Supplementary Materials: Assessment and Improvement of the Pattern Recognition Performance of Memdiode-Based Cross-Point Arrays with Randomly Distributed Stuck-at-Faults

Fernando L. Aguirre, Sebastián M. Pazos, Félix Palumbo, Antoni Morell, Jordi Suñé and Enrique Miranda

Table S1. Summary of reported studies that considers the case of Stuck-at-Faults in RRAM based neural networks. Those works accounting for a strategy to minimize the impact of SAFs are pointed out. Although the vast literature, notice that frameworks based on SPICE simulations (that is electrical circuit simulations) comprising real memristor devices and accounting for the CPA non-idealities are rare.

Work	Sim.?	RRAM/PCM Compact model	Realistic RRAM model?	Platform	SPICE compat.?	CPA non-idealities	SAF Mitigation tech.?	Structure
This Work	Yes	QMM	Yes	SPICE (HSPICE)	Yes	Yes	Yes	CPA
Mehonic *et al.*[1]	Yes	No	Yes	Python	No	No R_L	No	CPA
Dias *et al.*[2]	Yes	Pino	No	LTSpice	Yes	No	No	CPA
Zhang *et al.*[3]	Yes	---	No	C++ MATLAB	No	No	Yes	CPA
Zhang *et al.*[4]	Yes	---	No	C++ MATLAB	No	No	Yes	CPA
Cristiano *et al.*[5]	Mixed	No	Yes (Jump Table)	No data	No	N/A	No	2T2R+ 3T1C
Romero *et al.*[6]	Mixed	No	Yes (Jump Table)	No data	No	N/A	Yes	2T2R+ 3T1C
Liu *et al.*[7]	No	No	N/A	Hardware	No	Yes (non-controllable)	Yes	CPA
Chen *et al.*[8]	No	No	N/A	Hardware	No	Yes (non-controllable)	Yes	CPA
Xia *et al.*[9]	Yes	No	No	No data	No	No	Yes*[1]	CPA
Xia *et al.*[10]	Yes	No	No	No data	No	No	Yes	CPA
Chen *et al.*[11]	No	N/A	N/A	Hardware	No	Yes (non-controllable)	No	1T1R
Woo *et al.*[12]	Yes	Yes*	N/A	MATLAB	No	No	Yes	CPA
Huang *et al.*[13]	Yes	No	No	Python	No	No	No	CPA
Yeo *et al.*[14]	Yes	No	No	No data	No	No	Yes*[2]	CPA
Van Pham *et al.*[15]	Yes	Yes*[3] (verilog-A)	Yes	SPICE (Spectre)	Yes	No	Yes*[2]	1T1R

*[1]excessive redundancy, *[2]Costly network re-training, *[3]Not a SPICE compact model.

Table S2. Memdiode SPICE Model Code.

```
.subckt memdiode p n
.param H0=0 CH0=1e-4 beta=0.5
*Transition parameters
.param etaset=10.5 vset=0.78 etares=7.2 vres=-0.79 gam=1e-02 gam0=0 isb=1e-08, vt=8.7e-01 Hmin=2e-6
*I-V parameters
.param imax=6.06e-3  imin=1.16e-4 alphamax=3.5 alphamin=5.6 rsmax=47 rsmin=47 RPP=1e9 ri=1
*Auxiliary functions
.param I0(x)='x>Hmin ? imin +(imax-imin)*x : 0'
.param VSB(x)='x>isb ? vt : vset'
.param I0(x)='imax*x+imin*(1-x)'
.param A(x)='alphamax*x+alphamin*(1-x)'
.param RS(x)='rsmax*x+rsmin*(1-x)'
.param VSB(x)='x>isb ? vt : vset'
.param ISF(x)='gam==0 ? 1 : (pwr(x,gam)-gam0)'
.param S(x)='1/(1+exp(-etaset*(x-VSB(I(RS)))))'
.param R(x)='1/(1+exp(-etares*ISF(V(H))*(x-vres)))'
*********H-V*********
GH   gnd! H       cur='min(R(V(C,n)),max(S(V(C,n)),V(H)))'
Rpar gnd! H       R=1
CH   H    gnd!  C='CH0' IC='H0'
*********I-V*********
RE   p    C    R='ri'
RS   C    B    R='RS(V(H))'
GD   B    n    cur='I0(V(H))*(exp(beta*A(V(H))*V(B,n))-exp(-(1-beta)*A(V(H))*V(B,n)))'
RB   p    n    R='RPP'
.ends memdiode
```

`S(x)` and `R(x)` are the positive and negative ridge functions $\Gamma^{+}(V)$ and $\Gamma^{-}(V)$. `A(x)` and `RS(x)` stands for the α and *R* memdiode parameters, which are a function of the memory state. The memory state λ is represented by `V(H)`, with H0 being the initial state. Parameters modelling the HRS-LRS transition are `etaset`, `etareset`, `vset/vt` and `vreset` for η^{+}, η^{-}, V^{+}(device electroformed/no-electroformed) and V^{-}, respectively. if statements in `I0(x)` (with threshold `Hmin`) and `VSB(X)` (with threshold `isb`) allows representing the pre- and post- forming *I-V* characteristics. Voltage-controlled current sources are used to implement Eq 1. (`GD` and resistor `RS`) and Eq. 2 (`GH` and capacitor `CH`). The antiparallel diodes are modelled by the controlled current source `GD` in the script. `beta` defines whether the conduction is symmetric between positive and negative applied voltages (`beta`=0.5) or not (`beta`≠0.5). Eq.(1) corresponds to the `beta`=1 case but is symmetrized using the absolute value function. The model is written in terms of the HSPICE syntax.

S1. Methods: Supplementary Details

S1.1. Device Fabrication and Measurement

Experimental data used in this work to test the QMM accuracy were extracted from references[16,17]. Two kind of devices were considered. First, HfO_2-based structures with a TiN (70 nm) bottom electrode and Pt (30 nm) top electrode were investigated. The 5nm-thick HfO_2 film was prepared by Atomic Layer Deposition (ALD). The electrical characterization of the MIM structure was performed using a Keithley 4200 Semiconductor Characterization System. The devices required an initial electroforming step (~-1.5 V). While a current compliance of 0.5 mA was used in the positive bias region, no limit was applied for negative voltages. The second set of devices consisted in LCMO ($La_{1-x}Ca_xMnO_3$)-based films 88nm-thick with a composition of x=2/3. The material was grown by the Pulsed Laser Deposition (PLD) technique on top of commercial Pt/Ti/SiO_2/Si substrates. The Pt layer acting as bottom electrode has a thickness of 100 nm. The devices were characterized using pulsed *I-V* measurements with a Keithley 2612. A current compliance of 10mA was used during the SET process. The time width of the applied pulses was 1 ms. No electroforming step is required to activate the switching of the LCMO samples.

S1.2. Procedure for SPICE CPA Creation, Training, and Simulation

First, a software-based SLP or MLP with n^2 inputs, 10 outputs and a number N (N=0 if considering a SLP) of hidden neural layers (each of them comprising m_i neurons) is created and trained using a given dataset of $n \times n$ px. images (as those represented in Supplementary Fig. 1b) with m classes. For the sake of simplicity, *ex-situ* supervised learning is considered using the Scaled Conjugate Gradient [18] (SCG) as the training algorithm, as it provides a good trade-off between accuracy and learning time for the different datasets considered [19]. The possible impact of the learning method has been discussed elsewhere showing no statistically significant differences in terms of the inference accuracy for this scenario. This produces *N+1* weight matrices $W_{M_k} \in \mathbb{R}$, with k ∈ {1,2,…,N+1} (for instance for two hidden layers with m_1 and m_2 neurons each, three weight matrices W_{M_1}, W_{M_2} and W_{M_3} are obtained, with sizes $n^2 \times m_1$, $m_1 \times m_2$ and $m_2 \times 10$, respectively). Each weight matrix is further decomposed as $W_M = W_M{}^+ - W_M{}^-$ based on Eqns. (S1) and (S2) as proposed in Ref.[20] as:

$$w^+_{M_{i,j}} = \begin{cases} w_{M_{i,j}}, & w_{M_{i,j}} > 0 \\ 0, & w_{M_{i,j}} \leq 0 \end{cases} \quad (S1)$$

$$w^-_{M_{i,j}} = \begin{cases} 0, & w_{M_{i,j}} \geq 0 \\ -w_{M_{i,j}}, & w_{M_{i,j}} < 0 \end{cases} \quad (S2)$$

both (W_M^+ and W_M^-) comprising only positive elements. This allows rendering both positive and negative synaptic weights in W_M as well as doubling the dynamic range and reducing the noise and variability susceptibility [21].

In the next step, the conductance matrices G_M^+ and G_M^- associated to each synaptic layer (for the particular case of the SLP, these having a

size of $n^2 \times m$, thus totalling $2 \cdot n^2 \cdot m$ synapses) to be mapped into the CPAs are calculated by the linear transformation [22,23] shown in Eq. (S3):

$$G_M^{+(-)} = (G_{max} - G_{min}) W_{M_{Norm}}^{+(-)} + G_{min} \quad (S3)$$

where $[G_{min}, G_{max}]$ is a selected conductance range for a linear computation in matrix-vector calculations. For simplicity, we consider $G_{max} = G_{LRS} = 1/R_{ON}$ and $G_{min} = G_{HRS} = 1/R_{OFF}$, with R_{ON} and R_{OFF} defined by the *I-V* characteristic of the memristor model play evaluated at a given read voltage (V_{read}). Then the normalized (elements are in the range [0,1]) weight matrix $W_{M_{Norm}}^{+(-)}$ corresponding to the positive (negative) weight matrix are computed by any of the following normalization methods NM_1-NM_3, indicated by Eq. (S4)-(S6):

$$MN_1: \quad W_{M_{Norm}}^{+(-)} = \frac{W_M^{+(-)}}{max\{abs\{W_M\}\}} \quad (S4)$$

$$MN_2: \quad W_{M_{Norm}}^{+(-)} = \frac{W_M^{+(-)}}{max\{W_M\} - min\{W_M\}} \quad (S5)$$

$$MN_3: \quad w_{M_{i,j}} = \begin{cases} 1, & w_{M_{i,j}} > \mu_{W_M} + n\sigma_{W_M} \\ \dfrac{w_{M_{i,j}}}{\mu_{W_M} - n\sigma_{W_M}}, & \mu_{W_M} - n\sigma_{W_M} < w_{M_{i,j}} < 0 \\ \dfrac{w_{M_{i,j}}}{\mu_{W_M} + n\sigma_{W_M}}, & 0 < w_{M_{i,j}} < \mu_{W_M} + n\sigma_{W_M} \\ -1, & w_{M_{i,j}} < \mu_{W_M} - n\sigma_{W_M} \end{cases} \quad (S6)$$

where $max\{W_M\}$ and $min\{W_M\}$ are the maximum and minimum synaptic weight values in W_M and μ_{W_M} $\sigma_{W_M}$are the mean and standard deviation of the synaptic weights in W_M. In this way, the W_M^+ and W_M^- matrices are converted to conductance values within the range $[G_{HRS}, G_{LRS}]$.

The subsequent sub-routines write down the circuit netlist for the dual- $n^2 x\, m_i$, $m_i\, x\, m_{i+1}$, …, $m_N\, x\, m$ memristor CPA-based MLP, adding the parasitic line resistance (R_L), connection scheme, and control logic necessary for the inference phase. As in this work we focus on the artificial synapses modelling using the memdiode model, hidden neurons in the k^{th} hidden neural layer connecting the two adjacent layers of synapses *k-1* and *k+1* are implemented in terms of a behavioural SPICE model. The model for each neuron involves a Trans-Impedance Amplifier (TIA) that translates the output current in the associated bitline on the *i-1* synaptic layer to a voltage which is fed to a non-linear activation function and then propagated to the corresponding wordline in the *i+1* synaptic layer. In this paper, we consider a log-sigmoidal ($1/(1+e^{-x})$) activation function, though a tan-sigmoidal activation function could be used as well. Two approaches were followed in order to improve the overall read margin as shown in the simplified equivalent circuit depicted in Supplementary Fig. 1c: *i)* A Dual Side Connection (DSC) scheme and *ii)* the partitioning of the G_M^+ and G_M^- matrices in smaller structures [19,21,24]. Despite the increased peripheral circuitry complexity, DSC improves the voltage delivery to each synapse [13] by connecting both wordline terminals to the input

stimuli. Similarly, the small size of each partition helps reducing the parasitic voltage drop along the line interconnections. Exploding the integrability of the CPA with CMOS circuitry, the vertical interconnects connecting the outputs of each partition may be placed under the CPAs, as well as the analogue sensing electronics, minimising the area overhead of the partitioned architecture [21].

Finally, each memristor at the wire intersections in the CPA is set to the corresponding conductance value from the G_M^+ and G_M^- matrices associated to that specific synaptic layer by adjusting the control parameter λ or memory state (H_0 in the script from Supplementary Table 2). The required value of λ is obtained by solving Eq. (1) from the main manuscript for $I = g_{i,j} \cdot V$, with $g_{i,j}$ being each of the matrix elements in $G_M^{+(-)}$. In this way, during the inference process each of the test images is presented to the CPA as a vector of analogue voltages in the range [0, V_{read}]. Once the circuit netlist is written, it is passed to a SPICE simulator which evaluates the voltage and current distributions in the CPA circuit while it processes and classifies the input images, and then passes the resulting waveforms back to the MATLAB routine for evaluation and metrics extraction.

S1.3. Datasets and Input Stimuli

Two image datasets are considered for the training and inference procedures implemented in this work, namely the MNIST (Modified National Institute of Standards and Technology) dataset of handwritten digits and the frontal images of the Extended Yale Face Dataset B [25]. Each of them comprises a series of m input feature vectors ($x(m)$ for sample m) and a target output vector ($t(m)$, with 10 dimensions for the MNIST case -each corresponding to one digit- and 38 for the Yale Face Dataset B -one for each person in the dataset-). For the classification problem, $t_c(m) = 1$ if sample m belongs to class c and $t_c(m) = 0$ otherwise. The input feature vectors ($n^2 \times 1$) are the unrolled grayscale pixel values of the two-dimensional images ($n \times n$ px.). Pixel's brightness is codified in 256 gray levels between 0 (fully OFF, black) and 1 (fully ON, white). The MNIST dataset contains 60,000 training images and 10,000 testing images, both in grayscale and with a 28×28 pixels resolution[26]. A few examples of these images can be seen in Supplementary Fig. 1b, where the x and y axes stand for the pixel index. The Extended Yale Face Database B contains images of 38 different subjects with roughly 64 different illumination conditions and 9 different poses. For this work, only the frontal images were considered, and then centred and cropped to a 32×32 px. resolution. We have obtained full permissions to use the images from the Extended Yale Face Database B and we are compliant with Yale's policy of reuse/use of these images (http://vision.ucsd.edu/content/extended-yale-face-database-b-b). Informed consent for publication was obtained from all the study participants. The images from both datasets were further down sampled using bicubic interpolation to 8×8 and 16×16 px. (MNIST) and 16×16 px. (Yale Faces Dataset B) to allow smaller SLPs, which reduce the inference accuracy degradation due to the line resistance as well as speed up the simulations. Finally, the input stimuli are obtained by

scaling the input feature vector by a voltage V_{read}. V_{read} is chosen such as to prevent altering the memristor state during the inference simulation.

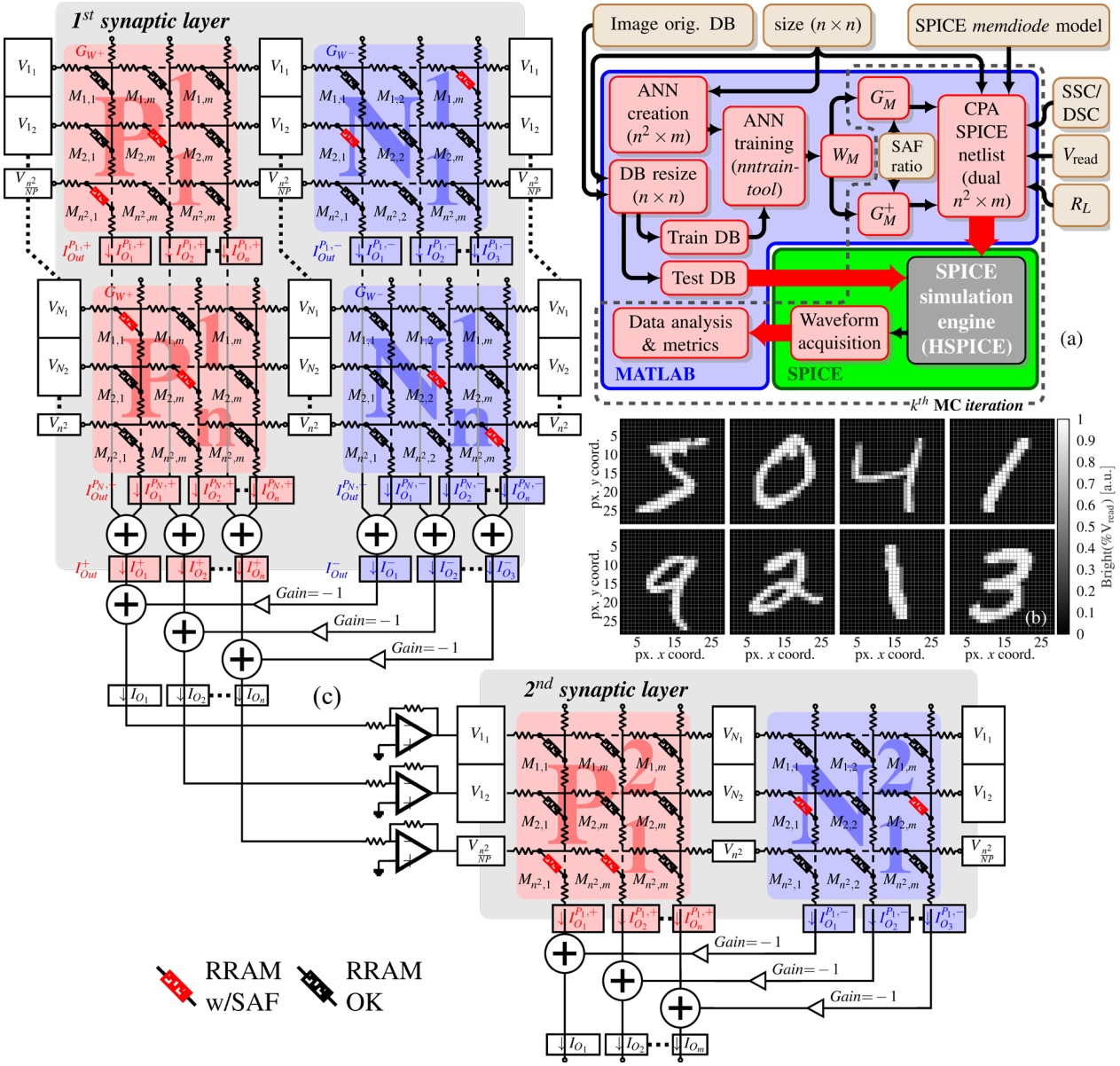

Figure S1. (a) Flowchart diagram for the simulation procedure. Starting with the memdiode SPICE model, image dataset, image size, R_L, V_{read}, connection scheme, partitioning and Stuck-at-Fault (SAF) ratio, the routine creates the dataset, trains the SLP, implements it with a pair of Cross Point Arrays (CPAs), performs the simulations and processes the results. Note that an iterative loop is indicated by the dashed line: It represents each of the Monte Carlo runs execute for each SAF ratio. (b) Samples of the MNIST database[26] considered in this article. In all cases images are represented in 28×28 px. Pixel brightness (or intensity) is codified in 256 levels ranging from 0 (fully OFF, black) to 1 (fully ON, white). (c) Simplified equivalent circuit schematic for a partitioned CPA-based MLP. Each CPA in the synaptic layer is subdivided into N identically sized partitions to minimize parasitic voltage drops. Partial output current vectors are indicated in the output of each partition. Randomly distributed SAFs are indicated by the red devices, as explained in the legend. Note that the SLP would be equal to considering the first part of the circuit, that is, up to the TIA.

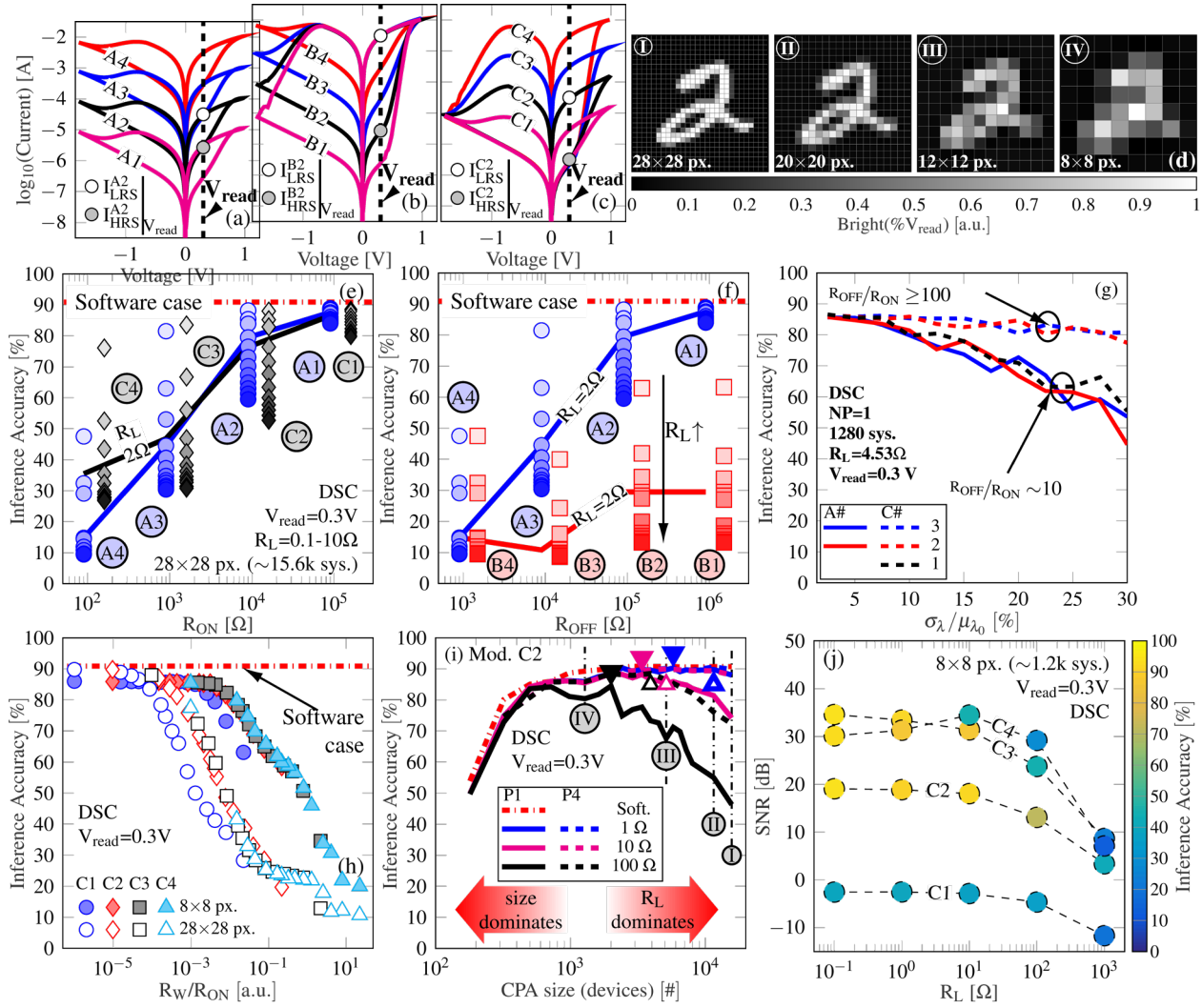

Figure S2. Summary of the parasitics impact on the inference accuracy, reproduced from Ref. [19]. Model plays (a) *A1-A4*, (b) *B1-B4* and (c) *C1-C4*. (d) Readability loss of the MNIST images as the images are downscaled. Inference accuracy as function of the QMM (e) R_{ON} and (f) R_{OFF} values. R_{ON} is found to have a major impact on the accuracy loss. (g) Nevertheless, a R_{OFF}/R_{ON} of at least 100 is suggested to minimize the sensitivity to the device-to-device variability. The CPA electrical characteristics also plays a role on the accuracy: Note that the line resistance (R_L) also plays a role on the accuracy degradation as shown in (h), with smaller images exhibiting a reduced dependence on it. (i) In fact, there is an optimum image size (and thereby CPA size) given by the trade-off between readability loss and the parasitic voltage drops along the line interconnections. (j) Last but not least, the model plays involving very small currents also suffer from a SNR reduction, which also compromises the inference accuracy. Adapted from Ref.[19]

References

1. Mehonic, A.; Joksas, D.; Ng, W.H.; Buckwell, M.; Kenyon, A.J. Simulation of inference accuracy using realistic rram devices. *Front. Neurosci.* **2019**, *13*, 1–15, doi:10.3389/fnins.2019.00593.

2. Dias, C.; Guerra, L.M.; Ventura, J.; Aguiar, P. Memristor-based Willshaw network: Capacity and robustness to noise in the presence of defects. *Appl. Phys. Lett.* **2015**, *106*, doi:10.1063/1.4922148.

3. Zhang, B.; Uysal, N.; Fan, D.; Ewetz, R. Handling Stuck-at-faults in Memristor Crossbar Arrays using Matrix Transformations. In Proceedings of the Proceedings of the Asia and South Pacific Design Automation Conference, ASP-DAC; Institute of Electrical and Electronics Engineers Inc.: New York, NY, USA, 2019; pp. 474–479.

4. Zhang, B.; Uysal, N.; Fan, D.; Ewetz, R. Handling Stuck-at-fault Defects using Matrix Transformation for Robust Inference of DNNs. *IEEE Trans. Comput. Des. Integr. Circuits Syst.* **2019**, 1–1, doi:10.1109/tcad.2019.2944582.

5. Cristiano, G.; Giordano, M.; Ambrogio, S.; Romero, L.P.; Cheng, C.; Narayanan, P.; Tsai, H.; Shelby, R.M.; Burr, G.W. Perspective on training fully connected networks with resistive memories: Device requirements for multiple conductances of varying significance. *J. Appl. Phys.* **2018**, *124*, doi:10.1063/1.5042462.

6. Romero, L.P.; Ambrogio, S.; Giordano, M.; Cristiano, G.; Bodini, M.; Narayanan, P.; Tsai, H.; Shelby, R.M.; Burr, G.W. Training fully connected networks with resistive memories: Impact of device failures. *Faraday Discuss.* **2019**, *213*, 371–391, doi:10.1039/c8fd00107c.

7. Liu, B.; Li, H.; Chen, Y.; Li, X.; Wu, Q.; Huang, T. Vortex: Variation-aware training for memristor X-bar. *Proc. - Des. Autom. Conf.* **2015**, *2015-July*, 1–6, doi:10.1145/2744769.2744930.

8. Chen, L.; Li, J.; Chen, Y.; Deng, Q.; Shen, J.; Liang, X.; Jiang, L. Accelerator-friendly neural-network training: Learning variations and defects in RRAM crossbar. In Proceedings of the Proceedings of the 2017 Design, Automation and Test in Europe, DATE 2017; Institute of Electrical and Electronics Engineers Inc., 2017; pp. 19–24.

9. Xia, L.; Huangfu, W.; Tang, T.; Yin, X.; Chakrabarty, K.; Xie, Y.; Wang, Y.; Yang, H. Stuck-at Fault Tolerance in RRAM Computing Systems. *IEEE J. Emerg. Sel. Top. Circuits Syst.* **2018**, *8*, 102–115, doi:10.1109/JETCAS.2017.2776980.

10. Xia, L.; Liu, M.; Ning, X.; Chakrabarty, K.; Wang, Y. Fault-Tolerant Training with On-Line Fault Detection for RRAM-Based Neural Computing Systems. In Proceedings of the Proceedings - Design Automation Conference; Institute of Electrical and Electronics Engineers Inc., 2017; Vol. Part 12828.

11. Chen, C.Y.; Shih, H.C.; Wu, C.W.; Lin, C.H.; Chiu, P.F.; Sheu, S.S.; Chen, F.T. RRAM defect modeling and failure analysis based on march test and a novel squeeze-search scheme. *IEEE Trans. Comput.* **2015**, *64*, 180–190, doi:10.1109/TC.2014.12.

12. Woo, J.; Van Nguyen, T.; Kim, J.H.; Im, J.P.; Im, S.; Kim, Y.; Min, K.S.; Moon, S.E. Exploiting defective RRAM array as synapses of HTM spatial pooler with boost-factor adjustment scheme for defect-tolerant neuromorphic systems. *Sci. Rep.* **2020**, *10*, 1–8, doi:10.1038/s41598-020-68547-5.

13. Huang, L.; Diao, J.; Nie, H.; Wang, W.; Li, Z.; Li, Q.; Liu, H. Memristor Based Binary Convolutional Neural Network Architecture With Configurable Neurons. *Front. Neurosci.* **2021**, *15*, 1–14, doi:10.3389/fnins.2021.639526.

14. Yeo, I.; Chu, M.; Gi, S.G.; Hwang, H.; Lee, B.G. Stuck-at-Fault Tolerant Schemes for Memristor Crossbar Array-Based Neural Networks. *IEEE Trans. Electron Devices* **2019**, *66*, 2937–2945, doi:10.1109/TED.2019.2914460.

15. Van Pham, K.; Van Nguyen, T.; Min, K.S. Partial-gated memristor crossbar for fast and power-efficient defect-tolerant training. *Micromachines* **2019**, *10*, 1–18, doi:10.3390/mi10040245.

16. Blasco, J.; Jančovič, P.; Fröhlich, K.; Suñé, J.; Miranda, E. Modeling of the switching I-V characteristics in ultrathin (5 nm) atomic layer deposited HfO 2 films using the logistic hysteron. *J. Vac. Sci. Technol. B, Nanotechnol. Microelectron. Mater. Process. Meas. Phenom.* **2015**, *33*, 01A102, doi:10.1116/1.4900599.

17. Miranda, E.; Román Acevedo, W.; Rubi, D.; Lüders, U.; Granell, P.; Suñé, J.; Levy, P. Modeling of the multilevel conduction characteristics and fatigue profile of Ag/La1/3Ca2/3MnO3/Pt structures using a compact memristive approach. *J. Appl. Phys.* **2017**, *121*, 205302, doi:10.1063/1.4984051.

18. Møller, M.F. A scaled conjugate gradient algorithm for fast supervised learning. *Neural Networks* **1993**, *6*, 525–533, doi:10.1016/S0893-6080(05)80056-5.

19. Aguirre, F.L.; Pazos, S.M.; Palumbo, F.; Suñé, J.; Miranda, E. Application of the Quasi-Static Memdiode Model in Cross-Point Arrays for Large Dataset Pattern Recognition. *IEEE Access* **2020**, *8*, 1–1, doi:10.1109/ACCESS.2020.3035638.

20. Prezioso, M.; Merrikh-Bayat, F.; Hoskins, B.D.; Adam, G.C.; Likharev, K.K.; Strukov, D.B. Training and operation of an integrated neuromorphic network based on metal-oxide memristors. *Nature* **2015**, *521*, 61–64, doi:10.1038/nature14441.

21. Fouda, M.E.; Lee, S.; Lee, J.; Eltawil, A.; Kurdahi, F. Mask Technique for Fast and Efficient Training of Binary Resistive Crossbar Arrays. *IEEE Trans. Nanotechnol.* **2019**, *18*, 704–716, doi:10.1109/tnano.2019.2927493.

22. Hu, M.; Strachan, J.P.; Li, Z.; Grafals, E.M.; Davila, N.; Graves, C.; Lam, S.; Ge, N.; Yang, J.J.; Williams, R.S. Dot-product engine for neuromorphic computing. In Proceedings of the DAC '16: Proceedings of the 53rd Annual Design Automation Conference; Association for Computing Machinery: New York, NY, USA, 2016; pp. 1–6.

23. Liu, C.; Hu, M.; Strachan, J.P.; Li, H.H. Rescuing Memristor-based Neuromorphic Design with High Defects. In Proceedings of the Proceedings - Design Automation Conference; Institute of Electrical and Electronics Engineers Inc., 2017; Vol. Part 12828.

24. Liang, J.; Yeh, S.; Simon Wong, S.; Philip Wong, H.S. Effect of wordline/bitline scaling on the performance, energy consumption, and reliability of cross-point memory array. *ACM J. Emerg. Technol. Comput. Syst.* **2013**, *9*, 1–14, doi:10.1145/2422094.2422103.

25. Georghiades, A.S.; Belhumeur, P.N.; Kriegman, D.J. From Few to Many: Illumination Cone Models for Face Recognition under Variable Lighting and Pose. *IEEE Trans. Pattern Anal. Mach. Intell.* **2001**, *23*, 643–660.

26. LeCun, Y.; Cortes, C.; Burges., C.J.C. MNIST handwritten digit database, Yann LeCun, Corinna Cortes and Chris Burges Available online: http://yann.lecun.com/exdb/mnist/ (accessed on Nov 21, 2019).