Solid-State Electronics 120 (2016) 32–40

Contents lists available at ScienceDirect

Solid-State Electronics

journal homepage: www.elsevier.com/locate/sse

An efficient method for evaluating RRAM crossbar array performance

Lin Song [a,*], Jinyu Zhang [a], An Chen [b], Huaqiang Wu [a], He Qian [a], Zhiping Yu [a]

[a] Department of Microelectronics, Tsinghua University, Beijing 100084, PR China
[b] GLOBALFOUNDRIES, Sunnyvale, CA 94085, USA

ARTICLE INFO

Article history:
Received 6 September 2015
Received in revised form 22 January 2016
Accepted 8 March 2016
Available online 23 March 2016

The review of this paper was arranged by Dr. Y. Kuk

Keywords:
RRAM
Crossbar array
Reduction of RRAM array

ABSTRACT

An efficient method is proposed in this paper to mitigate computational burden in resistive random access memory (RRAM) array simulation. In the worst case scenario, a 4 Mb RRAM array with line resistance is greatly reduced using this method. For 1S1R-RRAM array structures, static and statistical parameters in both reading and writing processes are simulated. Error analysis is performed to prove the reliability of the algorithm when line resistance is extremely small compared with the junction resistance. Results show that high precision is maintained even if the size of RRAM array is reduced by one thousand times, which indicates significant improvements in both computational efficiency and memory requirements.

1. Introduction

Resistance-switching random access memory (RRAM) is considered a promising candidate to replace traditional flash memory. RRAM has low programming voltage, fast speed, and high density, thereby it combines the advantage of flash memory and dynamic random access memory (DRAM) [1]. However, passive RRAM array suffers from drawbacks, such as poor uniformity, sneak path current, and interconnect parasitic resistance, which limit the size of the RRAM array [2]. To solve this problem, a selector in series with RRAM (also called 1S1R structure) is proposed [3–5]. Simulation and analysis of RRAM crossbar array can provide a valuable reference for experiments. There are already some studies related to simulation of RRAM crossbar array. In the work of Flocke et al. [6], both linear and nonlinear RRAM crossbar arrays are simulated, but the parasitic interconnect resistance is neglected. What's more, selector is not considered and the maximum size is limited to 4K. In the work of Chen [7], nonlinearity, parasitic resistance and selector are considered, and the simulation tool utilized is MATLAB based on Kirchhoff's law. However the maximum size of the RRAM is limited because of accuracy and memory limitations. In the work of Deng et al. [8], nonlinearity, parasitic resistance, and different kinds of selection devices are studied, and the simulation tool used is SPICE. However, the maximum size of RRAM is limited to 16k, which is too small. The larger the array size is, the required simulation time and memory resources increase dramatically. One of the solutions to this problem is reducing the size of array. In the work of Liang et al. [9,10], a reduced circuit model of crossbar array is analyzed but the simulated array size is limited to 64×64 without consideration of 1S1R structure. In the work of Narayanan et al. [11], the unselected cells are aggregated into one cell to reduce the size of the crossbar array, but the reduction ratio is fixed and the accuracy is not convincing because the maximum verified array size is only 32 Kb.

This study proposes an algorithm to reduce crossbar size for both linear and nonlinear crossbar arrays with various reduction ratios. While in a linear crossbar array, the RRAM junction element is a linear resistance (also called 1R structure). In a nonlinear crossbar array, the RRAM junction element is a linear resistance in series with a bipolar nonlinear selector (also called 1S1R structure) to deal with "sneak path" problem.

In this study, the crossbar array is simulated by HSPICE, and the maximum array size is set to 4 Mb. The RRAM array is reduced under worst-case scenario because its size is limited by the worst case's performance [8,12,13]. The worst-case pattern in this study is from Deng's work [8]. Using the algorithm from this study to simplify the crossbar array, we can reduce the array size to about 1/1000 of the original array, and various electric properties, such as equivalent resistance can maintain a good accuracy. Moreover, both reading and writing process is simulated and the simplified arrays show excellent agreement with original arrays. Besides, the variation on resistance is also considered. This paper is organized as follows: Section 2 describes the details of the reduction method. In Section 3, some simulation results are presented and

* Corresponding author.
E-mail addresses: l-song14@mails.tsinghua.edu.cn (L. Song), zhangjinyu@tsinghua.edu.cn (J. Zhang).

http://dx.doi.org/10.1016/j.sse.2016.03.004
0038-1101/

analyzed. Section 4 shows an error analysis of the proposed reduction method. Finally, a conclusion is provided in Section 5.

2. Reduction algorithm

To better understand the reduction algorithm, some basic concepts of the RRAM array are discussed in this section. A typical 1R-RRAM crossbar array is presented in Fig. 1. During the writing process, the junction between the word line and the bit line connected with Vdd and GND is operated. The biases on word lines and the bit lines are used to prevent errors [14]. Generally, bias modes are classified into three: 1/2 bias,1/3 bias, and floating bias. In the 1/2 bias scheme, all the unselected lines are biased to $V_{dd}/2$, so in Fig. 1, the unselected devices in reverse biased group are zero biased, whereas the unselected devices in forward biased group are $V_{dd}/2$ biased. In the 1/3 bias scheme, all the unselected lines are biased to $V_{dd}/3$ or $2V_{dd}/3$, so all the unselected devices are either $+V_{dd}/3$ biased or $-V_{dd}/3$ biased. In the floating scheme, all the unselected lines are floating. Usually, the writing margin (WM) is used to evaluate the performance of writing:

$$WM = \frac{V_{cell,selected} - \max(V_{cell,unselected})}{V_{dd}} \times 100\% \quad (1)$$

where $V_{cell,selected}$ is the voltage drop on the selected cell, and $\max(V_{cell,unselected})$ is the maximum voltage drop on unselected cells. V_{dd} is the applied voltage in the source. During the reading process, different from the traditional readout [15], a sensing resistance is used to indicate the on/off state of the selected junction element. Usually, the sensing margin (SM) is used to evaluate reading performance and is defined as

$$SM = \frac{V_{rsense,on} - V_{rsense,off}}{V_{dd}} \times 100\% \quad (2)$$

where $V_{rsense,on}$ and $V_{rsense,off}$ are the voltages on sensing resistance when the selected RRAM is in low resistance state and high resistance state, respectively. Region X in Fig. 1 indicates the building block of the resistance array. Our method is to simplify the building block to construct a new resistance array with reduced size. Fig. 2(a) shows the details of Region X indicated in Fig. 1, where r_t is the equivalent resistance between nodes E and F. Fig. 2(b) and (c) shows the reduction details of Region X. If R_{rram1} is approximately equal to R_{rram2}, based on the assumption that r_t is relatively small compared with R_{rram} and is thus negligible, then we have $V_A \approx V_B$ and $V_E \approx V_F$. Thus Fig. 2(a) can be simplified to Fig. 2(b). Using equivalent transformation, we can convert the resistance network in Fig. 2(b) into Fig. 2(c). Thus, the number of junctions connected by word line or bit line is reduced by half. The matrix of the resistance network across nodes A, B, C, and D can be derived easily. Let the matrix elements of Fig. 2(b) and (c) be equal, then r'_1, r'_2, and R'_{rram} can be obtained as follows:

$$r'_1 = \frac{R_{rram1} r}{R_{rram1} + R_{rram2} + r} \quad (3)$$

$$r'_2 = \frac{R_{rram2} r}{R_{rram1} + R_{rram2} + r} \quad (4)$$

$$R'_{rram} = \frac{R_{rram1} R_{rram2}}{R_{rram1} + R_{rram2} + r} \quad (5)$$

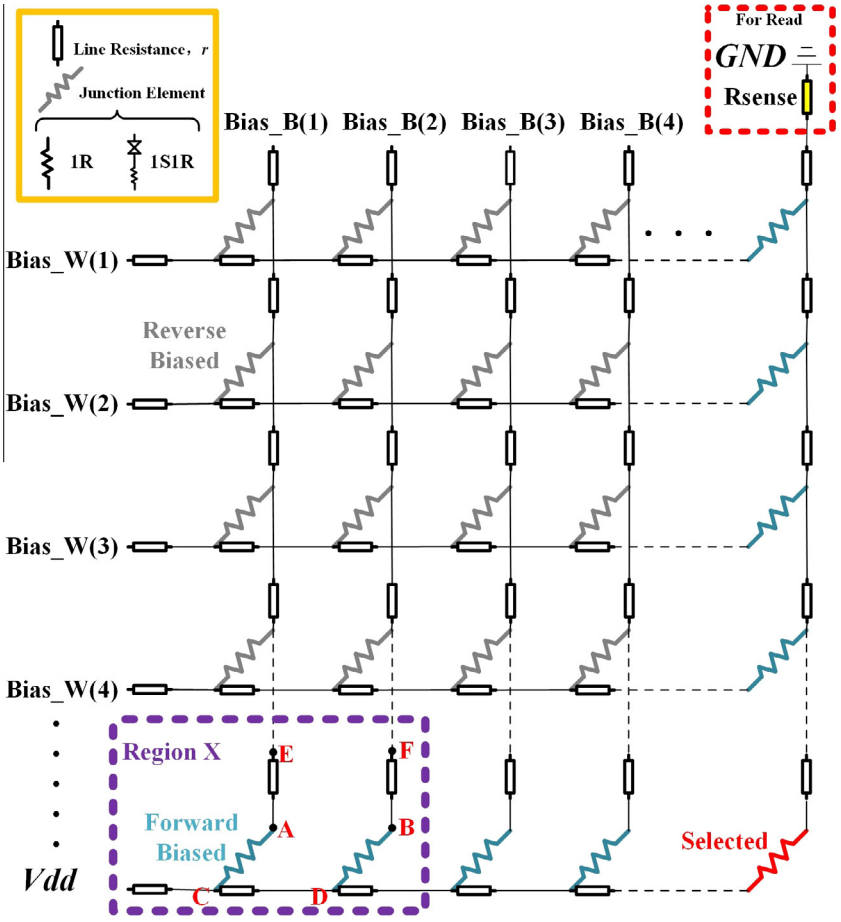

Fig. 1. RRAM crossbar array with line resistance and bias scheme. The different colors represent the different biased regions. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)