 12-bit nonvolatile memory. That is, ACC(c, d) stores an intermediate value of the SAD calculation results between the reference window and one of the candidate windows \((c, d)\) to \((c + 3, d + 3)\). Note that since all intermediate values in PEs and ACCs are maintained in a nonvolatile manner, the power supply for each unit can be controlled every operation cycle without data retention and restoration, which leads to a great reduction in leakage current.

Figure 4 shows the circuit structure of a PE with the dynamic error masking function. The basic function of the PE is to calculate the absolute difference between the input value and the stored pixel value in an 8-bit nonvolatile memory. The proposed error masking function is embedded into each PE compactly by partially sharing the circuit for the absolute difference function. The PE has two modes: the test and operation modes. If non-negligible errors are found in the check mode, the 1-bit nonvolatile flip-flop (NVFF) is set to...
1, and the PE outputs 0 in the operation mode independent of the input data until the next write operation occurs. It prevents ACCs, each of which stores a sum-of-absolute-difference calculation result between a candidate window in the search area and the reference window, from accumulating errors due to the invalid stored data. The operation of MV prediction using a dynamic error checking scheme is summarized as a flowchart in Fig. 5.

4. Evaluation results

Figure 6 shows a test chip photomicrograph of the MV prediction unit designed on the basis of using an automated design flow for the MTJ-based NV-LIM LSI and fabricated in a 90 nm MOS and 75 nm perpendicular MTJ process produced on a 300 mm wafer fabrication line. Sixteen PEs are regularly arranged at the bottom, while the other circuit blocks are arranged at the top, where their circuit hierarchies are automatically ungrouped by the design tool16,17 in terms of optimized performance. Note that since the MTJ devices are stacked over a CMOS layer through metal layers, a nonvolatile memory function can be embedded into the logic circuits with a small area overhead.

Table I summarizes the performances of MV prediction units with three different architectures. Note that each unit is designed to have as low power and small area as possible while maintaining the same duration of delay to confirm the impact of the proposed architecture. Since the write operation of input data (B in Fig. 4) to the 8-bit NV memory in PE (M in Fig. 4) determines the maximum delay time of the circuit, the addition of the proposed error masking function does not affect the delay or operating frequency of the circuit. Despite the addition of the error masking function, the proposed circuit greatly reduces the circuit area since the numbers of transistors and MTJ devices are reduced to 14.7 and 8.5% of the circuit with the MV-based architecture, respectively. The power dissipation is also reduced to 20.8% of that with the SRAM-based one. Note that the area overhead for adding the error masking function is less than 6%, which becomes negligible with increasing circuit size.

Figure 7 shows the waveforms of a PE in the fabricated MV prediction unit measured using a logic analyzer. Low-2-
bit input and output sequences of write, check, and operation modes with two different operation results are presented. We can see that both normal operation [Fig. 7(a)] and mask operation [Fig. 7(b)] are performed depending on the existence of a non-negligible write error.

Figure 8 shows the relationship between MTJ write error rate and root-mean-square error occurring in the MV prediction operation. A 15-frame video sequence at 352 × 240 pixels in the uncompressed YUV4MPEG format (sample video sequence: Foreman31) is used for this evaluation example. Five distinct trials with different error patterns are performed for each method of error handling. We can see that the proposed error checking scheme can suppress the effect of MTJ write error rate on the operation result compared with the conventional static error checking scheme, which determines the point of failure first and statically masks it until the logic operation has finished, and effectively relaxes the yield requirement for MTJ devices.

Figure 9 shows the effect of the proposed error checking function on the relaxation of an acceptable MTJ write error rate. We compare three different methods of handling error by evaluating an acceptable MTJ write error rate for satisfying less than 0.5-pixel average root-mean-square error in predicted MVs. The proposed scheme is quite simple but

Table I. Comparison of performance.

<table>
<thead>
<tr>
<th></th>
<th>SRAM-based/ MV-orienteda)</th>
<th>MTJ-based/ MV-oriented (Ref. 16)</th>
<th>MTJ-based/ pixel-oriented (This work)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Power (mW)</td>
<td>1.22 3.21</td>
<td>1.22 0.83</td>
<td>0.70 0.22</td>
</tr>
<tr>
<td>Dyn. Leak (mW)</td>
<td>4.43 2.05</td>
<td>4.68 2.05</td>
<td>0.92</td>
</tr>
<tr>
<td>Delay (ns)</td>
<td>3.68 3.68</td>
<td>3.68 3.68</td>
<td>3.68</td>
</tr>
<tr>
<td>Area (mm²)</td>
<td>1.15 1.36</td>
<td>1.15 1.36</td>
<td>0.38</td>
</tr>
<tr>
<td># Transistors</td>
<td>554 × 10³</td>
<td>474 × 10³</td>
<td>70 × 10³</td>
</tr>
<tr>
<td># MTJs</td>
<td>0</td>
<td>13 × 10³</td>
<td>1 × 10³</td>
</tr>
<tr>
<td>Nonvolatility</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>ECC</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
</tbody>
</table>

a) Circuit architecture is based on Ref. 16.
effectively mitigates the impact of failed memories regardless of the content of video sequences, and achieves a 7.8-fold improvement at maximum. This technique can apply LIM-style circuitry in general and would contribute to the design of highly reliable and low-power NV-LIM LSI.

5. Conclusions

In this paper, we proposed a write-error-rate relaxation scheme for MTJ-based NV-LIM LSI. The impact of the proposed scheme was demonstrated through the fabrication and measurement of NV-LIM video coding hardware. As a future prospect, we will consider applying the proposed error masking technique for realizing energy-efficient brain-inspired computing systems. Since brain-inspired algorithms such as neural networks have general-purpose approximate functions and are inherently error-resilient, they can be optimized to handle noisy or partially missing data appropriately. The proposed scheme will help to realize a good balance between the error resilience and computation quality, which will open the door to spintronics-based brain-inspired computing.

Acknowledgments

The authors thank Y. Takako of Focal Agency for excellent technical assistance. Part of this research was supported by the JSPS ImPACT Program, R&D for Next-Generation IT technical assistance. Part of this research was supported by the JSPS KAKENHI Grant Number 16KT0187.

31) Xiph.org test media [http://media.xiph.org/video/derf/].