# State-Dependent Coarse-Fine Control: A Fast, Accurate, Energy-Efficient Programming Algorithm for Multiple Bits-per-Cell RRAM Array

B.Q. Le[1,2] Senior Member, IEEE, T. Wu[1], A. Levy[1], E.R. Hsieh[1,3], E. Vianello Member, IEEE, H.-S.P. Wong[1], S. Wong[1], S. Mitra[1]

[1]Dept. of Electrical Engineering, [5]Dept. of Computer Science, Stanford University, Stanford, CA, USA

[2]Dept. of Electrical Engineering, San Jose State University, San Jose, CA, USA

[3]Institute of Electronics, National Chiao Tung University, Hsinchu, Taiwan, [4]CEA, LETl, MlNA TEC, Grenoble, France

E-mail: binh@stanford.edu

**Abstract –** 3-bit-per-cell $HfO_2$-based RRAM memory has been recently demonstrated at the array level for the first time. Prior RRAM programming methods perform sub-optimally for array-level RRAM with 3 bits per cell. Here, we present the State-Dependent Coarse-Fine Control (**SDCFC**) method, a fast, accurate and energy-efficient programming algorithm for 3 bits-(or more bits) –per-cell at the array level, constructed based on a combination of coarse and fine control that we have identified specifically for RRAM, from a large amount of data collected on a 1Mbits 1T4R $HfO_2$-based RRAM array, using 130nm CMOS process. We achieve 80% (for fresh) and 60% (for cycled devices) improvement over prior method in our apples-to-apples comparison (using the same resistance ranges for 8 states).

**Introduction –** Data storage in multi-level RRAM is based on different cell resistance values. For example, for 3 bits-per-cell memory, cell resistance values are distributed into 8 distinct ranges (or states) with gaps between for sensing margin. Compared to 1 bit-per-cell memory (2 states), programming is more difficult, since the ranges and the gaps between 8 resistance states are much smaller. The conventional incremental step pulse programming (ISPP) method (Fig. 2a, discussed in [1]), in which the programming voltage is increased for each programming pulse until the cells pass verify, is slow and ineffective for 3 bits per cell, because to control 8 distributions accurately, the step size must be small. Another method called fixed pulse program-verify (FPPV) algorithm discussed in [1] and [2] (Fig. 2b) can achieve excellent speed, but the accuracy attained is inadequate for 3-bit-per-cell RRAM programming. In this work, we demonstrate our algorithm, called State-Dependent Coarse-Fine Control (SDCFC), that can achieve tuning accuracy sufficient for 3-bit-per-cell RRAM without sacrificing programming speed, using a well-designed combination of a coarse resistance control phase (fast speed) and a fine resistance control phase (slow speed). The algorithm is compared to the popular FPPV algorithm using data taken from a 1Mbit 1T4R $HfO_2$-based RRAM array fabricated in a 130nm CMOS process. The cell schematic, technology used and array architecture are shown in Fig. 1 and Fig. 2. To facilitate an apples-to-apples comparison, both SDCFC and FPPV use the same resistance ranges (Fig. 4), allocated based on the sigma-based allocation (SBA) technique, where the resistance ranges are not uniform, but proportional to the standard deviation(s) of resistance values [3].

**State-Dependent Coarse-Fine Control Algorithm –** First, all cells are brought to level 7 by performing a blanket RESET operation, then they are sequentially programmed to 7 other levels (0, 1, 2, 3, 4, 5, 6) or they may stay at level 7 (Fig. 4). To achieve fast and accurate programming operation, SDCFC first uses a coarse resistance control phase (fast speed), to bring the cell resistance close to the target, then it employs a fine resistance control operation phase (slow speed) to nudge the cell resistance into the intended range, as shown in Fig. 5. We propose two close-to-target detection levels called Rmm[n] and RMM[n], which respectively define upper and lower boundaries for switching from coarse to fine-grained resistance tuning (Fig. 5). The span between Rmm[n] and RMM[n] is called the coarse range for state n. The novelty here is that the offsets from Rmm[n] and RMM[n] to Rm[n] and RM[n], respectively, are proportional to the width of the range RM[n] – Rm[n] (hence state-dependent), and the detection levels are on both sides of the distribution (unlike NAND or NOR flash). The question remains, what and if there are control knobs specifically for RRAM technology, that can provide coarse resistance control (fast speed) and fine resistance control (slow speed). From measured data (shown in Fig. 6) we have identified that the wordline voltage VWL (which is used to control the compliance current) is the coarse resistance control knob since the cell resistance change is very sensitive to the wordline voltage VWL. We have also identified that the source line voltage VSL and the bitline voltage VBL are the fine resistance control knobs, since the cell resistance change is a weak function of VBL and VSL. Leveraging these RRAM-specific control knobs, SDCFC realizes the coarse resistance control phase by applying a *state-dependent* SET pulse (with VWL and VBL specified in Table 1. Note that VBL is used in addition to VWL for position the cell resistance more accurately). If this pulse fails to bring the cell resistance into the coarse range, it will be RESET (by a coarse RESET pulse) back to state 7, and another state-dependent SET pulse is applied, and so on. Similarly, SDCFC realizes the fine resistance control phase (after the cell resistance has been brought into the coarse range) by stepping up the bitline voltage VBL (for fine SET) or the source line voltage VSL (for fine RESET), as the cell resistance change is less sensitive to VBL or VSL. Note that the step sizes for VBL and VSL are state-dependent because the offsets between Rmm[n] and RMM[n] to Rm[n] and RM[n] are also state-dependent (Table 1). The complete SDCFC algorithm is shown in Fig. 7.

**Results –** Table 2 shows the programming speed improvement of the SDCFC algorithm over the FPPV algorithm, 80% for fresh devices and 60% for 50K-cycled devices. The large amount of data is based on 4 1Mbit 1T4R devices. For each device, we measure 32K cells, so the total number of bits measured is 128Kbits. Note that for each 1T4R cell, we only use 1R out of 4R for this work. The SET or RESET pulse width is 200ns.

**Conclusion –** By carefully combining and optimizing the *state-dependent* coarse and fine programming phases with close-to-target verify detection levels, our SDCFC algorithm can achieve (based on a large amount of measured data) excellent performance and accuracy for 3-bit per cell RRAM memory with optimum energy efficiency. This algorithm can be easily adapted to support 2 bits-per-cell or more than 3 bits-per-cell RRAM.

**Acknowledgements –** This work was supported by DARPA, NCTU and the NSF Graduate Research Fellowship program.

**References**

[1] MH Lee et al., ICSICT (2016)
[2] F. M. Puglisi et al., IEEE Electron Device Letters, 2015.
[3] B. Le et al., TED (2019)

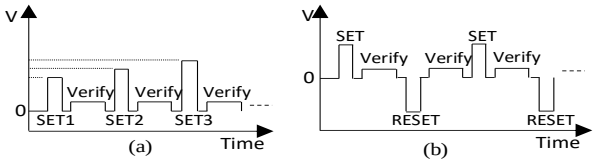


Fig. 1 (a) Conventional ISPP method, (b) Fixed pulse programming-verify method, discussed in [1].

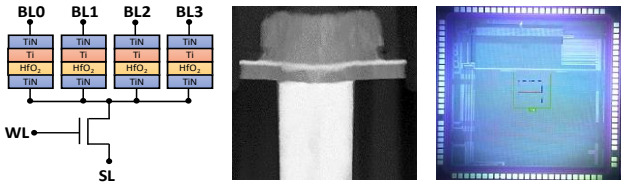


Fig. 2 (a) 1T4R RRAM cell used in this work, (b) X-sectional TEM of $HfO_2$ RRAM, (c) Die image (optical) of the 1Mbit array.

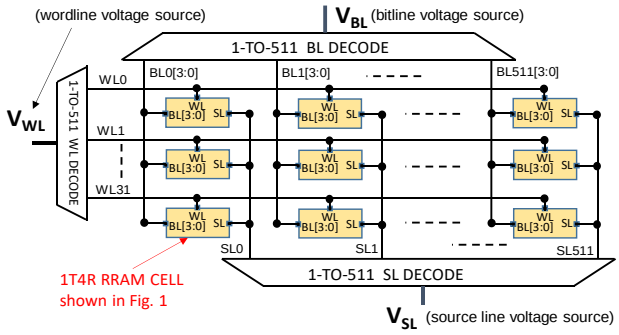


Fig. 3 1Mbit 1T4R array architecture.

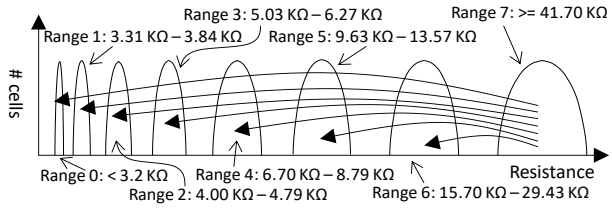


Fig. 4 Resistance ranges and the programming sequence which starts from range 7 after a blanket RESET.

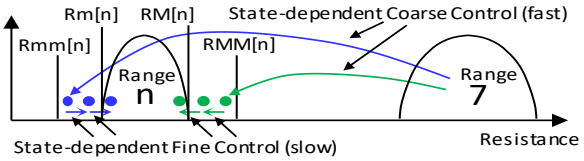


Fig. 5 SDCFC uses a combination of State-dependent Coarse Control operation and State-dependent Fine-control operation to achieve good speed and accuracy. For each range n, two detection levels Rmm[n] and RMM[n], n = 0, 1, ..., 7, are defined. Range 0 doesn't need to have Rmm[0] and range 7 doesn't need to have RMM[7].

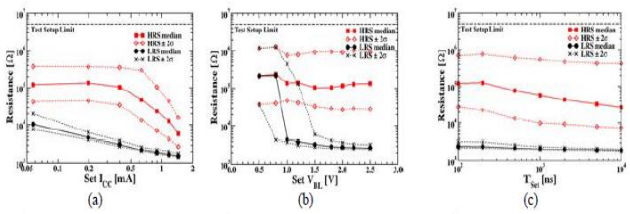


Fig. 6 Measured data that shows (a) fast SET speed with respect to wordline voltage VWL, and slow SET and RESET speed with respect to the bitline voltage VBL and the source line voltage VSL, respectively.

Table 1 VWL/VBL combination for each state, used in the coarse resistance control phase. Also show the Rmm[n] and RMM[n] values used in the fine resistance control phase. The values in the table are optimized from the data collected on the 1Mbit 1T4R array. VBL range is [1.6V – 2.5V] and VSL range is [2.5V – 3.4V] → Starting voltages for VBL and VSL are 1.6V and 2.5V, respectively. Coarse RESET pulse voltage condition is VWL = 4.5V, VBL = 0V, VSL = 2.5V.

| | Coarse Control | | Fine Control | | | |
|---|---|---|---|---|---|---|
| State | VWL (V) | VBL (V) | Rmm offset (kΩ) | RMM offset (kΩ) | VBL step size (V) | VSL step size (V) |
| n = 0 | 1.2 | 1.5 | 2.5 | 2.7 | 0.05 | 0.05 |
| n = 1 | 1.3 | 1.6 | 3.6 | 3.7 | 0.06 | 0.06 |
| n = 2 | 1.4 | 1.7 | 5.7 | 5.9 | 0.07 | 0.07 |
| n = 3 | 1.5 | 1.8 | 7.8 | 7.9 | 0.08 | 0.08 |
| n = 4 | 1.6 | 1.9 | 9.9 | 10.9 | 0.09 | 0.09 |
| n = 5 | 1.7 | 2.0 | 11.0 | 11.9 | 0.10 | 0.10 |
| n = 6 | 1.8 | 2.1 | 13.1 | 12.9 | 0.11 | 0.11 |
| n = 7 | 1.9 | 2.2 | 15.2 | 15.9 | 0.12 | 0.12 |

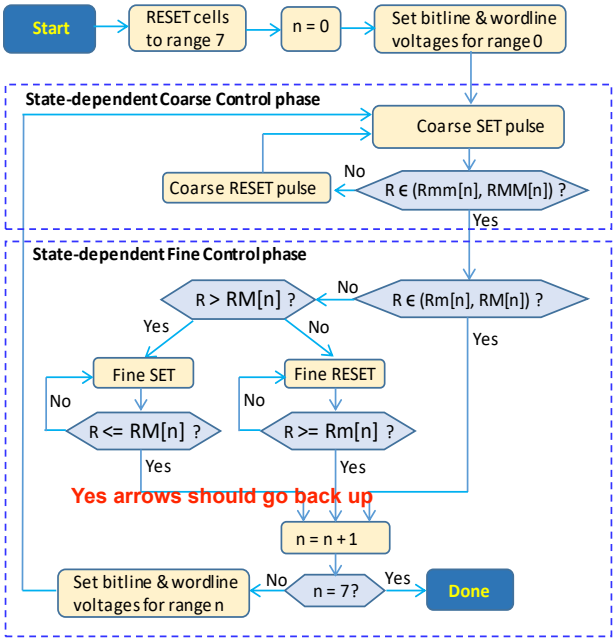


Fig. 7 Algorithm diagram. Note that the Fine Control phase has a limit on the number of pulses, over which the program operation will be declared "fail" (not specified in the diagram). Fine SET and Fine RESET conditions must be optimized such that the resistance change ΔR per fine-control pulse is smaller than the range width (to avoid resistance overshooting across the entire resistance range).

Table 2 Measurement result: Comparison between our SDCFC algorithm with the currently popular fixed pulse program-verify algorithm. The number of pulses is counted until the number of cells with resistance outside the target ranges (0, 1, ..., 6) is less than 1% of the total number of cells. 1% represents the error tolerance that we assume an error correction scheme can correct. For 3 or more bits-per-cell, low density parity check (LDPC) error correction is usually used.

| | Fresh | | After 50K cycles | |
|---|---|---|---|---|
| Device number | # of PGM pulses for SDCFC | # of PGM pulses for FPPV | # of PGM pulses for SDCFC | # of PGM pulses for FPPV |
| 1 | 10 | 20 | 15 | 15 |
| 2 | 12 | 22 | 17 | 17 |
| 3 | 11 | 21 | 16 | 16 |
| 4 | 9 | 19 | 14 | 14 |
| Average Improvement | 80 % | | 60 % | |