Resistive Memory for Neuromorphic Algorithm Acceleration

NICE Workshop

Albuquerque, NM

February 25, 2015

Matthew J. Marinella, Sapan Agarwal, David R. Hughart, Patrick R. Mickel, Alex Hsia, Steve Plimpton, Timothy J. Draelos, Seth Decker, Roger Apodaca, J. Bradley Aimone, Conrad R. James

*matthew.marinella@sandia.gov

Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

Outline

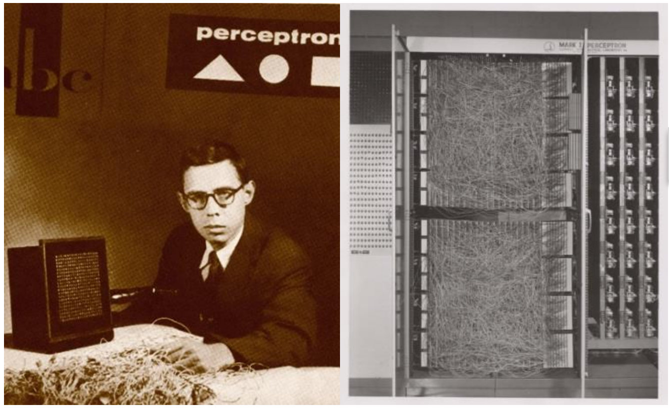
- **Motivation**
- **Neural Core Design and Architecture**
- **Resistive Memory Device Assessment**
- **Summary and Future Work**

Brain Inspired Computing at Sandia

- Sandia established a multidisciplinary LDRD project in neuromorphic computing project in October 2015
  - Hardware Acceleration of Adaptive Neural Algorithms (HAANA)
  - Algorithms research
  - Hardware and Device Architectures
  - Resistive Memory Devices
- Some of the initial work on learning architecture and resistive devices covered in this talk

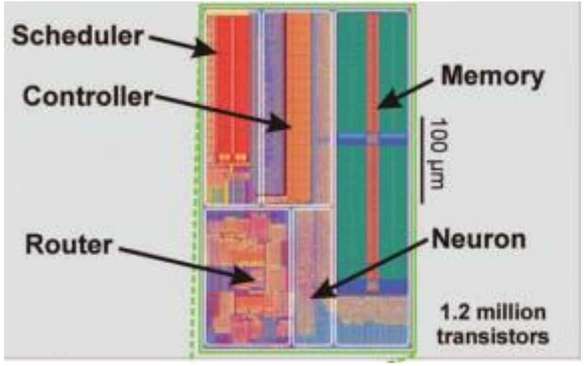
Neural-inspired Computing Hardware

Mark I Perceptron (Rosenblatt 1960):

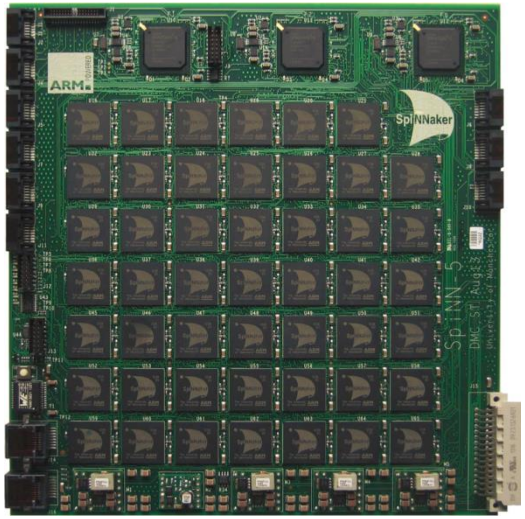


Arvin Calspan Advanced Technology Center; Hecht-Nielsen, R. *Neurocomputing* (Reading, Mass.: Addison-Wesley, 1990); Cornell Library;

DARPA , IBM TrueNorth (2014):



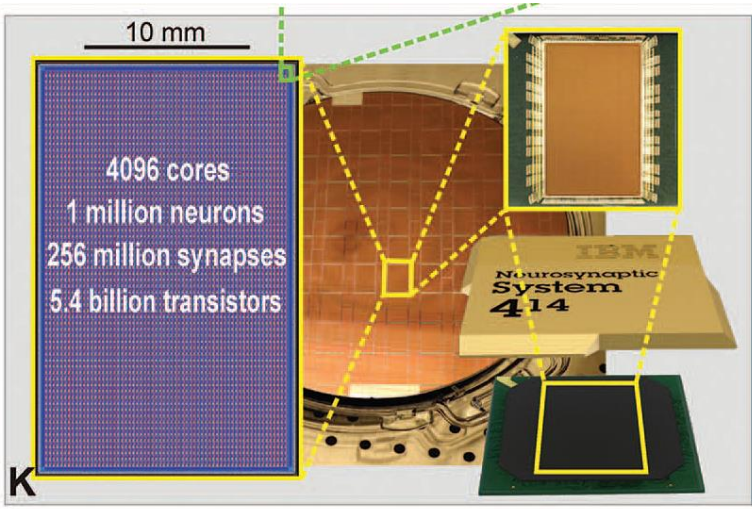
EU HBP, SpiNNaker (2014):



?

Current State of the Art in Neural Algorithm Computing

- CPU/GPU
  - Most general; common programming languages
  - Lowest power efficiency and performance
  - Memory separate from chip
  - Example: Google deep learning study (CPU→GPU)
- FPGA
  - General; requires hardware design language
  - Moderate performance and efficiency
- Custom IC (Truenorth, Spinnaker)
  - Specific: ex. executes STDP
  - Highest performance and efficiency
  - Expensive, 40MB local memory
  - Example: IBM Truenorth
- Moore law has provided enormous benefits to all of these technologies

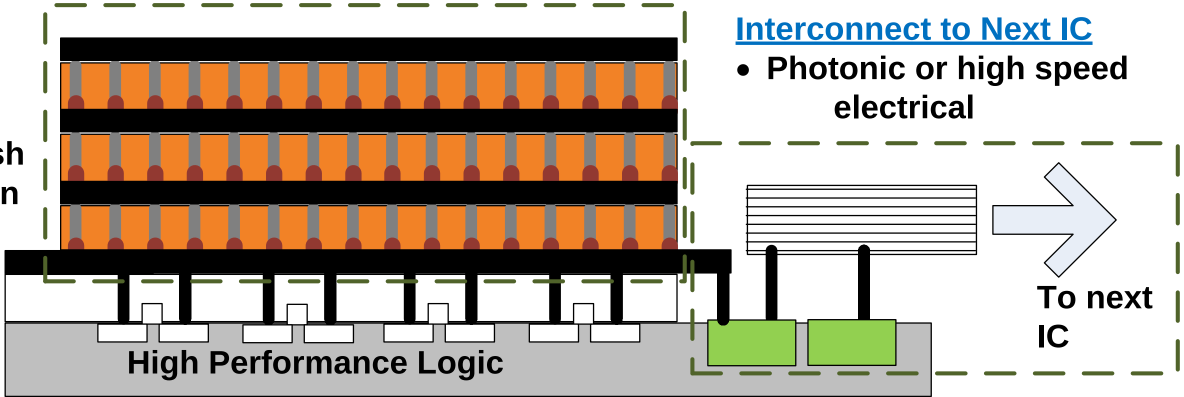


Next Generation of Neural Algorithm Computing

- Problem: neural algorithm training requires significant memory and logic interaction
- What is the most efficient way to combine memory, logic and interconnects?
- SRAM: on chip cache memory is limited to ~40MB digital (Intel E7)
  - ns latency, max regardless of CPU, GPU or ASIC
- Off chip communication to DRAM costs >100 pJ/op, ~10ns latency
- Resistive memory on chip: can be stacked to >TB/cm^2, >100 layers
  - On chip access, <pJ per op and <1ns latency possible
  - Terabit densities on single chip – on chip wiring is low energy!
  - Sub 1V switching – minimal CV^{2}f loss (DRAM 2-5V)
- Significant power savings using a ReRAM based HW accelerator
  - Example: Taha found 16x reduction in power, 6x improvement in perf per chip over SRAM

ReRAM Layers:

- Terabit cm^{-2} per layer
- Replaces DRAM & flash
- <1 pJ, <10 ns operation



Interconnect to Next IC

- Photonic or high speed electrical

Outline

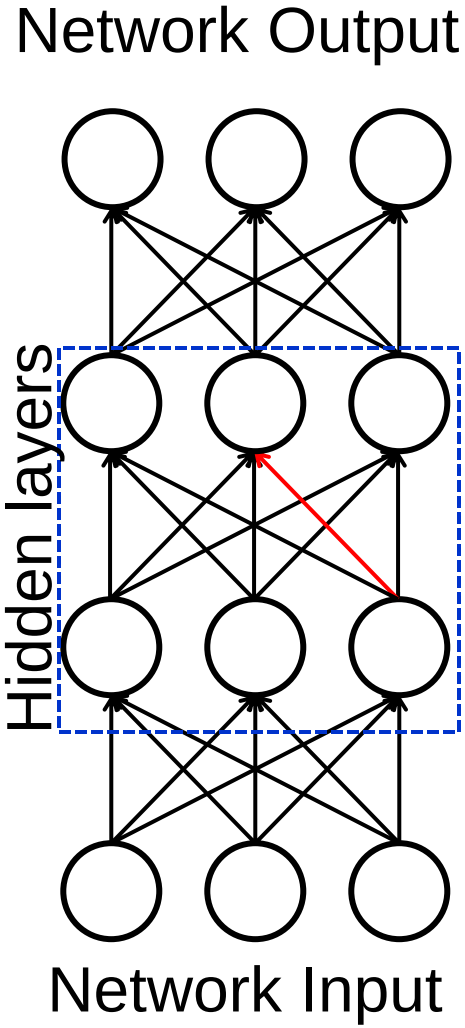
- Motivation
- Neural Core Design and Architecture
- Resistive Memory Device Assessment
- Summary and Future Work

Backpropagation of Errors

- **Straw-man algorithm – others can be used**
- **Initial linear network training did not have a method of training hidden layers**
- **Backpropagation of errors gave a method to train hidden network layers**

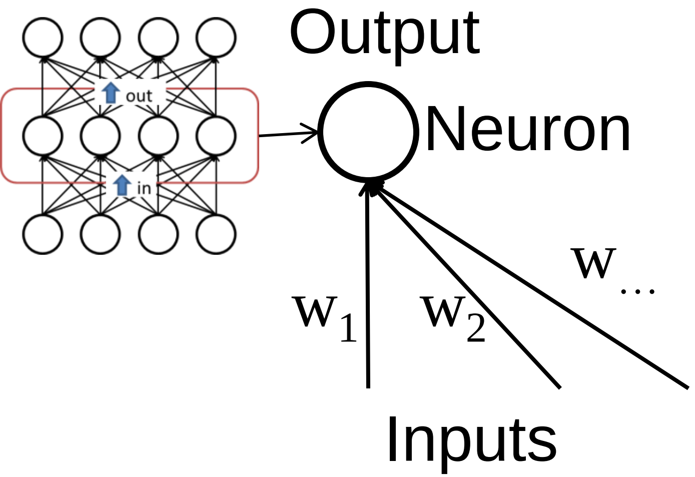
How does backpropagation work?

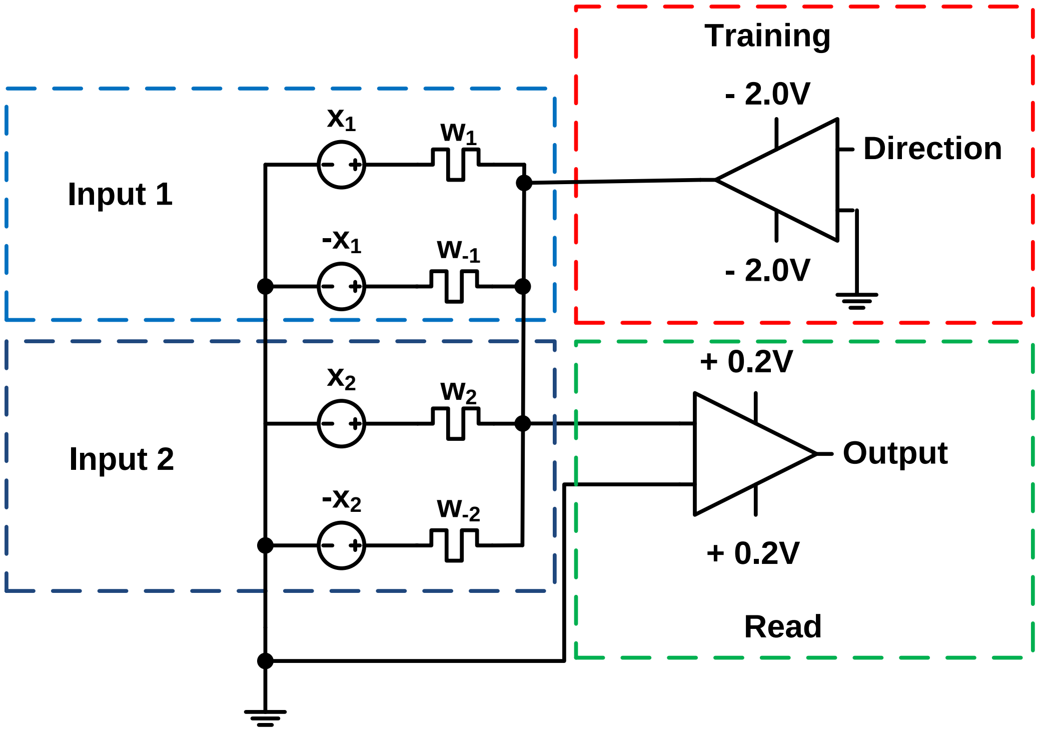
1. **Forward calculation of output**
2. **Calculate all weight deltas and compute error derivative (backprop errors)**
3. **Combine this with learning rate to update weight**



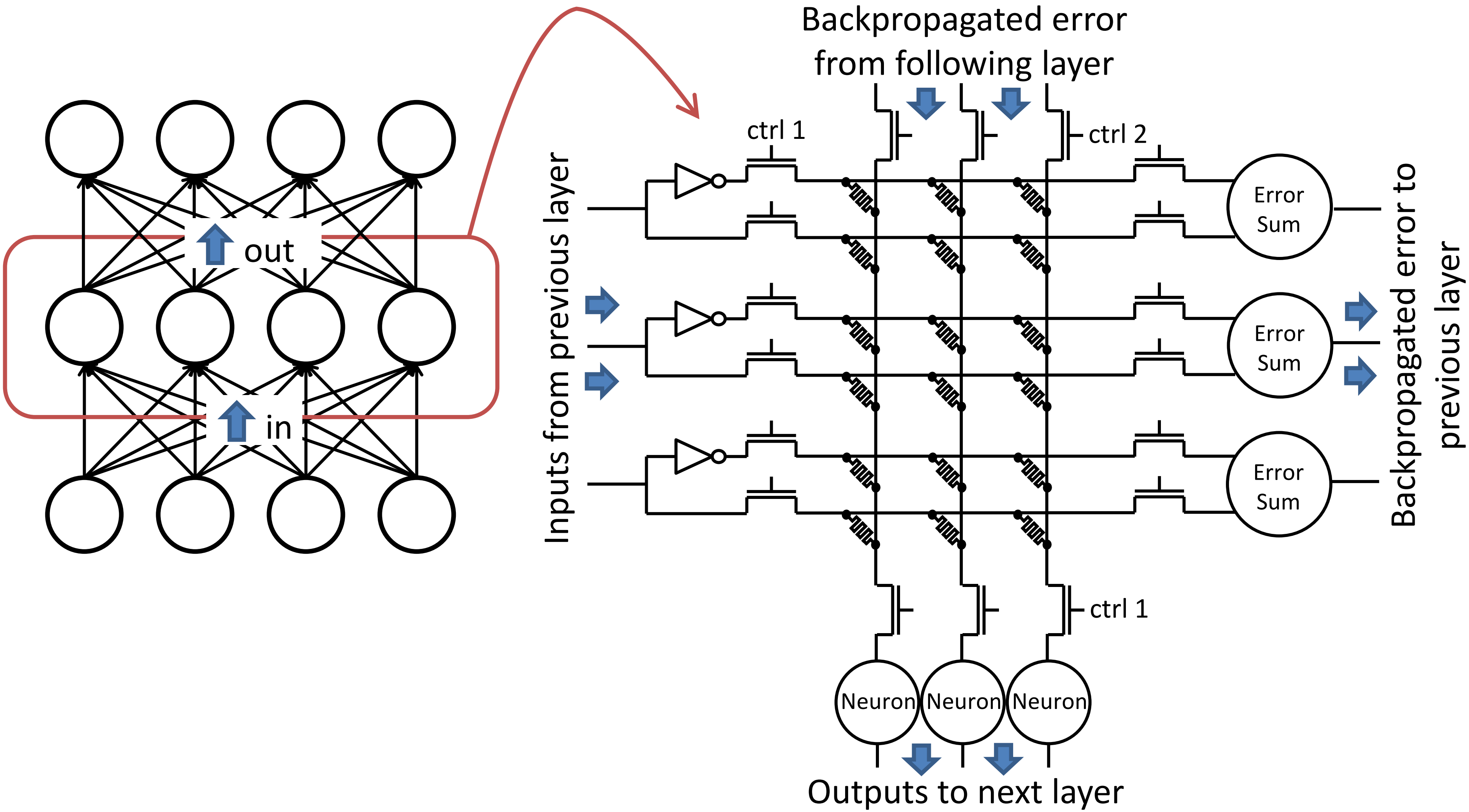
Mapping weight matrix to ReRAM

- **Single ReRAM based neuron**

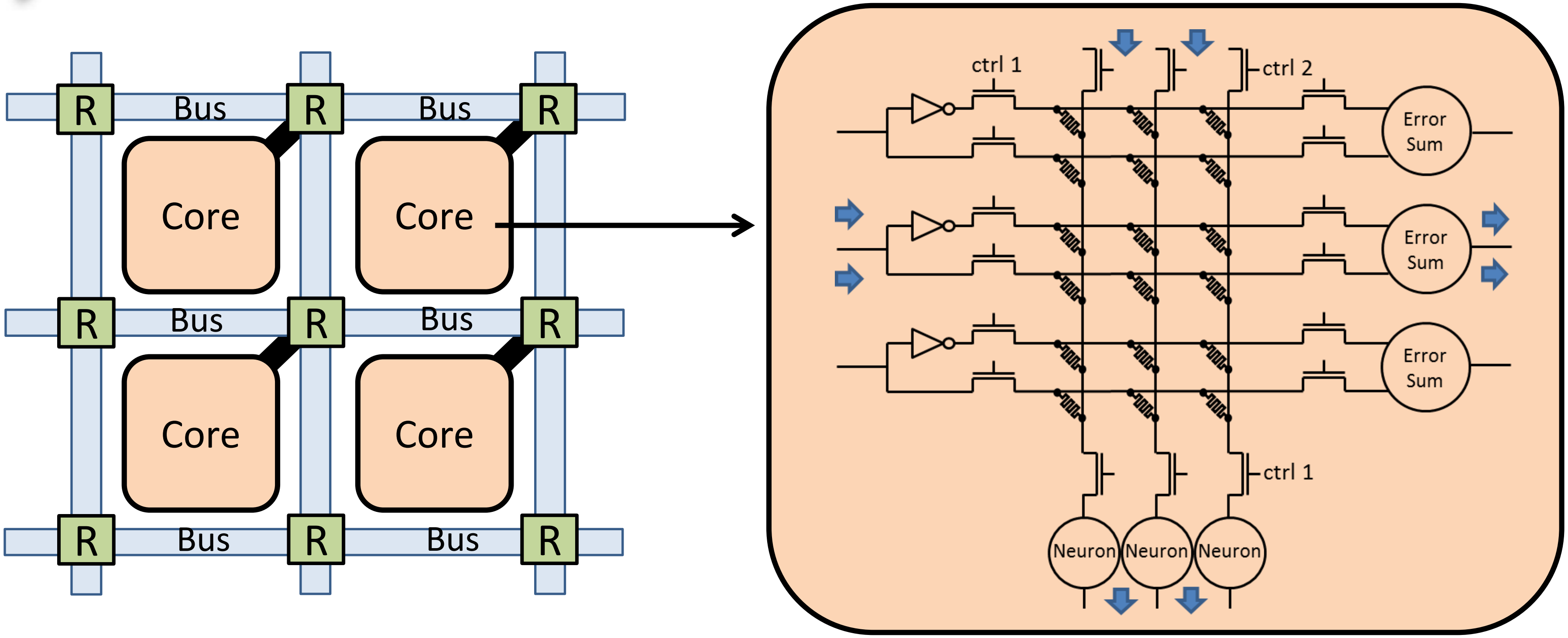




Backpropagation Circuit



System Level Architecture



Neuromorphic core:

- Stores and updates weights
- Computes weighted sum of inputs
- Computes error derivatives
- Applies non-linear neuron function

Inputs:

- Previous layers' neuron outputs
- Following layers' error derivatives
- Control signals

Outputs:

- Neuron outputs to following layer
- Error derivatives to prev. layer

Inter-Core Connectivity Options

Must consider options for interconnecting multiple neural cores:

Maximize Flexibility:

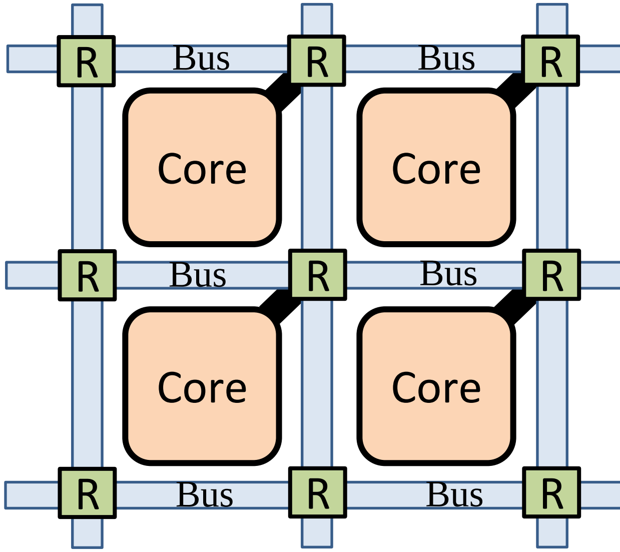
- **All neuron inputs/outputs go off chip, training signals come in**
- **Least energy efficient, most complex wiring**

Maximize Energy Efficiency:

- **Connections hardwired, perhaps some flexibility through FPGA architecture**
- **Least flexible design**
- **Each core independently does learning based on local inputs and outputs**

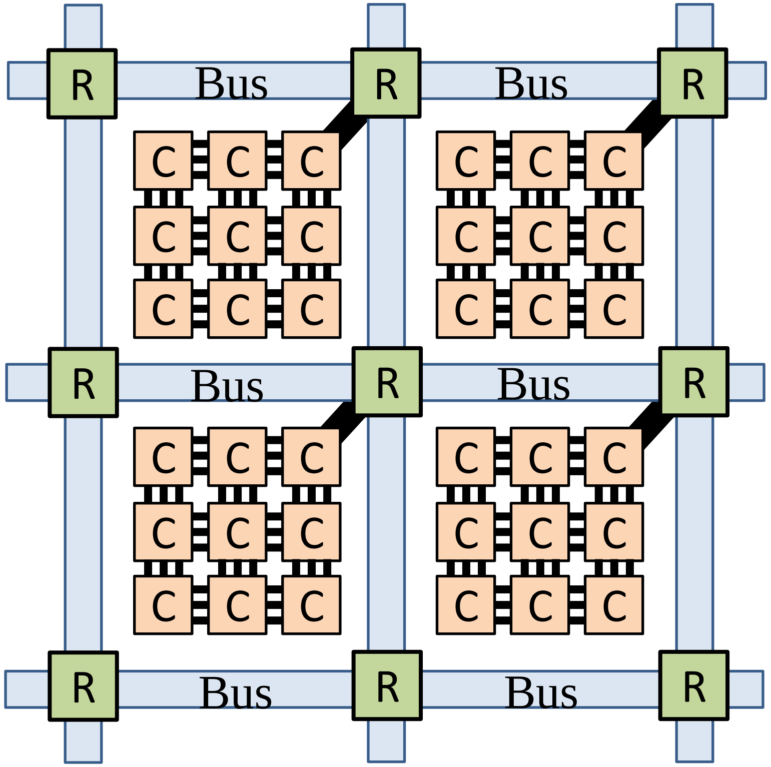
Hybrid Approach Also Possible

- **Use local buses with some off-chip capability**



Configuring Crossbars

- The size of a single ReRAM crossbar will be limited by analog signal to noise ratio and interconnect line resistance and max current considerations
- Weights in a network will most likely not correspond to the number of weights in one crossbar
- Solution: multiple crossbars per core
  - Analog signal enhancement between separate cores
- Maximize efficiency:
  - Share a single A/D converter between cores in different layers
  - Use digital communications for long distance communications

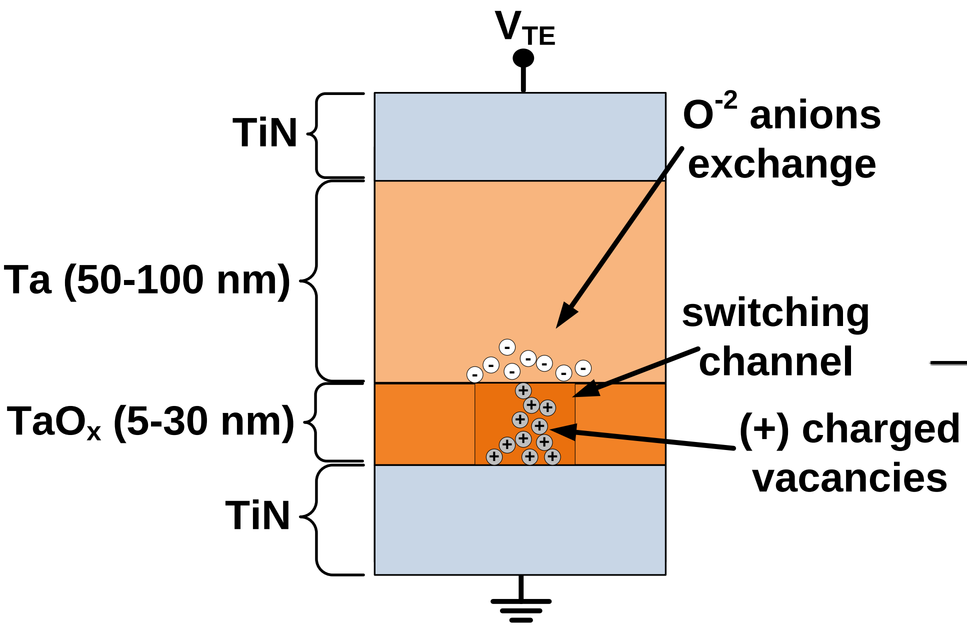


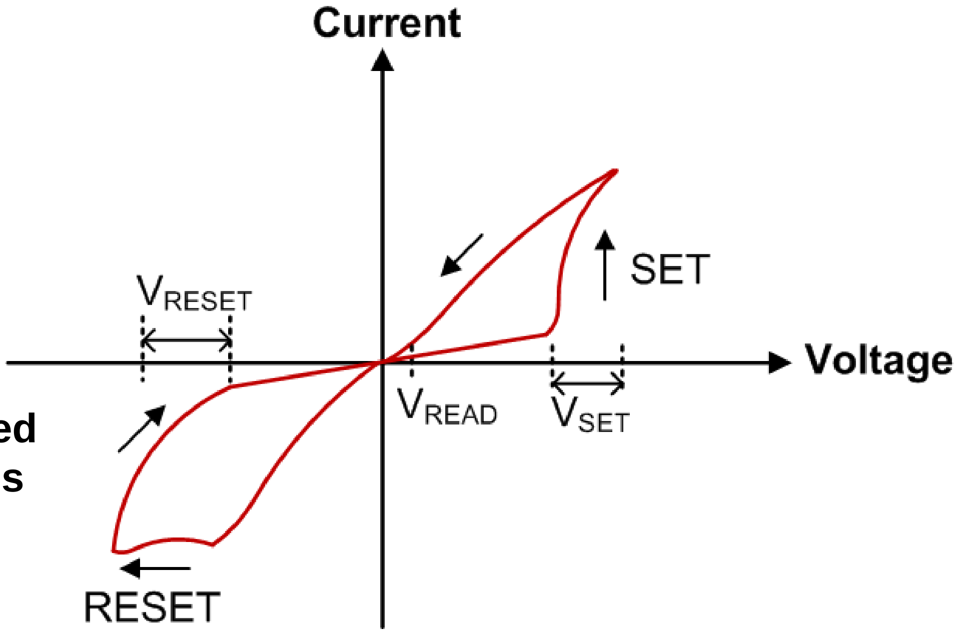
Outline

- Motivation
- Neural Core Design and Architecture
- Resistive Memory Device Assessment
- Summary and Future Work

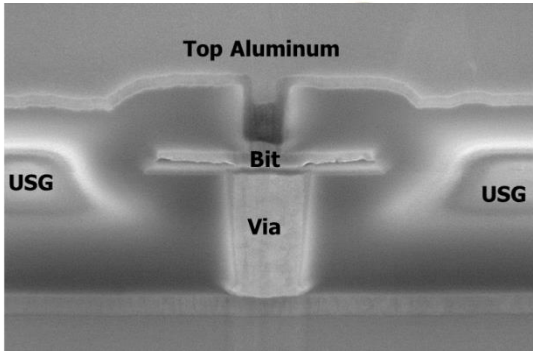
Oxide-Based Resistive Memory

- "Hysteresis loop" is simple method to visualize operation
  - (real operation through positive and negative pulses)
- Resistance Change Effect (polarities depend on device):
  - Positive voltage/electric field: low R – O^{-2} anions leave oxide
  - Negative voltage/electric field: high R – O^{-2} anions return
- Common switching materials: TaO_x, HfO_x, TiO_2, ZnO

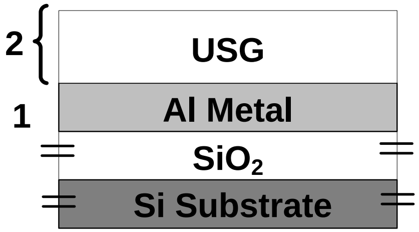




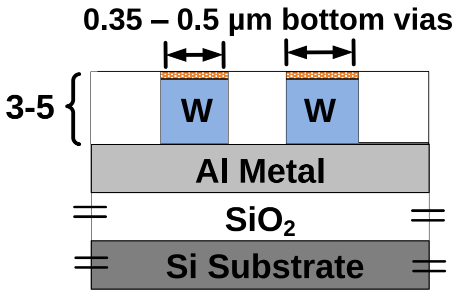
Typical Device Fabrication



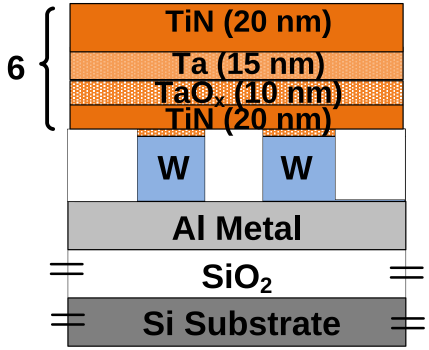
1. Deposit Bottom Metal (Al)
2. Deposit USG



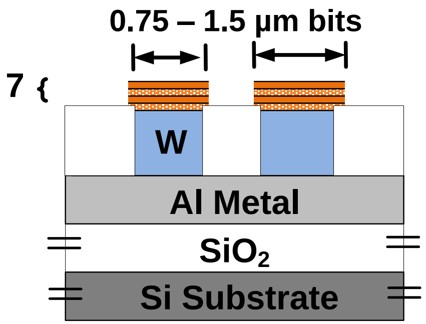
3. Etch via holes in USG
4. Deposit W and TiN layers
5. CMP



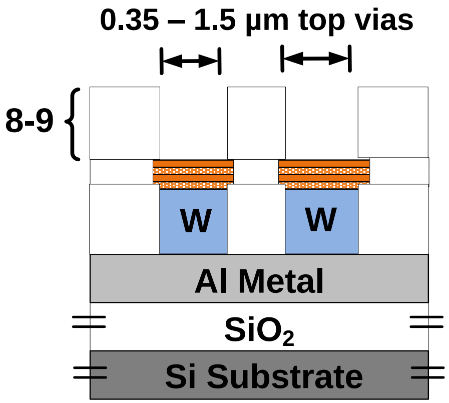
6. Deposit bit stack
(layers enlarged for clarity)



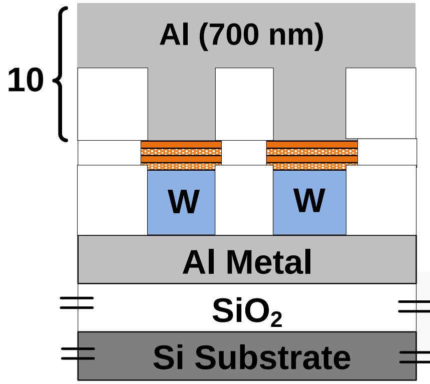
7. Etch bits



8. Deposit top USG
9. Etch top via holes in USG



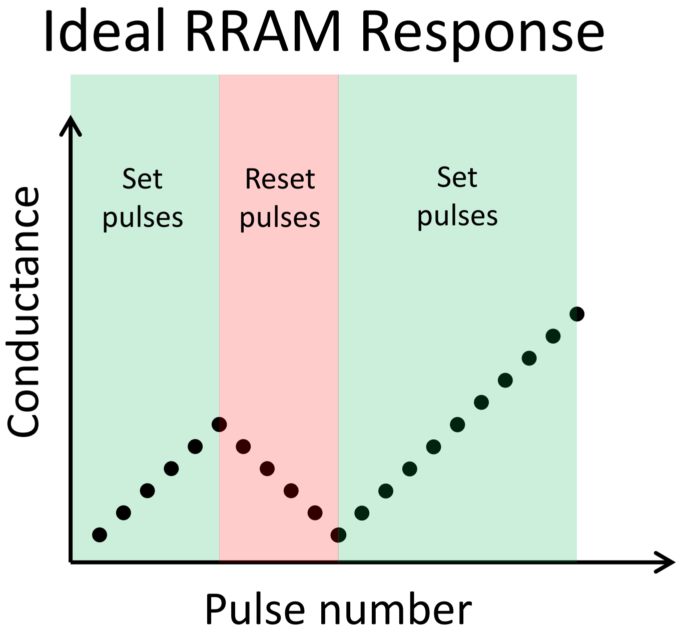
10. Deposit top Al



ReRAM Requirements for a Neural Accelerator

Different requirements than standard memory:

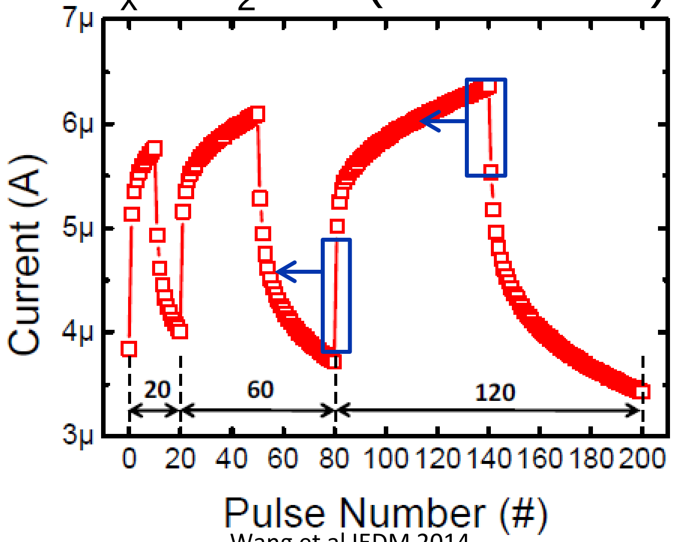
- **Linear dynamic response**
  - **Perhaps some flexibility with this**
- **Low variability**
  - **Device to device, cycle to cycle**
  - **Need correct average behavior**
- **Wide resistance range**
- **Low read noise**
- **Endurance: high**
- **Retention: weeks**



Weight Updates with ReRAM

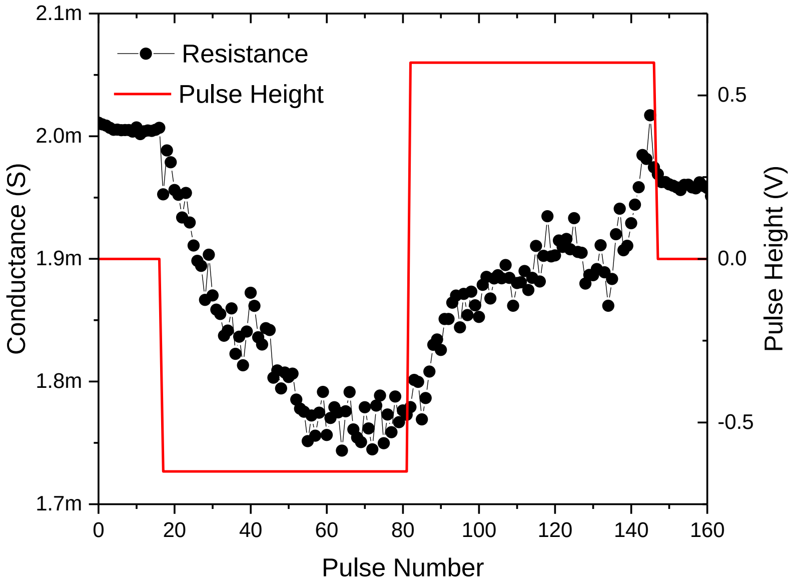
- Square pulse train of fixed amplitude and duration (typ. µs)
- Most common method of characterizing
- Do not want to erase many pulses with opposing pulse

TaO_x/TiO_2 cell (IEDM 2014)



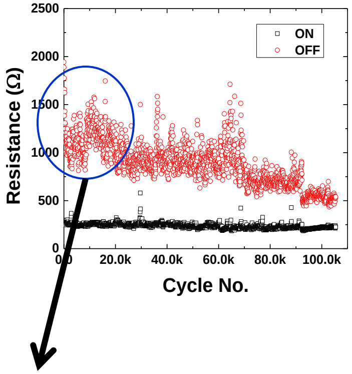
Wang et al IEDM 2014

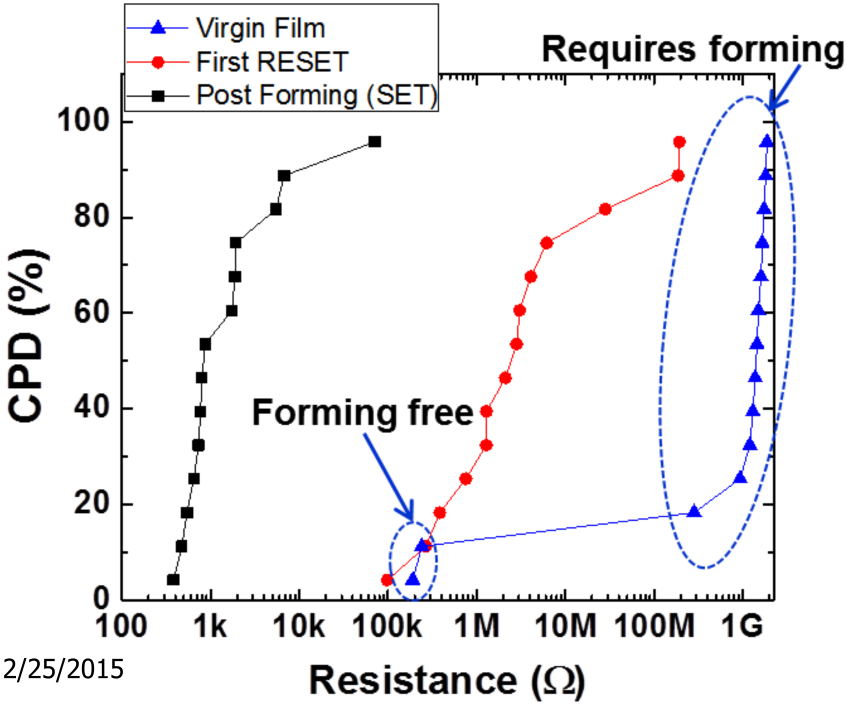
Sandia TiN/TaOx/Ta/TiN ReRAM

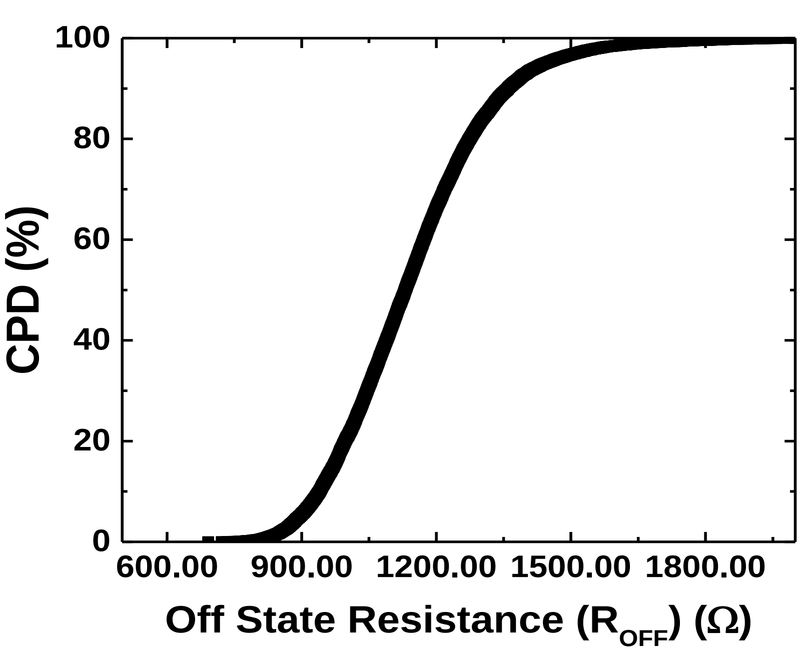


Variability and Noise

- **Interdevice variability**: device to device, can be >10x
  - Variations in film thickness, topography
- **Intradevice variability**: cycle to cycle, can be >2x
  - Fundamental physical attributes
- **Random telegraph noise:**
  - Affects read current, usually least significant

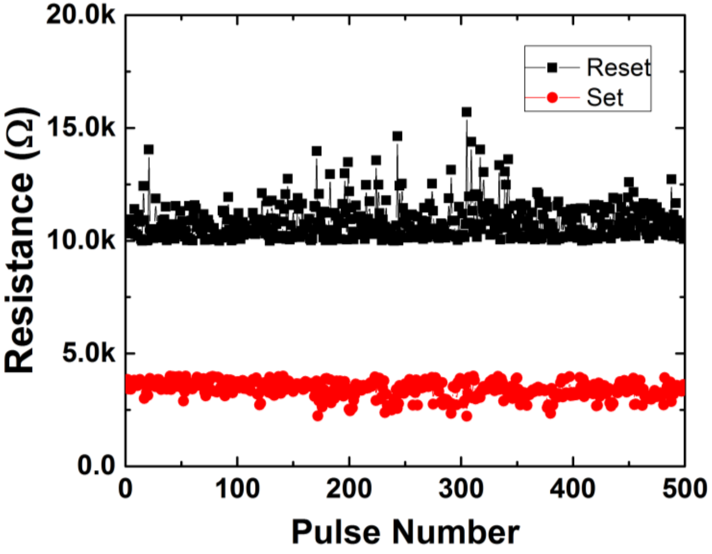


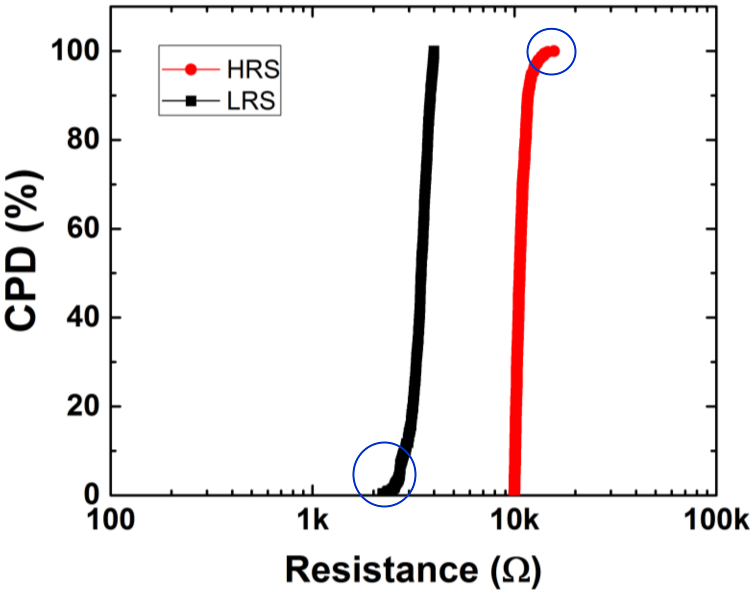


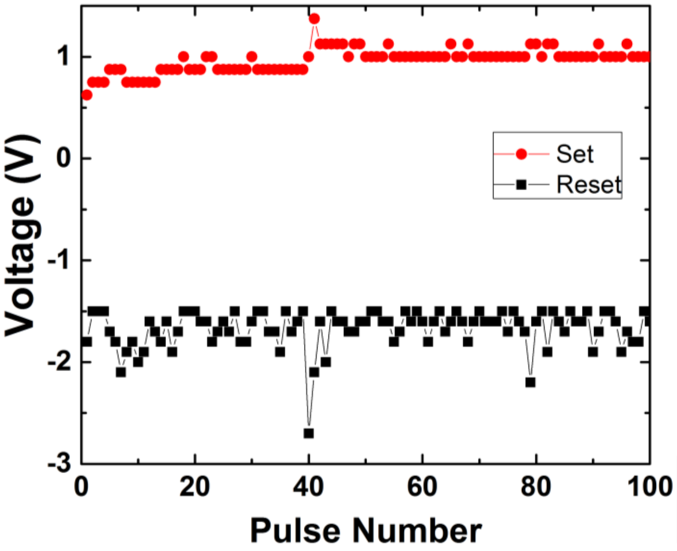


Closed Loop Cycling

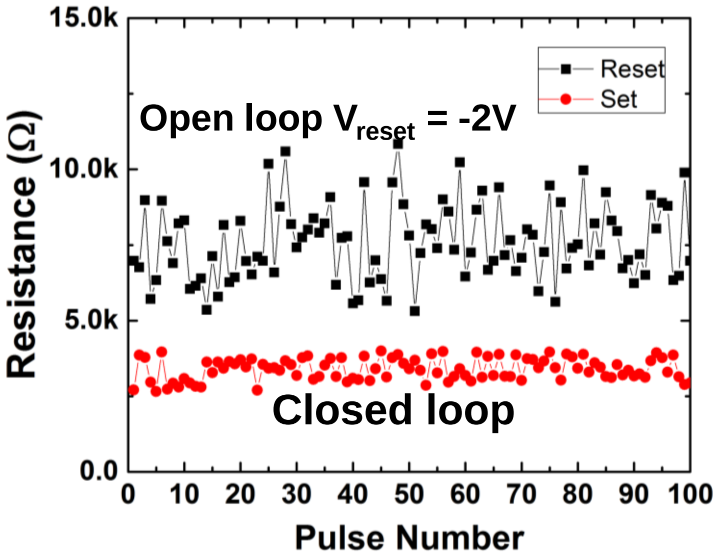
- Resistance "find" routine
  - 400 μs pulses
  - V_{pulse}=0.5→$V_{R\text{-}target}$ (100 mV inc)
- End
  - R>R_{target} (off)
  - R<R_{target} (on)

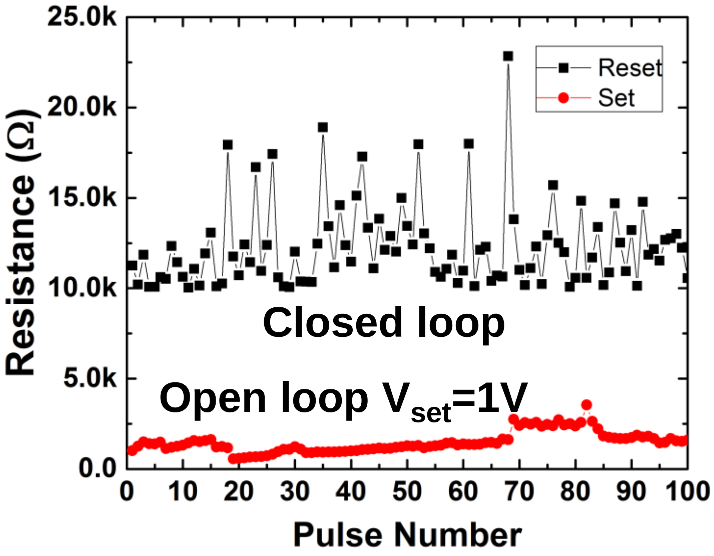


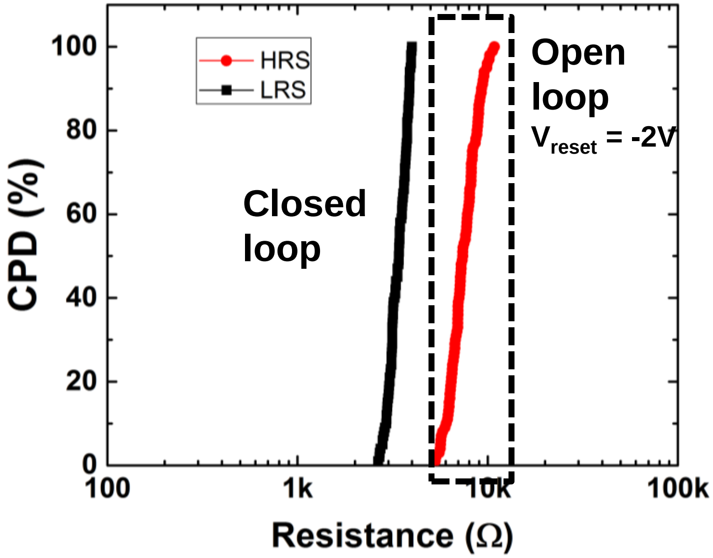


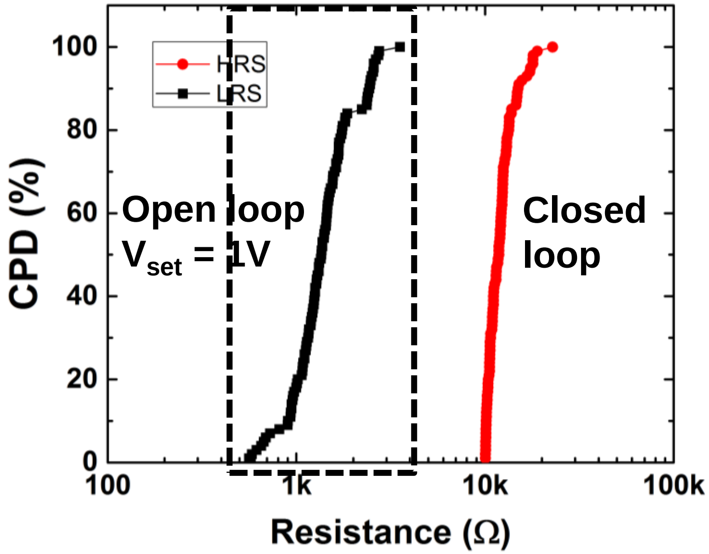


Open/Closed Loop Cycling



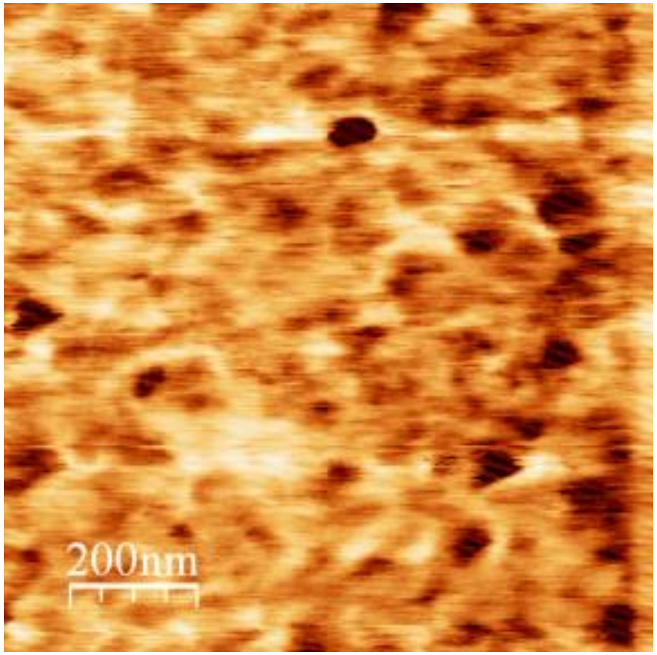


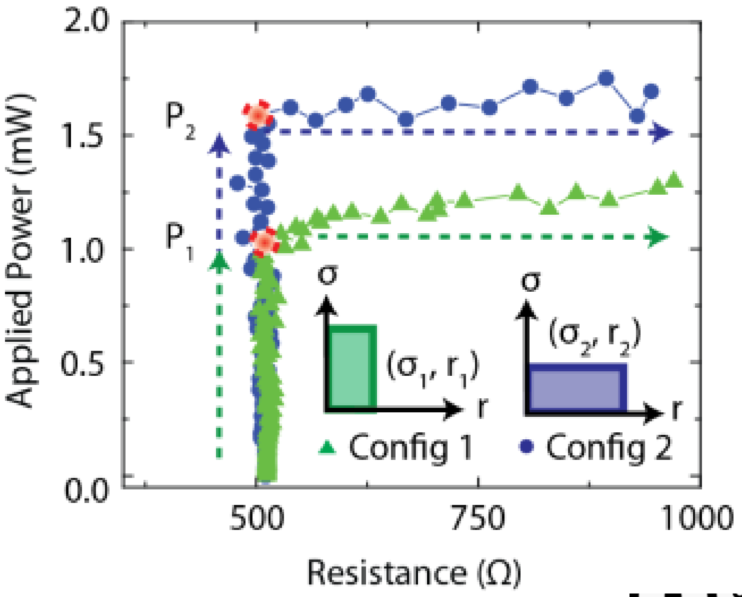




Where does noise originate from?

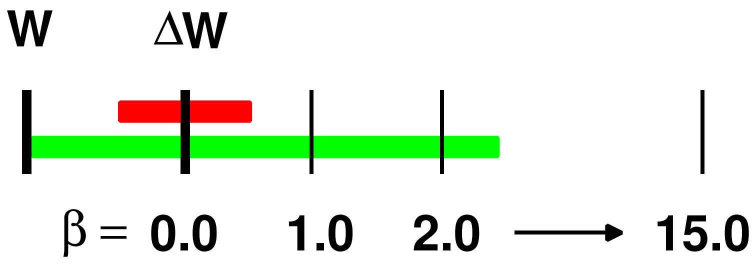
- Important question – not yet a great answer
- Device to device: film thickness variations?
- Cycle to cycle: Internal state variations?





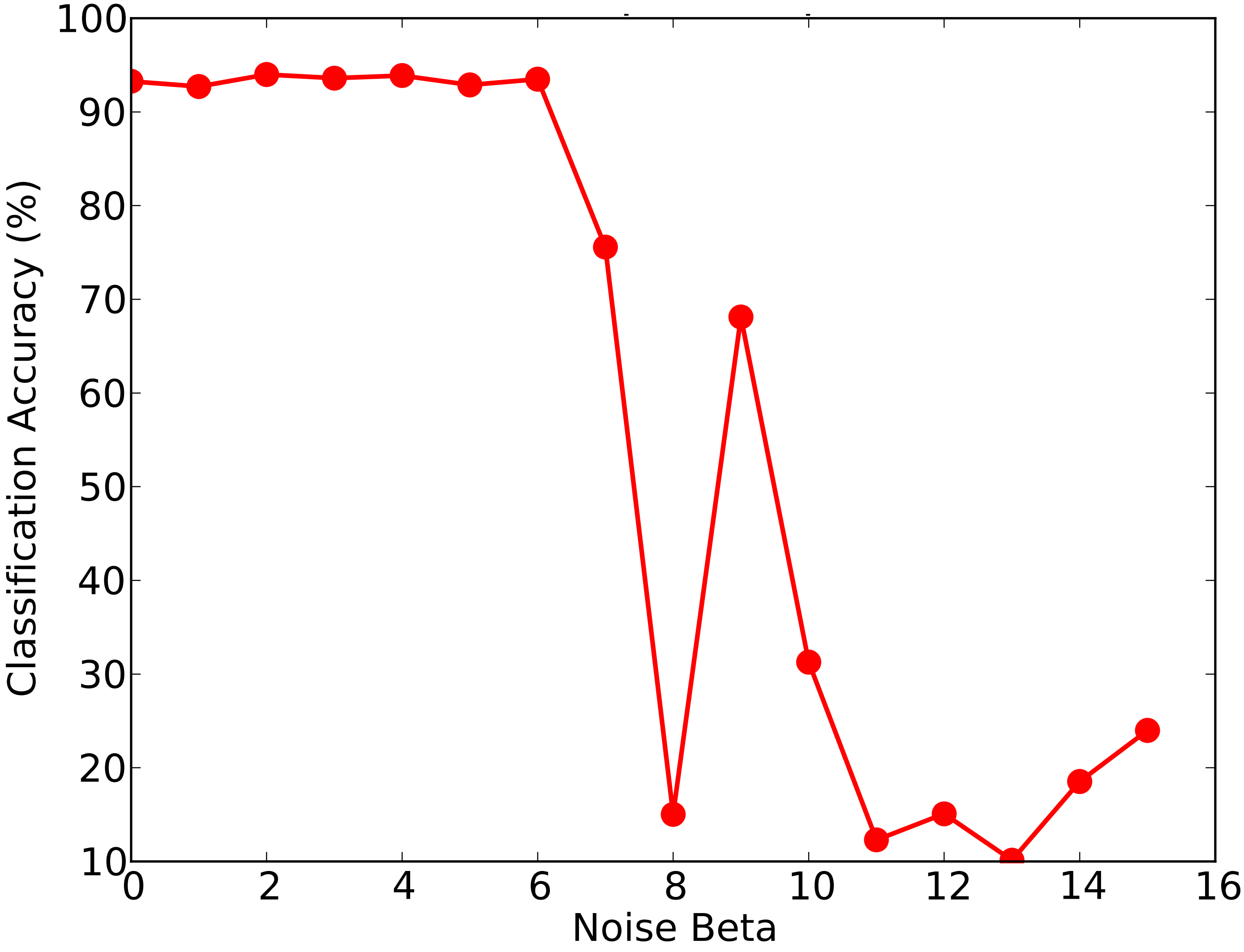
How much noise can be tolerated?

- First step: Simulation Study
  - Mimic on-line learning with incremental open-loop resistance nudges
  - Assume adjustment of each W by ΔW is done imprecisely
  - Add white noise to each weight update:



  - Train on individual images or mini-batch or full epoch
  - Train for fixed epoch count, or to convergence
  - Adjust backprop step-size "α" if needed
  - Classify as usual with final weights

Noise vs Accuracy (Fixed Iteration)



Outline

- Motivation
- Neural Core Design and Architecture
- Resistive Memory Device Assessment
- Summary and Future Work

Summary and Future Work

Summary

- Sandia has launched a new project which includes significant effort in accelerating neural algorithms
- ReRAM will allow TB of memory to be monolithically integrated circuit with high performance logic transistors
  - Saves expensive off-chip (DRAM) operations
  - Lower energy on-chip routing
  - Overcomes density limitations of digital SRAM

Next Steps

- Assess and model variability of ReRAM
- Use this data to create an accurate system model in Xyce
- Finalize system architecture
- Perform power and performance estimates of ReRAM over SRAM accelerator

Acknowledgements

- This project was funded in part by Sandia's Laboratory Directed Research and Development (LDRD) Program
- Collaborators: Stan Williams, Dick Henze, John-Paul Strachan (HP), Jianhua Yang (U Mass), Gennadi Bursuker (Sematech), Tarek Taha, Chris Yakopcic (Dayton), et al.