Electronic Supplementary Material (ESI) for Materials Advances.

Supporting Information of

Reservoir computing using back-end-of-line SiC based memristors

Dongkai Guo,[a] Omesh Kapur,[a] Peng Dai,[a] Yisong Han,[b] Richard Beanland,[b] Liudi Jiang,[c] C. H. (Kees) de Groot[a] and Ruomeng Huang[a]*

[a]School of Electronic and Computer Science, University of Southampton, Southampton, United Kingdom
[b]School of Physics, Warwick University, Coventry, United Kingdom
[c]School of Engineering, University of Southampton, Southampton, United Kingdom

Email: r.huang@soton.ac.uk

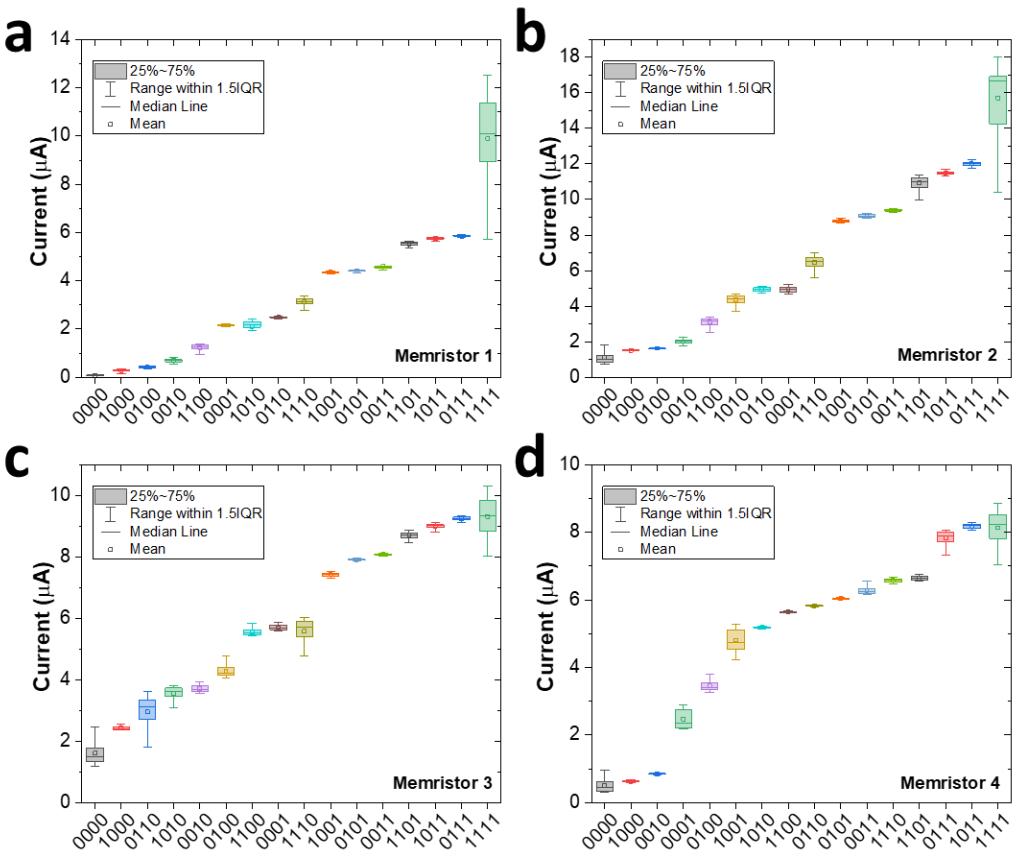

Figure S1. The current output of all 16 pulse trains in the 4-bit system from 4 SiC based memristors.

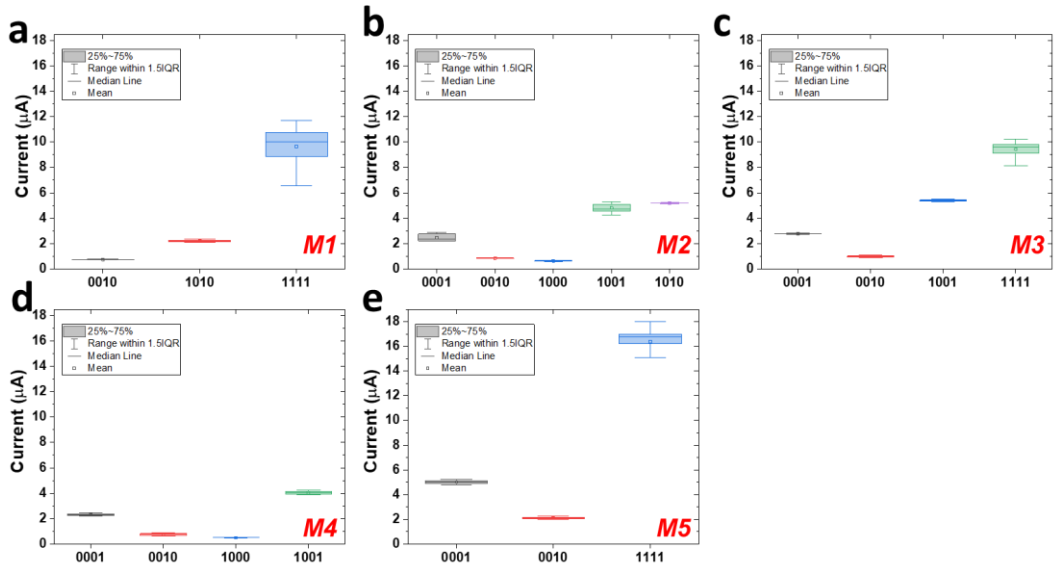

Figure S2. Reservoir states used to train the readout network for all five memristor devices, which is also the states for each row, (a) the first, (b) the second, (c) the third, (d) the fourth, and (e) the fifth.

Readout network training: A softmax activation function (equation S1) follows the weight matrix during training and validation to scale the probability vector's value in the range of 0 to 1 with a total of 1. Applying the cross entropy (Equation S2) as a loss function in a multinominal logistic regression calls for and benefits from this non-linear process. This loss function projects the probability distribution difference between prediction ($\hat{y}$) and ground truth (y), thus SoftMax may avoid the value loss function less than 0 and make the training process converge quickly and increase the robustness of the network. Because the non-linear behaviour is challenging to implement in the circuit, the soft max activation function is not used during testing. However, the categorization outcomes won't be altered by the removal of SoftMax. The highest value still remains in the same index as before softmax was removed, despite the fact that SoftMax normalises the probability vector's sum and increases the probability difference between the values in the vector. In order to increase learning stability and speed, we can train the neural network with SoftMax; nevertheless, for practical purposes, classifications must only be performed using the weight matrix. Table S1 lists how the letter from 0 to 9 are represented by the digits of 0-9 in dataset. Table S2 lists the specific training hyperparameters in detail.

$$\grave{x_i} = \frac{e^{x_i}}{\sum_i^n e^{x_i}} \qquad \text{(Equation S1)}$$

$$CrossEntropy = -\sum_i^n y_i \log \hat{y}_i \qquad \text{(Equation S2)}$$

Table S1 The Mapping relationship between labels and letters.

Label	0	1	2	3	4	5	6	7	8	9
Letter	0	1	2	3	4	5	6	7	8	9

Table S2 The neural network training hyperparameters

Hyperparameter	**Values**
Epochs	1000
Learning rate	$1 \cdot 10^{-3} \cdot \left[1 - Max(0, \frac{epoch - 500}{500})\right]$
Batch size	8
Optimizer	Adam(default)
Loss function	Cross entropy