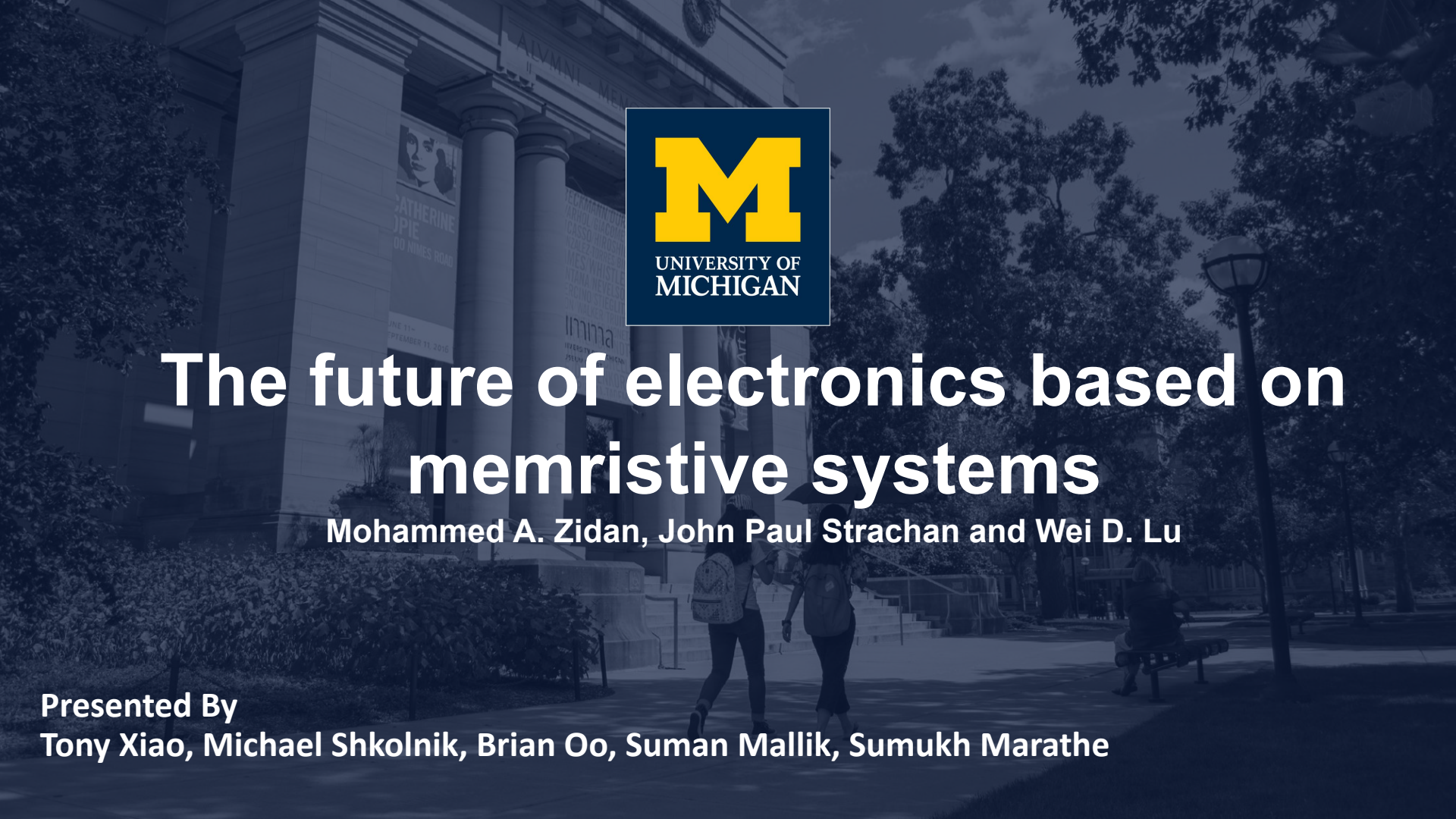


The future of electronics based on memristive systems

Mohammed A. Zidan, John Paul Strachan and Wei D. Lu

Presented By
Tony Xiao, Michael Shkolnik, Brian Oo, Suman Mallik, Sumukh Marathe

Introduction

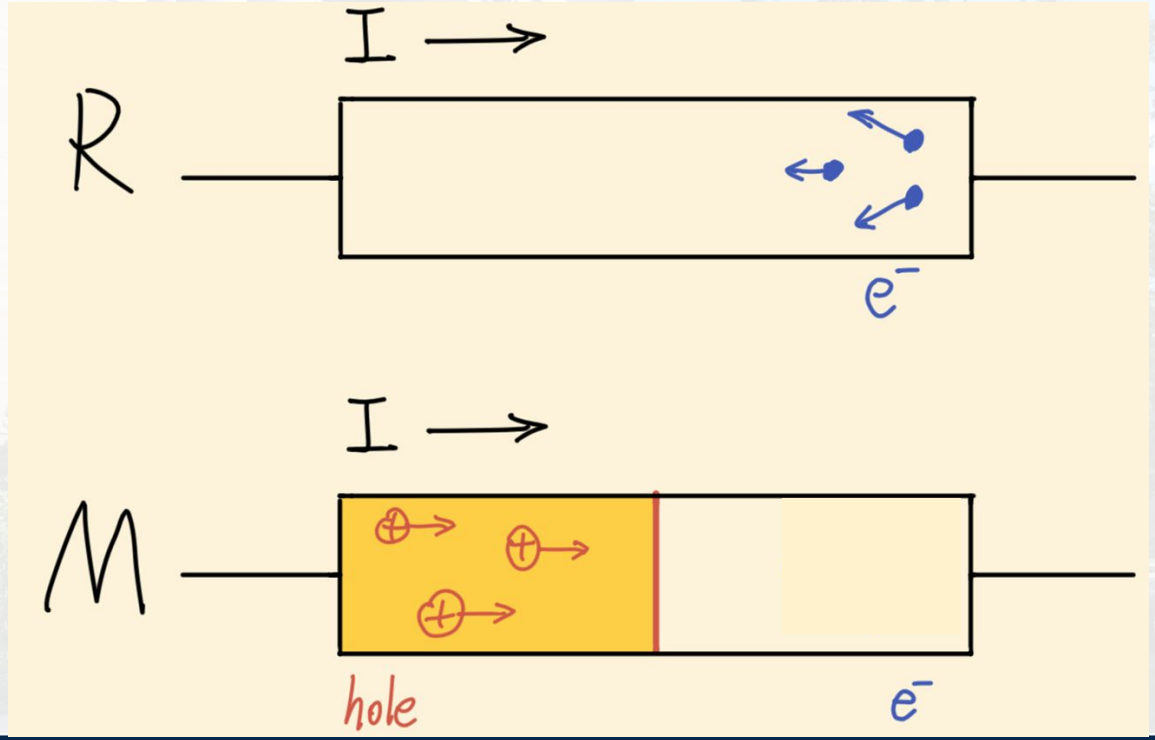
- CMOS device scaling reached bottleneck
  Memristors are the future

What’s a memristor

- Resistor with memory
- Variable resistor, but only has 2 connections
- Remember the resistant it’s set to

What's a memristor actually

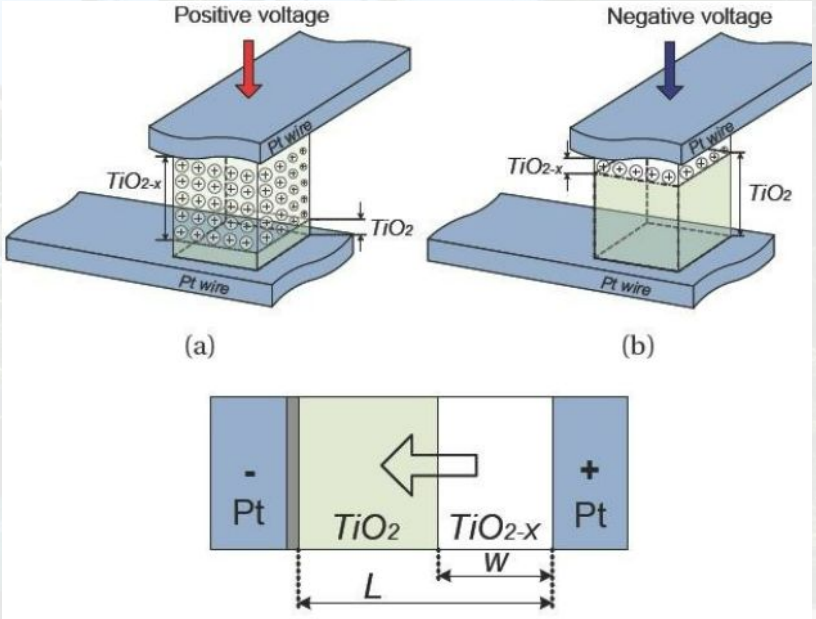
- Memristor = charge-dependent resistor



What's good about a memristor

- Simple => small
- Fast switching
- Long endurance
- Low programming energy

- But: no material can do all these



State of the art (Classical computing)

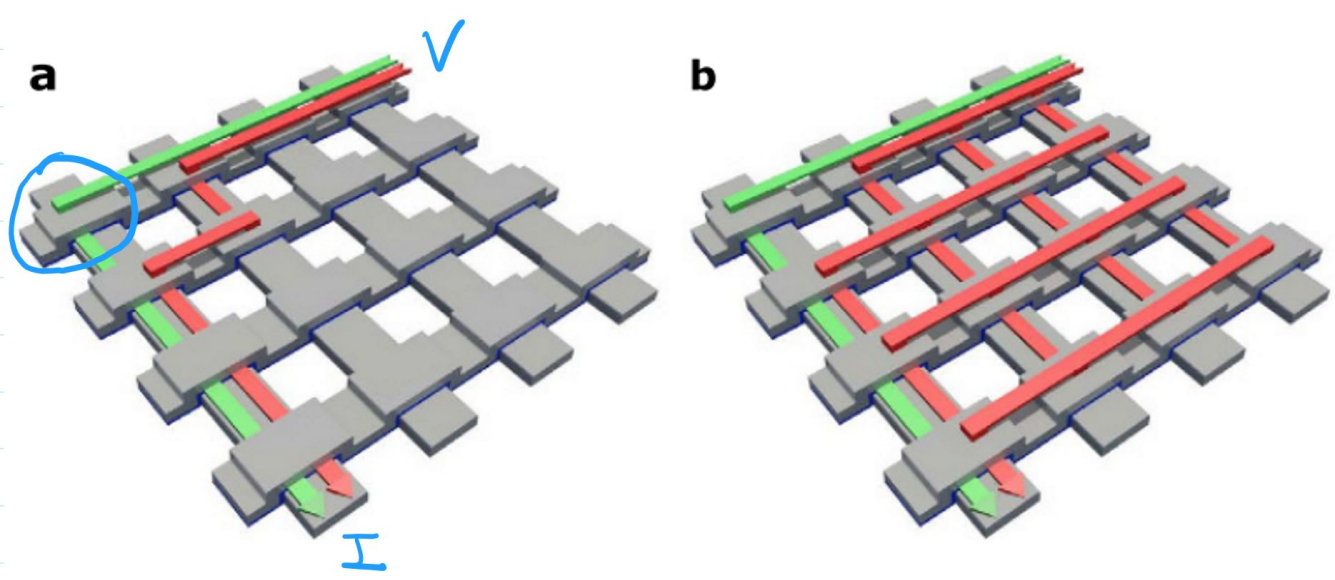
- Memristors as RAMs = RRAMs
  - Combat current memory bottleneck

- Higher speed and density than SRAM or DRAM
- Non-volatile = good for embedded memories
- Can be directly integrated onto processor

State of the art (Classical computing)

- Problem: sneak current & wire resistance
- Cause inaccuracy and waste energy

memristor-crossbar architecture:



Gray:
conductors
Cross:
memristor

State of the Art - benefits

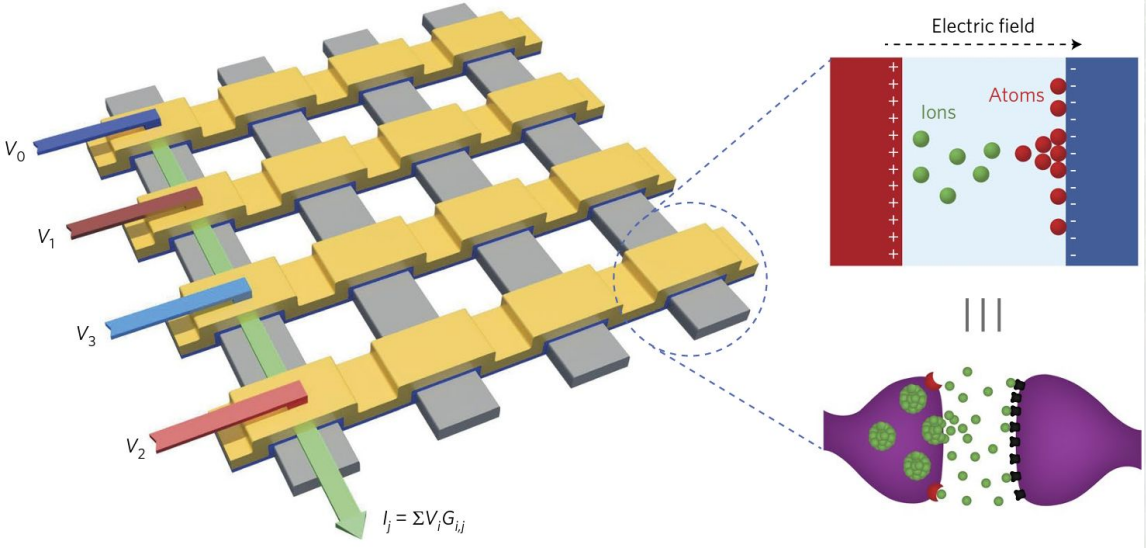
- Don’t need rigorous device endurance, since memristive elements have years of memory storage
- Can process raw signals directly without the need to convert to digital values
  - Reduces energy, latency, and chip area in this application

State of the art - neural networking

- Memory and compute are collocated
- This is ideal for implementing neural networks
- Direct parallel to neurons connected by weighted synaptic connections in real brains
- Overcomes von-neumann bottleneck associated with moving weights between memory and the computer

State of the art - neural networking

- Very efficient reads and writes
- One read is equivalent to a N x (N x M) matrix multiplication
- On traditional computer this would require N x M multiply-accumulate operations

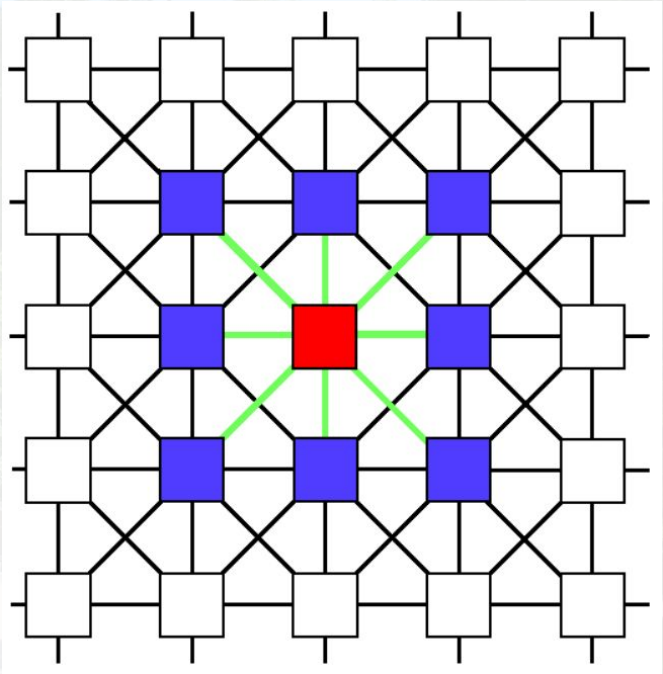


State of the Art - SNNs

- Useful for implementing Spiking Neural Networks (SNNs)
- SNNs try to mimic how synapses in the brain gain potential that at some point “spikes” the gets reset
- This allows us to process spatio-temporal data (real time sensory data)

State of the Art - Cellular

- Memristors are also useful for implementing cellular neural/nonlinear networks
- In these networks, synapses only connect to nearest neighbors, so changes propagate over time
- Useful for image processing and pattern recognition



State of the Art - Boltzmann

- Restricted Boltzmann Machines rely on learning the probability distribution of the input data (stochastic neural network)
- Computation speed is heavily dominated by fetching weight values
- One architecture using memristors solved combinatorial optimization problems with 50x increased performance and 25x lower energy than a single-threaded multi-core system

State of the Art - Application

- Pattern recognition and dot-product engines have been made
- How an engineer may utilize a 32x32 memristor array

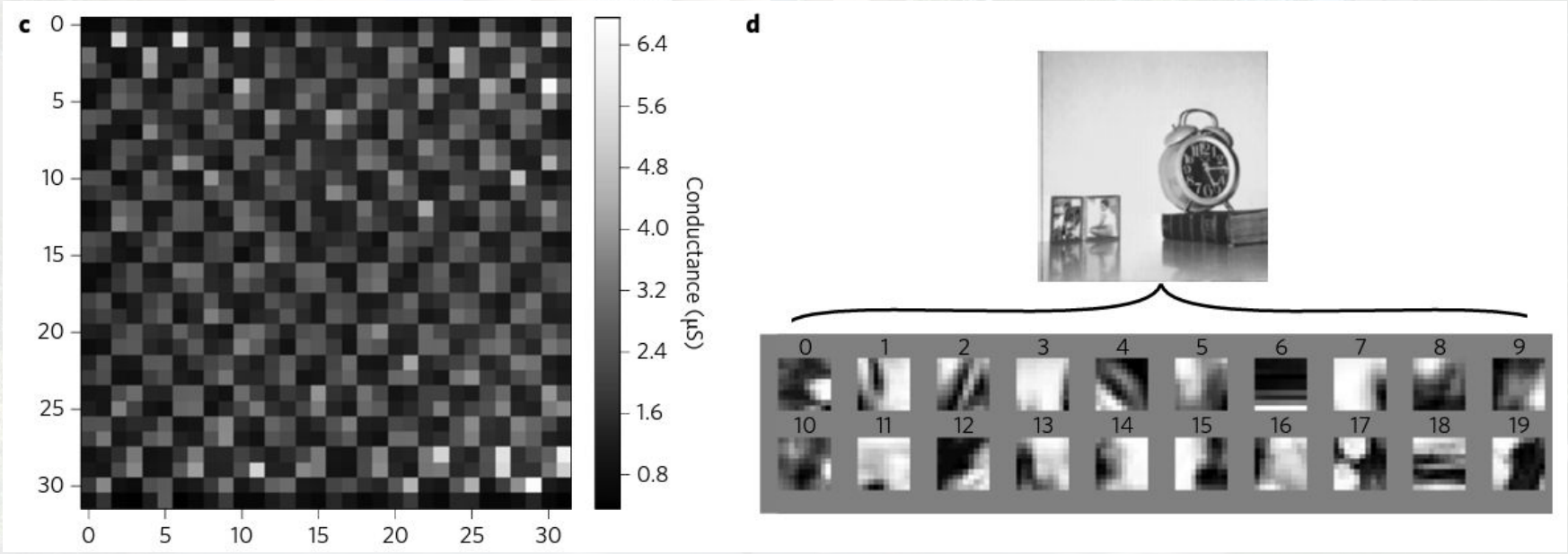


Image: “Sparse coding with Memristor networks”

State of the Art - Computation

- The crossbar structure can significantly speed up other computations using the same optimizations:
  - vector arithmetic operations
  - linear algebra
- Improves energy efficiency in data congestive systems

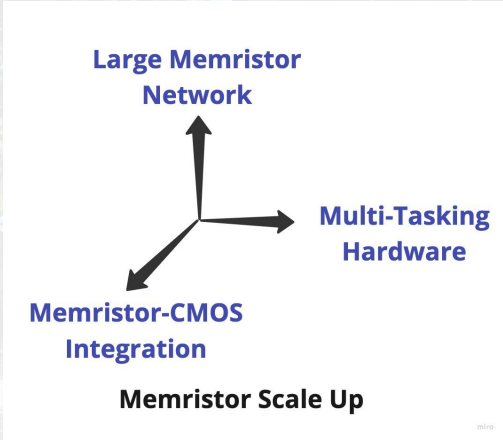
Scaling up and scaling down

Scaling Up

Prior academic studies on memristor research focused on proving concepts rather practical implementation.

Real world implementation requires **Scale-Up** in 3 axis

1. Increasing the size of functional memristor network.
2. Multi-Tasking Hardware System
3. Reliable Memristor-CMOS Integration.

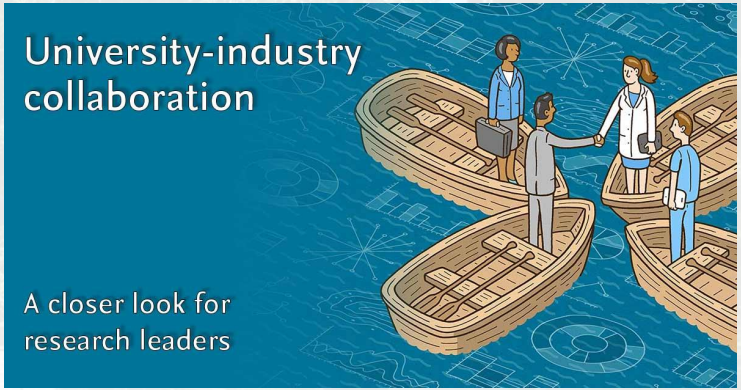


1. Increasing the size of functional memristor network.

- A practical memory or computing system requires billions of functional memristors.

Requirement

- Improving the yield of memristor device fabrication.
- Close collaboration of university researchers with industry partners.
- Improving the scalability of the hardware.



2. Multi Tasking Hardware System

- Same hardware can be used for different function like Neural Network, Arithmetic Operations, Data Storage depending on task and data structure.
- Dynamically reconfigurable for different workload in runtime through software without hardware modification.

Challenges

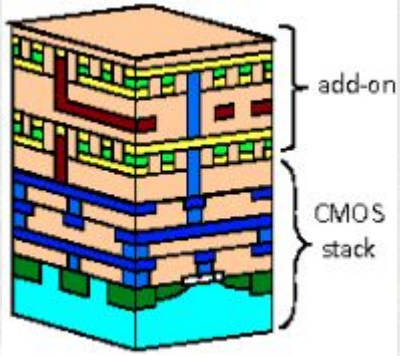
- Arithmetic Operation requires tight device distribution compared to others.
- Long device endurance cycle needed for logic implementation.

3. Reliable Memristor-CMOS Integration

- CMOS circuitry to provide the necessary interface and control operations.
- But Chip-Level Integration through silicon vias doesn't provide required bandwidth.

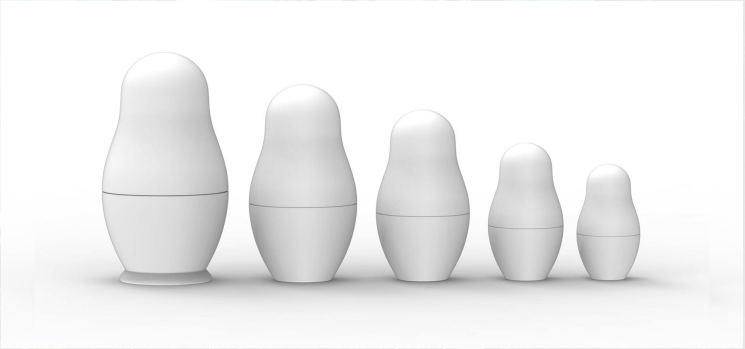
Solution

- Monolithic integration of memristor arrays directly on top of CMOS circuitry with very-high-density local interconnects.
- Cost effective with few additional masks.
- 3 D multi layered memristor integration are fabricated layer by layer in stacked fashion which significantly increases density.



Scale Down

- Internal ion redistribution in response to external stimulation drives memristor.



What does **Scale Down** mean?

- Doesn’t only mean reducing device size but ability to precisely control device operation at atomic level(single atom level).
- This precise control gives high functional density and optimal device performance.
- Atomic process address low current trade-offs and provide device stability.

Scale In

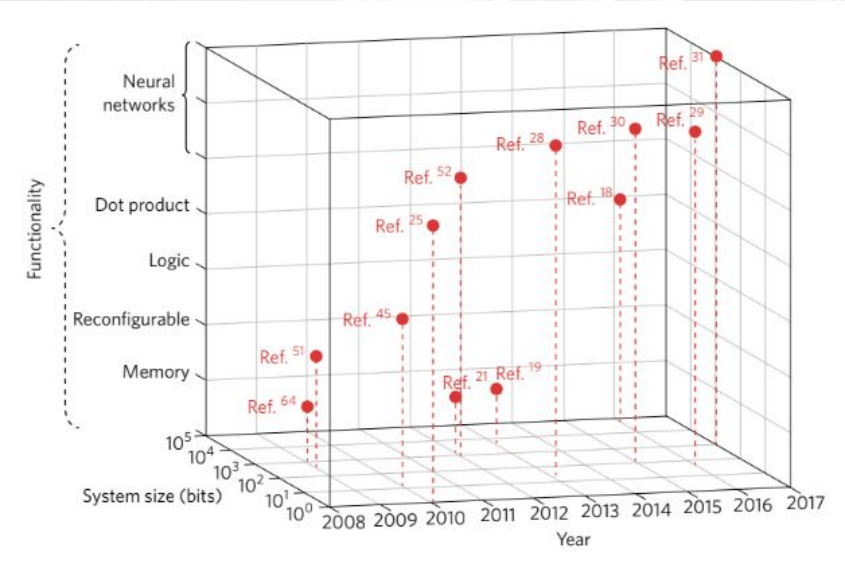
- Increasing the density of information available for storage and processing without area increase.
- Tuning the conductance range of storage memristor by exploiting state variables.
- Single Memristor = 5 to 64 Conductance Levels

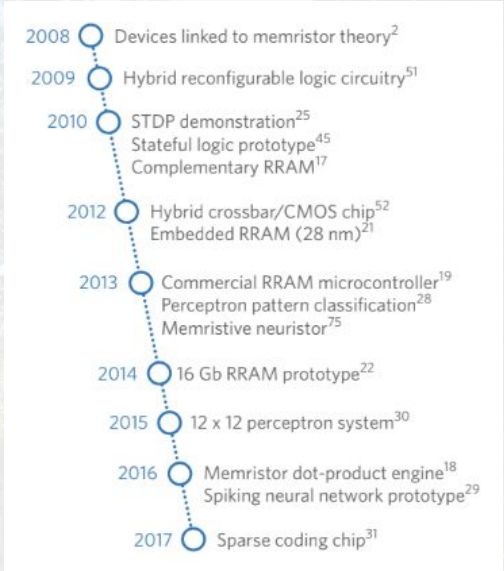
State Variables

- Density of free electrons, hopping sites, radius of the filamentary metallic channel, width of the tunnel barrier

Limitation: Repeatedly control state variables and range of each state variable.

Scaling up and scaling down





Growth of memristor hardware system functionality and size.

The role of chemistry and biological details

Bio-Inspired Computing

- Evolution -> Genetic Algorithms
- Neuron Networks -> Artificial Neural Networks
- **Neuron Chemistry -> ??**

Bio-Inspired Computing

- Evolution -> Genetic Algorithms
- Neuron Networks -> Artificial Neural Networks
- **Neuron Chemistry -> ??**

Key Challenges:

- Hard and expensive to implement at scale
- Unknown benefits of bio-inspired systems

Bio-Inspired Computing

- Evolution -> Genetic Algorithms
- Neuron Networks -> Artificial Neural Networks
- **Neuron Chemistry -> Memristors**

Key Challenges:
- Hard and expensive to implement at scale
- Unknown benefits of bio-inspired systems

Use bio-realist devices? → Memristors

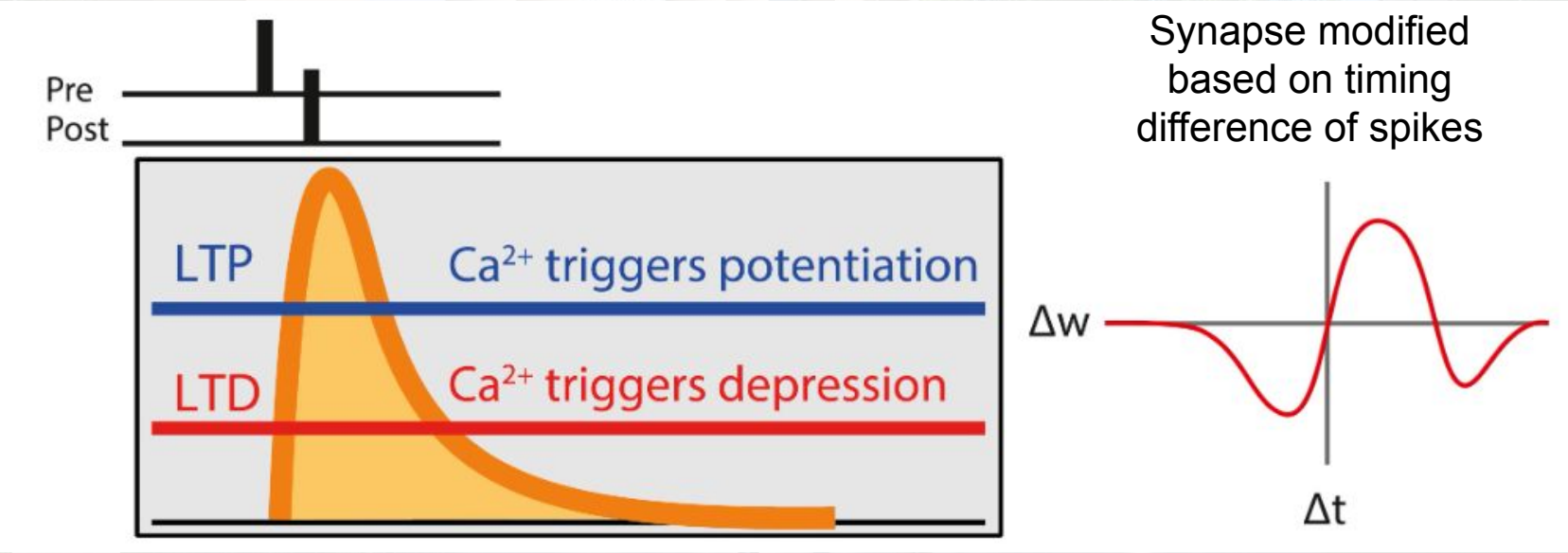
Bio-Realistic Properties of Memristors

Example: Calcium Effect in neurons

- Spike-timing–dependent plasticity (STDP) is a form of synaptic modification thought to constitute a mechanism underlying formation of new memories
- The polarity of synaptic changes -> function (relative timing between pre- and postsynaptic activity)

Bio-Realistic Properties of Memristors

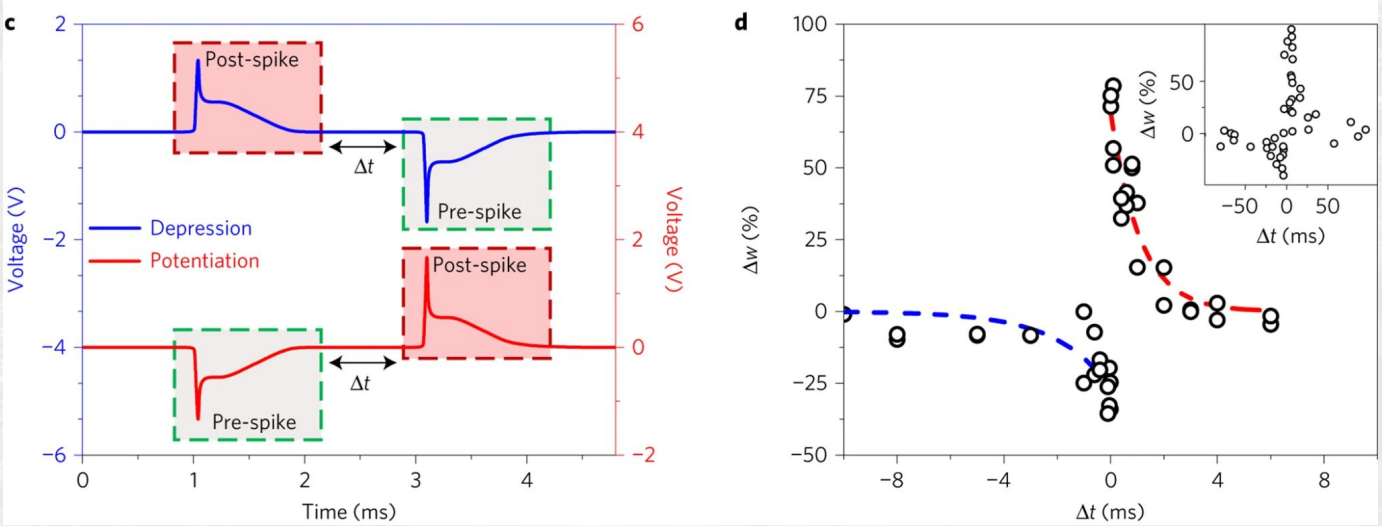
Example: Calcium Effect in neurons



Synaptic plasticity rules with physiological calcium levels https://doi.org/10.1073/pnas.2013663117

Bio-Realistic Properties of Memristors

Example: Calcium effect mimicked by second-order memristor devices



Memristors can help realize Spiking Neural Networks!

Wang, Z., Joshi, S., Savel'ev, S. et al. Memristors with diffusive dynamics as synaptic emulators for neuromorphic computing. Nature Mater 16, 101–108 (2017)

Bio-Realistic Properties of Memristors

Example: Neuronal Active Gain

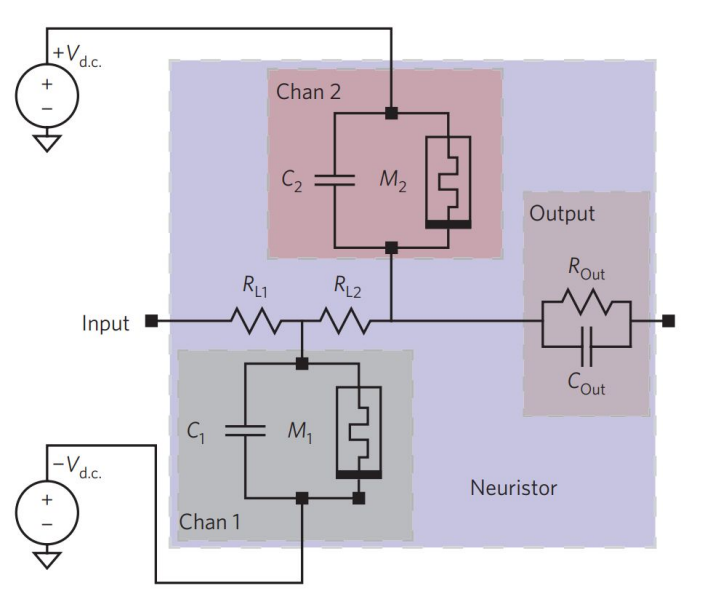
- Dynamic Gain of Neuron
- Small Signals + Appropriate conditions => High Amplification

Bio-Realistic Properties of Memristors

Example: Neuronal Active Gain -> Local Device Temperature

- Memristor materials such as VO_2 or NbO_2 have negative temperature coefficient
- This leads to positive feedback loop because of the Negative Differential Resistance (NDR)
- Positive feedback -> Small input signal generating large response

Neuristor



The channels consist of:

- Mott memristors (M1 and M2)
- Characteristic parallel capacitance (C1 and C2, respectively)
- Biased with opposite polarity d.c. voltage sources

Models

- Signal Gain
- Spiking timing dependent weights

Device Challenges and Possible Solutions

Device Challenges and Possible Solutions

- While promising, memristors still have material and device challenges
  - These challenges are often application specific

- High performance memory applications (DRAM Replacement)
  - lower programming current/voltage
  - minimize sneak current
  - minimal device-device and cycle-cycle variability

Device Challenges and Possible Solutions

- Online training of neural networks via back propagation
  - programming bit precision
  - asymmetry in ON vs. OFF switching
  - selectively optimizing the ON/OFF point can lead to fully stuck ON/OFF devices, leading to large errors

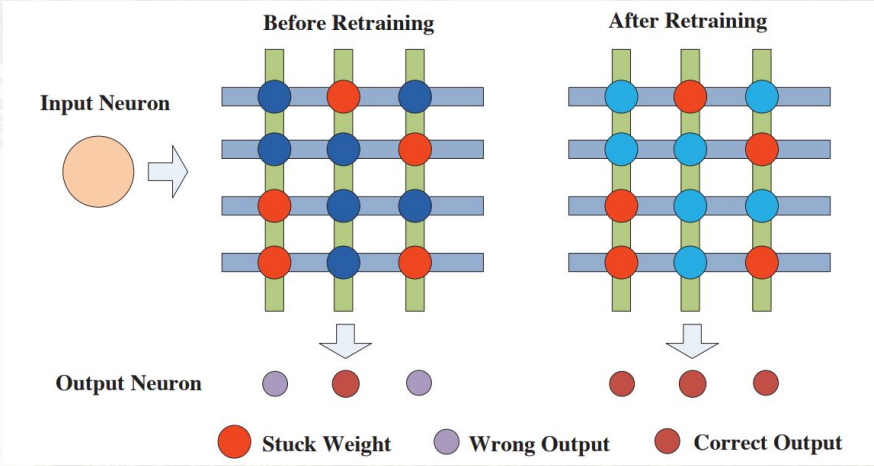
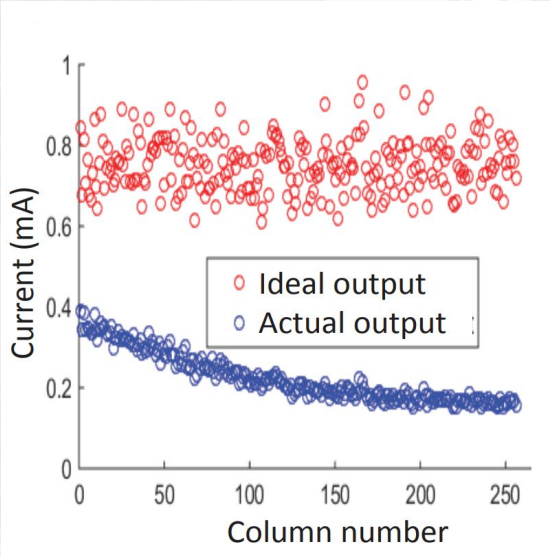


Image: “ISAAC: a convolutional neural network accelerator with in-situ analog arithmetic in crossbars."

Device Challenges and Possible Solutions



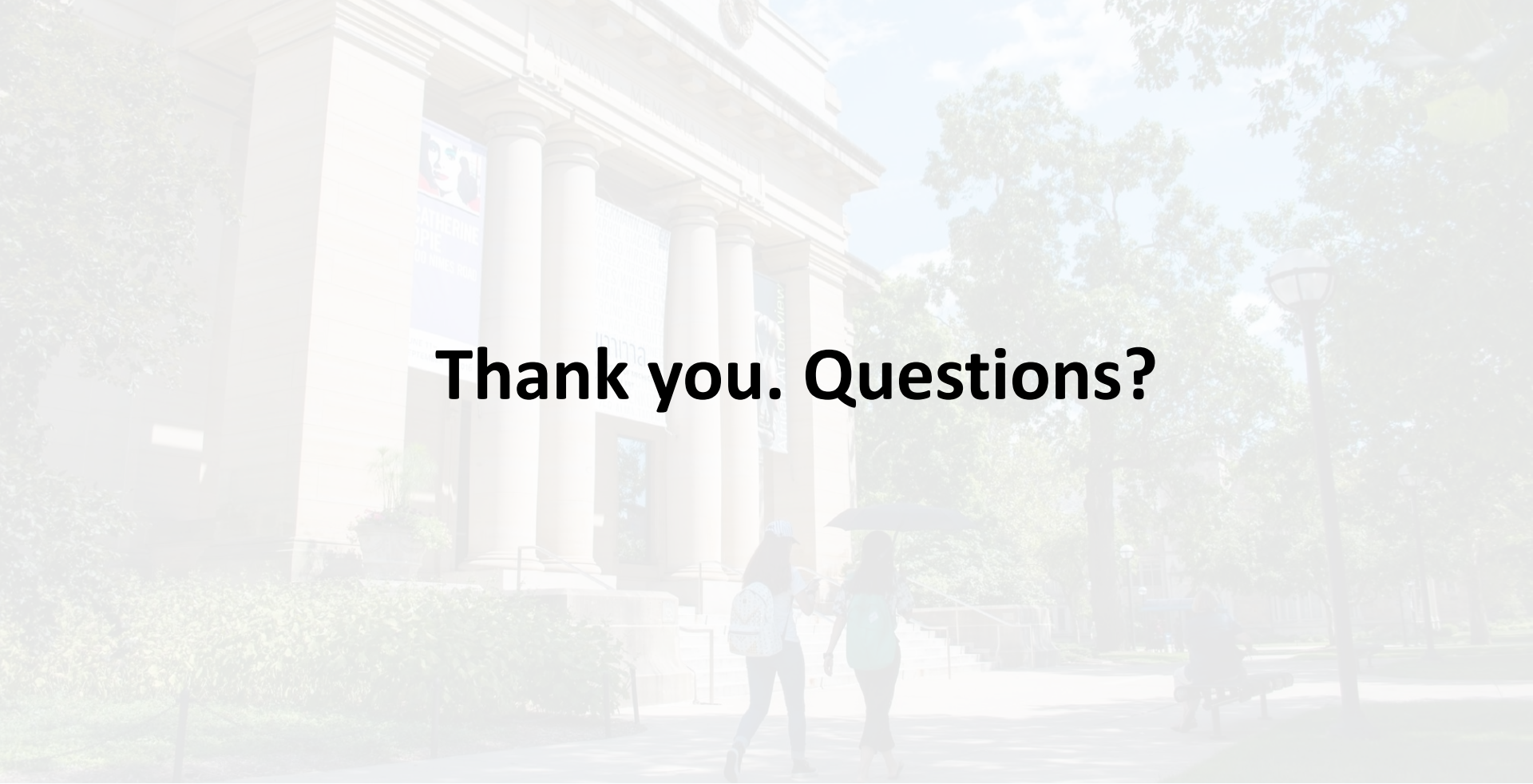
Naive mapping leads to poor computing accuracy in real crossbars. This figure shows an example of all voltages across devices in a 256×256 crossbar array, positive matrix values are linearly mapped to memristor conductance.

- Offline trained, inference-only neural network accelerators
  - requires high accuracy of computed matrix operations
  - computational errors from unideal memristors and wire resistance
  - a strong model that accounts for non-idealities and wire resistance, compensating algorithm can eliminate these effects

Image: “Dot-product engine for neuromorphic computing: programming 1T1M crossbar to accelerate matrix-vector multiplication."

Conclusions

- Memristors have great potential to push computing systems
  - Short term: high density, on chip, non-volatile memory improve performance and can find applications in any computing task
  - Further Advances: large-scale implementation of memristor-based neuromorphic computing systems
  - Long term: memristor-based general-purpose, in-memory computing platform



Thank you. Questions?

Appendix: SNN Training

- A common learning rule is spike-timing-dependant-plasticity
- Strength changes depending on causality of synapse firing
- Resulting values are long-term potentiation and long-term depression
- The more recently a connection is made, the stronger it is