Analog resistive ~~Memristive~~ neuromorphic hardware

IBM Research – Almaden

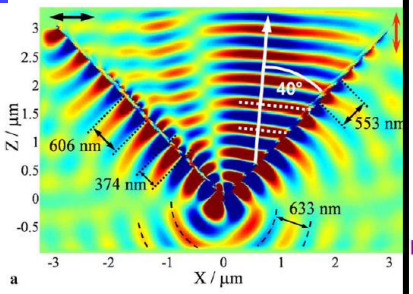
Geoffrey W. Burr

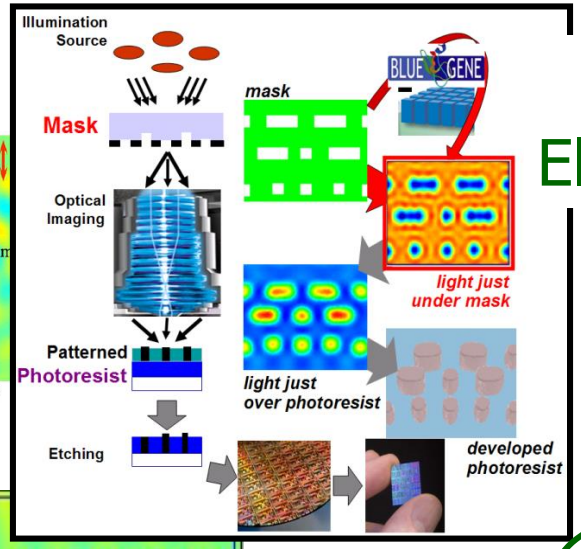
June 30, 2017

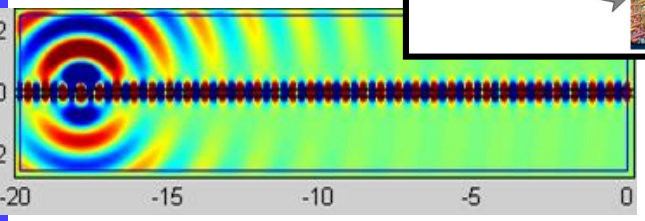
Outline

- **Part I –** Non-volatile memory (NVM) for digital data storage
  - Storage Class Memory
  - Types of NVM devices
  - Importance of 2-terminal selectors
  - Outlook

- **Part II –** Brain-inspired computing: an Industry perspective







Cognitive Computing

Computational Electromagnetics

Principal RSM

Phase-change & Storage class memory

photon echoes

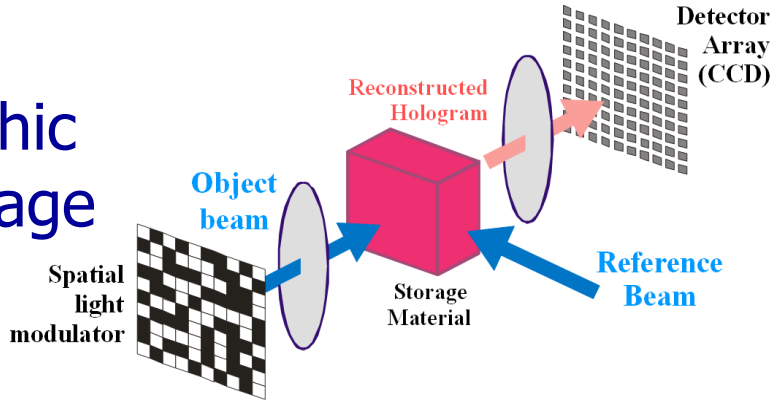
RSM

Joined IBM Almaden

Caltech

Holographic Data Storage

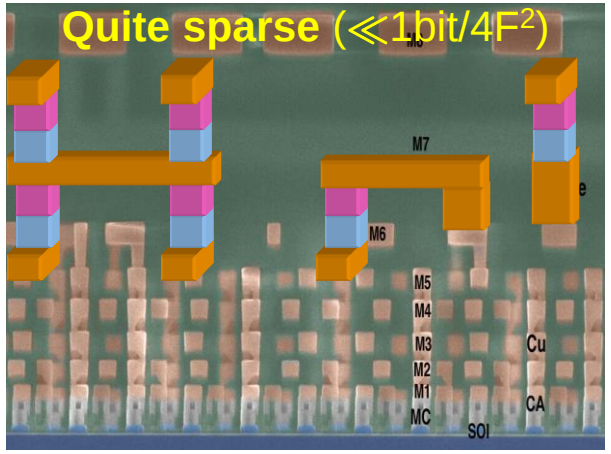
Geoffrey W. Burr

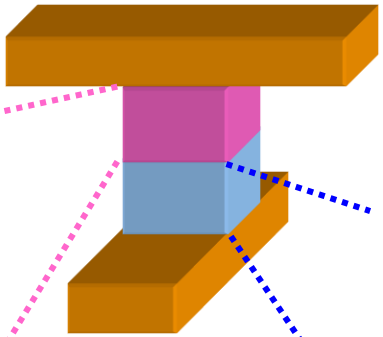


1991 1992 1993 1994 1995 1996 1997 1998 1999 2000 2001 2002 2003 2004 2005 2006 2007 2008 2009 2010 2011 2012 2013 2014 2015

Back-End-Of-the-Line-compatible Non-Volatile Memory:

a fundamental "building block"
enabling a range of applications





Access Device (highly nonlinear or transient switch)

NVM (long-term digital storage)

Programmable e-fuses
(FPGAs, reconfigurable computing)

Embedded storage
(Automotive)

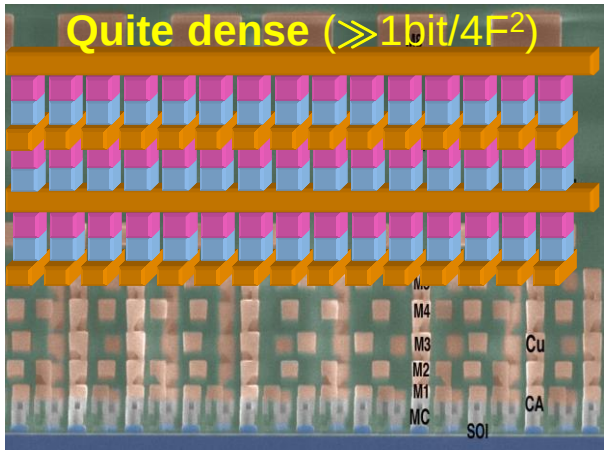
Embedded memory
(Low-power, mobile computing)

Standalone M-class SCM
(Hybrid memory)

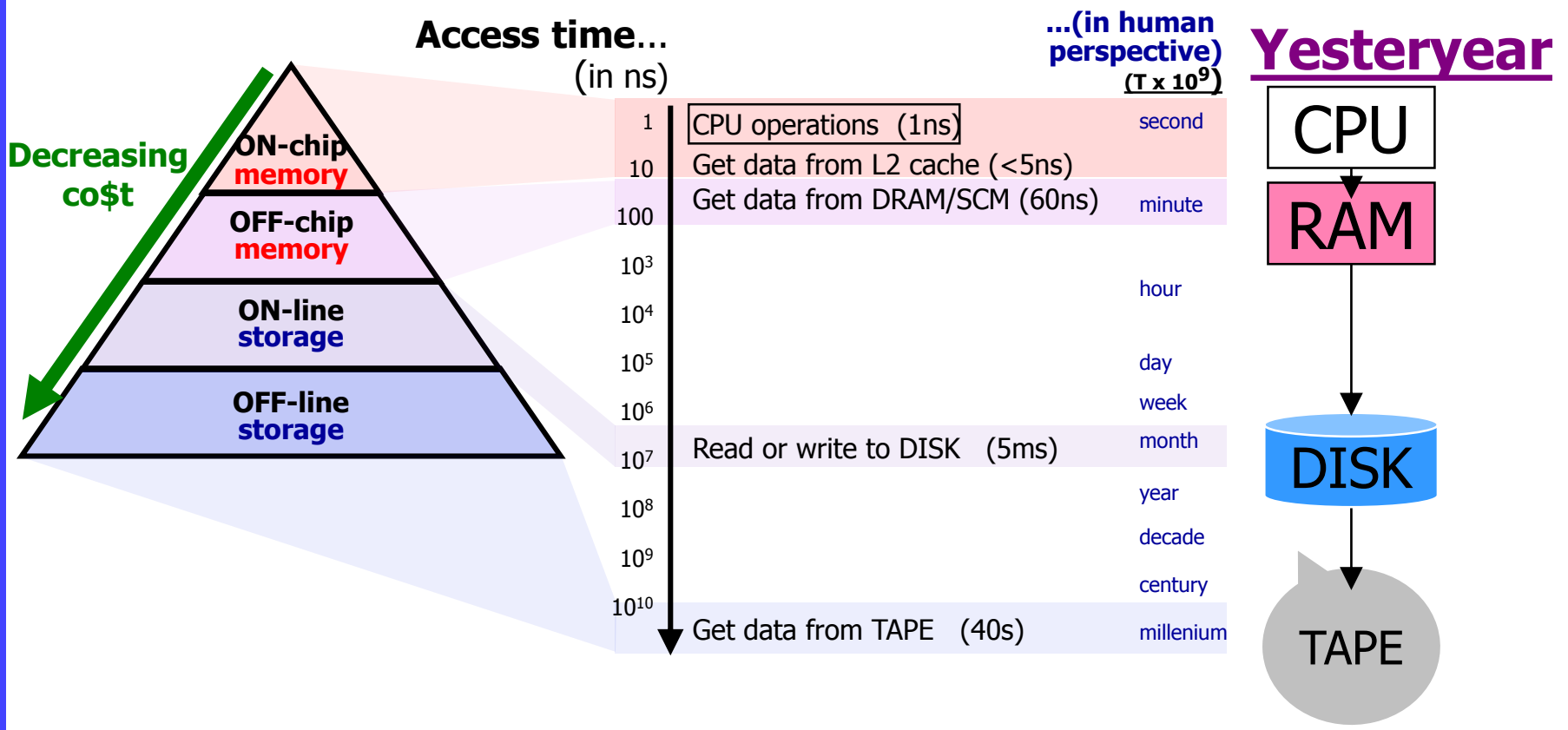
Computation-in-Memory
(Distributed computing)

Standalone
S-class Storage Class Memory
(Enhanced Flash)

Artificial synapses
(Non-VN Computing)



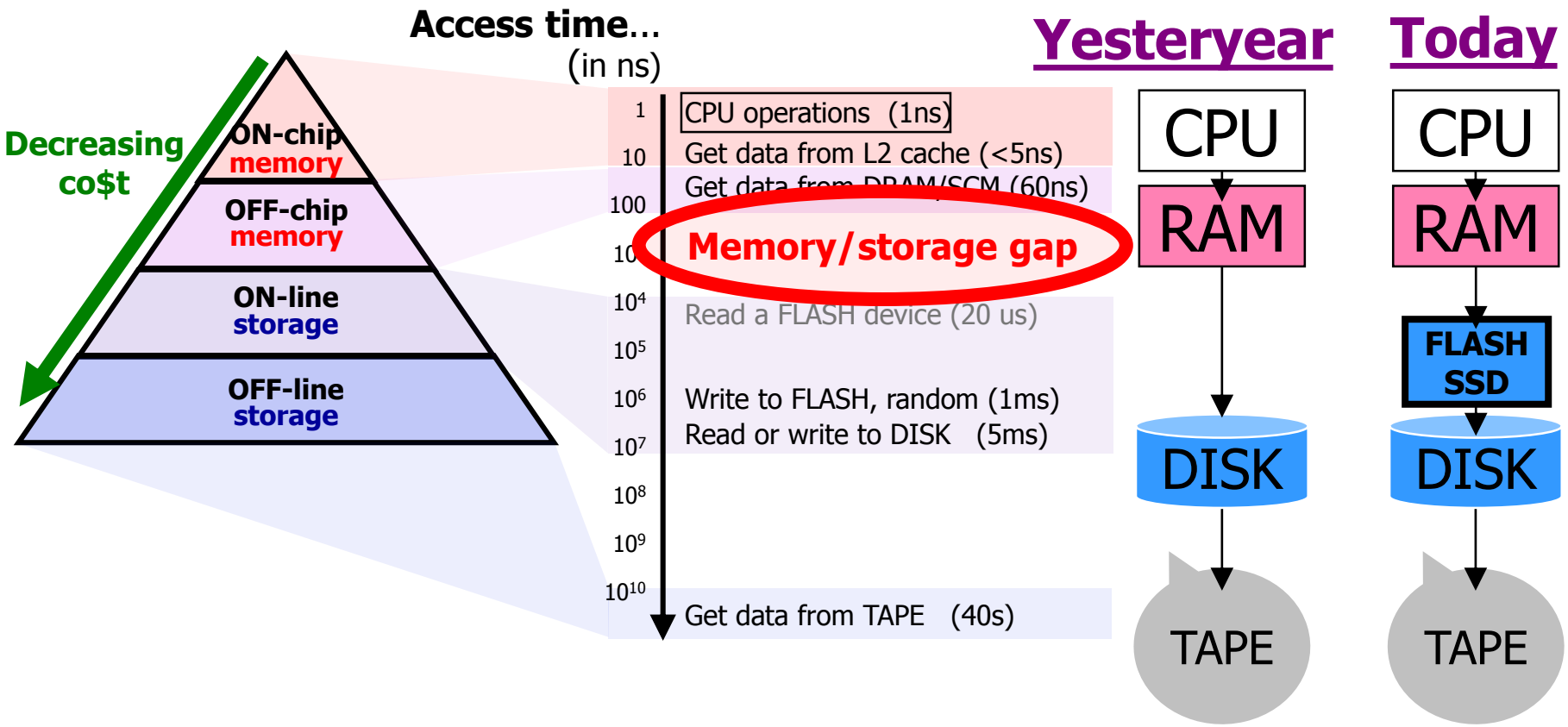
Problem (& opportunity)**:** The access-time gap between memory & storage



- Modern computer systems have long had to be designed around **hiding the access gap** between **memory** and **storage** → caching, threads, predictive branching, etc.

- "Human perspective" – if a CPU instruction is analogous to a 1-second decision by a human, retrieval of data from off-line tape represents an analogous delay of 1250 years

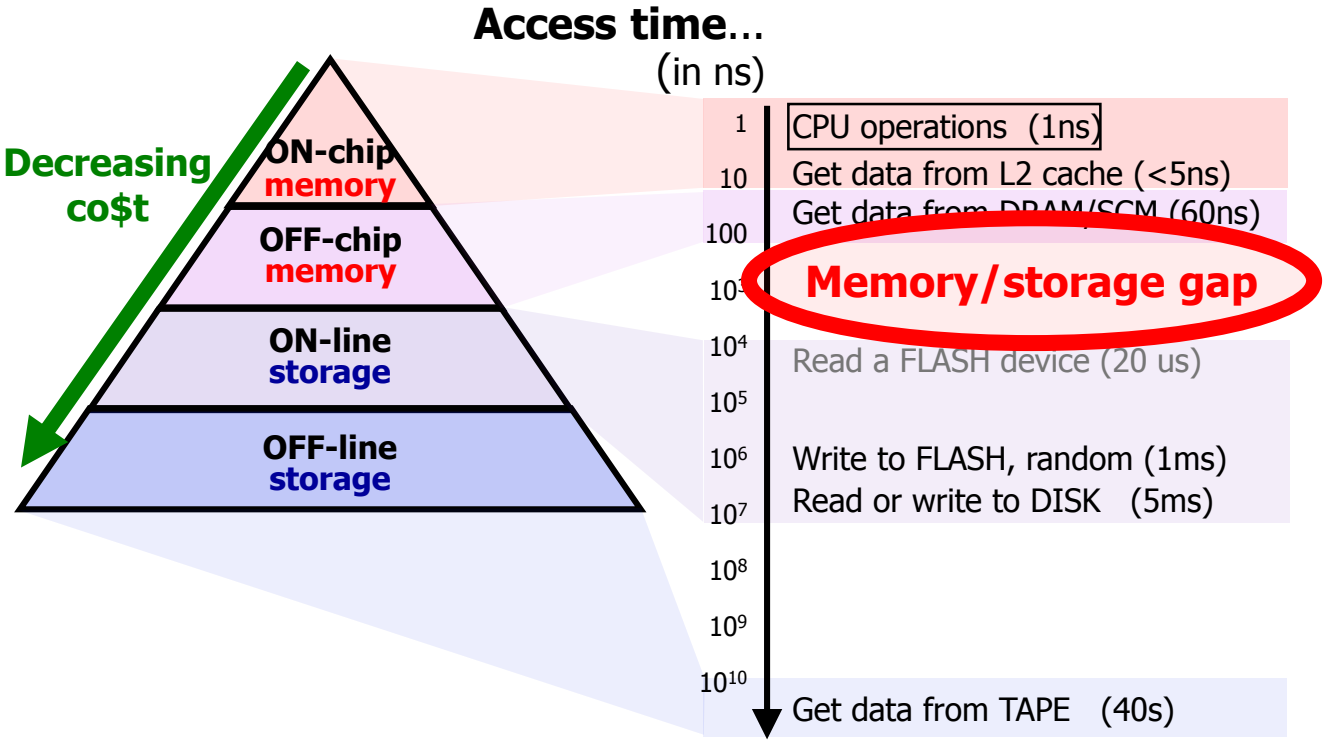
Problem (& opportunity)**:** The access-time gap between memory & storage

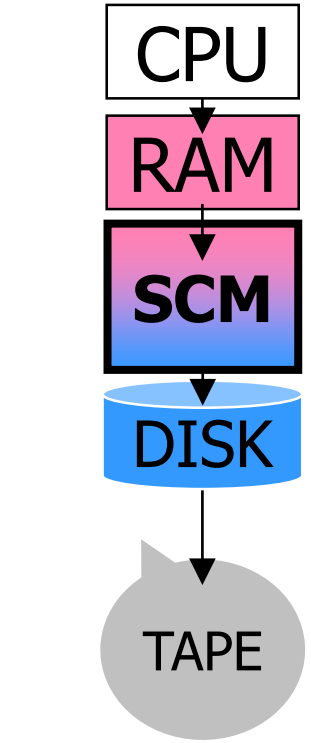


- Today, **Solid-State Disks** based on NAND Flash can offer fast ON-line storage, and storage capacities are increasing as devices scale down to smaller dimensions…

  …but while prices are dropping, the **performance gap** between memory and storage remains significant, and the **device endurance** of Flash is not likely to improve.

Problem (& opportunity): The access-time gap between memory & storage





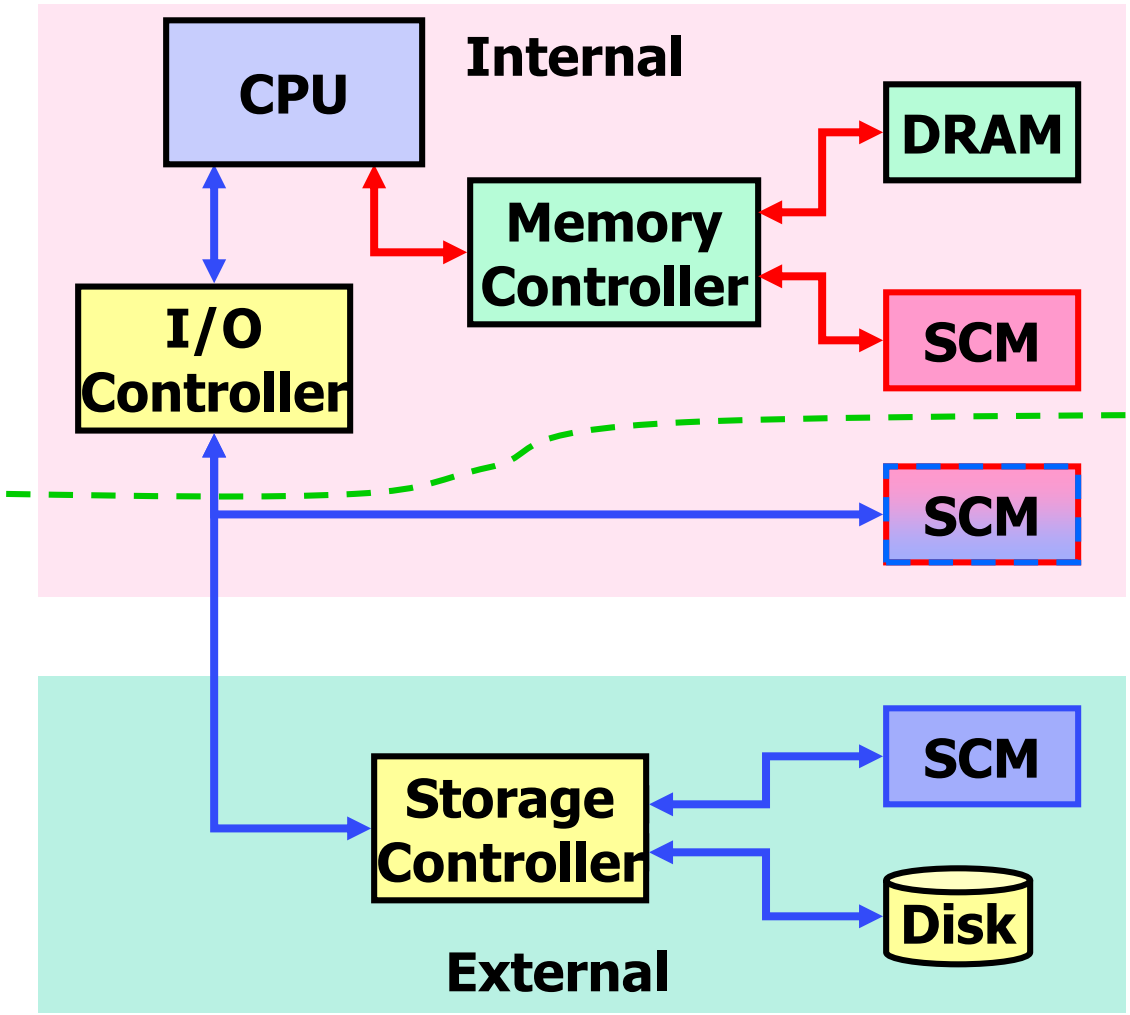
Research into new solid-state non-volatile memory candidates
– originally motivated by finding a "successor" for NAND Flash –
has opened up several interesting ways to change the memory/storage hierarchy…

1) **Embedded** Non-Volatile Memory – low-density, fast ON-chip NVM
2) **Embedded** Storage – low density, slower ON-chip storage

3) M-type **Storage Class Memory** – **high-density**, fast OFF- (or ON*)-chip NVM
4) S-type **Storage Class Memory** – **high-density**, very-near-ON-line storage

* ON-chip using 3-D packaging

S-type vs. M-type SCM



M-type: Synchronous

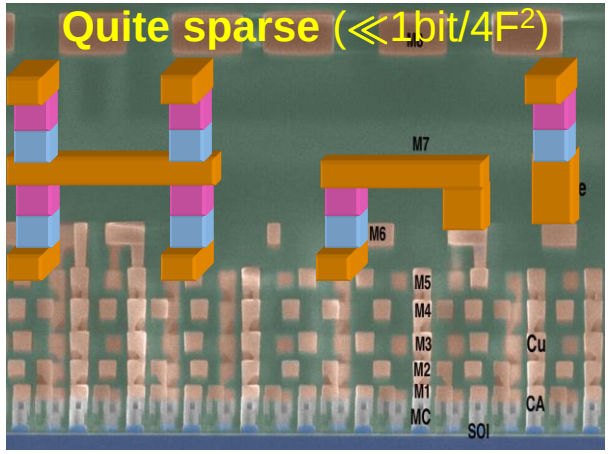
- Hardware managed
- Low overhead
- Processor waits
- New NVM → **not Flash**
- Cached or pooled memory
- Persistence (data survives despite component failure or loss of power) requires redundancy in system architecture

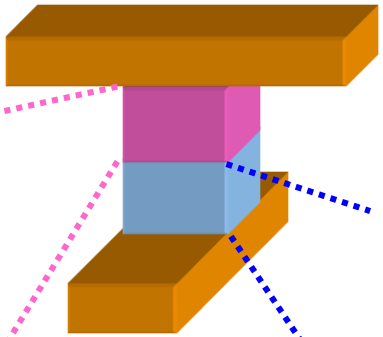
S-type: Asynchronous

- Software managed
- High overhead
- Processor doesn't wait, (process-, thread-switching)
- Flash or new NVM
- Paging or storage
- Persistence → RAID

Back-End-Of-the-Line-compatible Non-Volatile Memory:

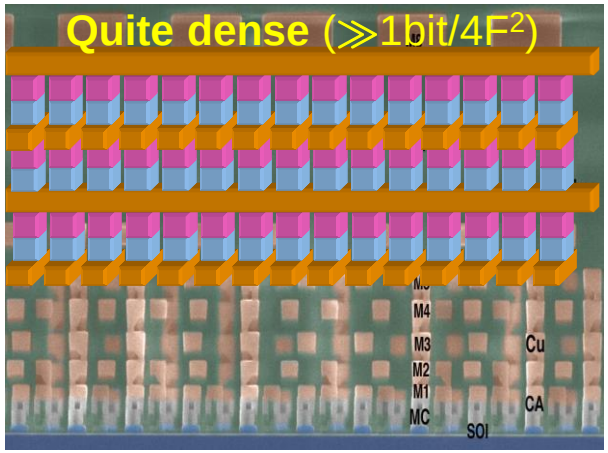
a fundamental "building block"
enabling a range of applications





Access Device (highly nonlinear or transient switch)

NVM (long-term digital storage)



Programmable e-fuses
(FPGAs, reconfigurable computing)

Embedded storage
(Automotive)

Embedded memory
(Low-power, mobile computing)

Standalone M-class SCM
(Hybrid memory)

Computation-in-Memory
(Distributed computing)

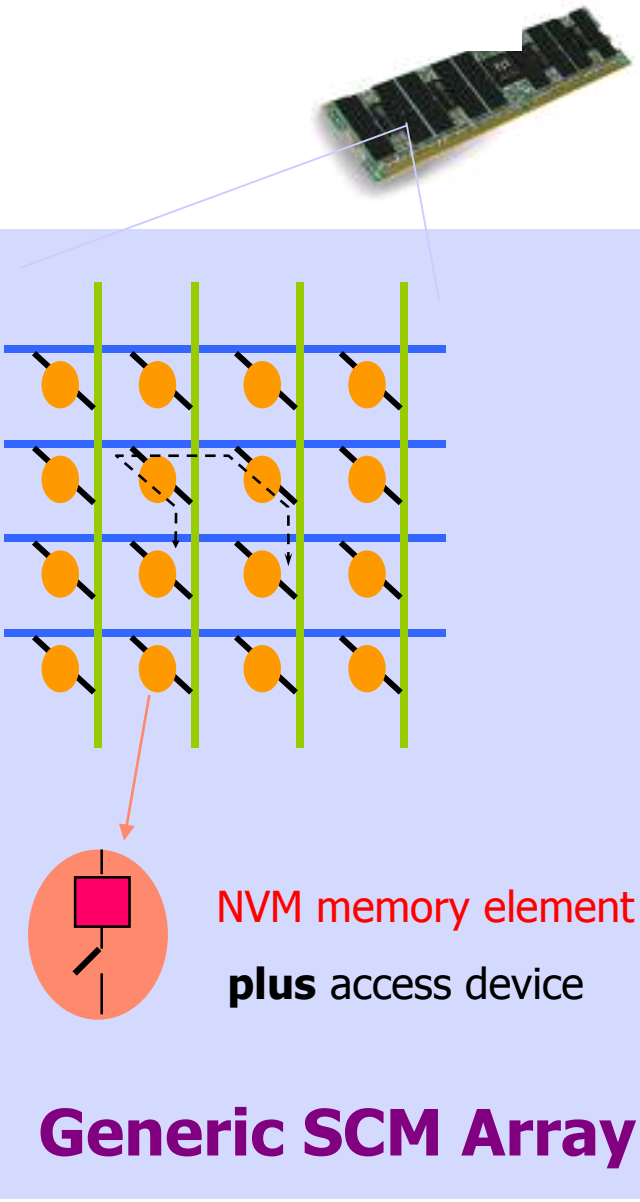
Artificial synapses
(Non-VN Computing)

Standalone
S-class Storage Class Memory
(Enhanced Flash)

Ingredients of crosspoint memory

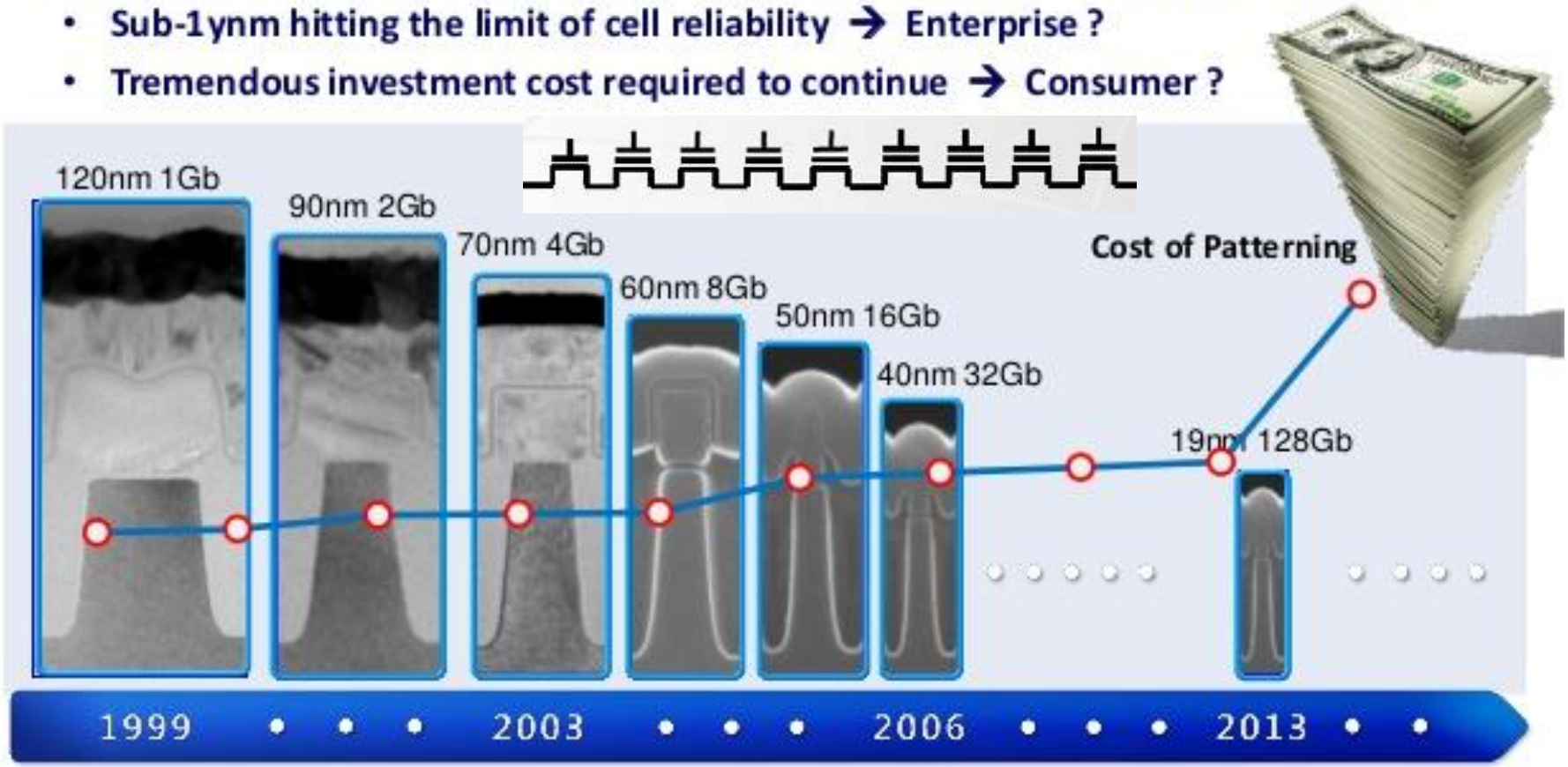
1) **NVM element**
   - **Improved FLASH**
   - **Magnetic Spin Torque Transfer**
     - → **STT-RAM**
     - → **Magnetic Racetrack**
   - **Phase Change RAM**
   - **Resistive RAM**
   - **Ferroelectric RAM**

2) High-density access device (A.D.)
   - **2-D** – silicon transistor or diode
   - **3-D** → **higher density per $4F^2$**
     - polysilicon diode (but <400°C processing?)
     - MIEC A.D. (Mixed Ionic-Electronic Conduction)
     - OTS A.D. (Ovonic Threshold Switch)
     - Conductive oxide tunnel barrier A.D.



Has Moore's Law Come to an End for NAND?

- **Maintaining planar evolution so far… But, Scaling is getting difficult**
  - Sub-1ynm hitting the limit of cell reliability → Enterprise ?
  - Tremendous investment cost required to continue → Consumer ?



Future die shrinks:

Prohibitively expensive, reliability concerns, diminishing wafer productivity

How to power the Internet of Everything with NAND?

Flash Memory SUMMIT

SAMSUNG

Limitations of Flash

Asymmetric performance

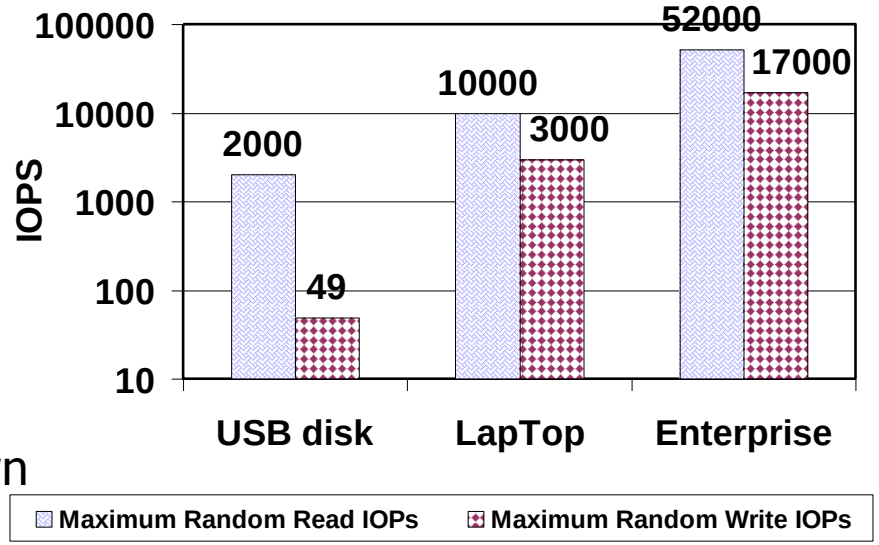
Writes much slower than reads

Program/erase cycle

Block-based, no write-in-place

Data retention and Non-volatility

Retention gets worse as Flash scales down

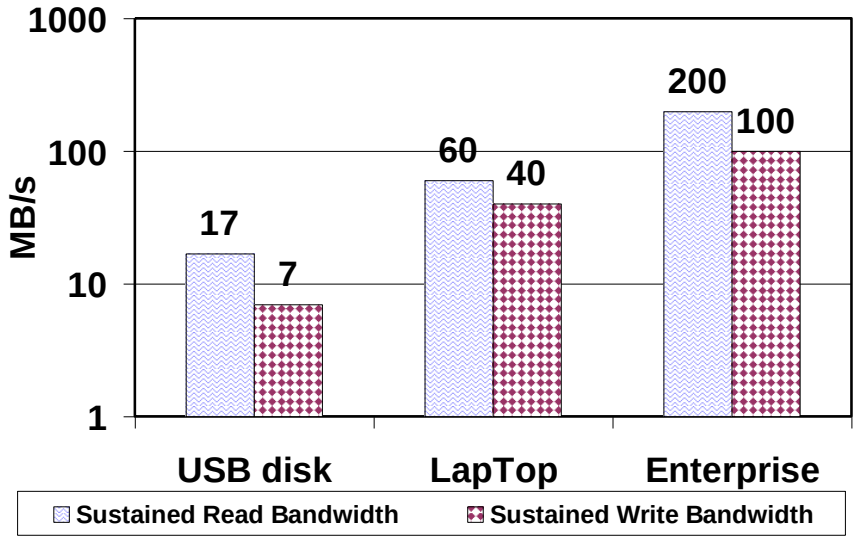


Endurance

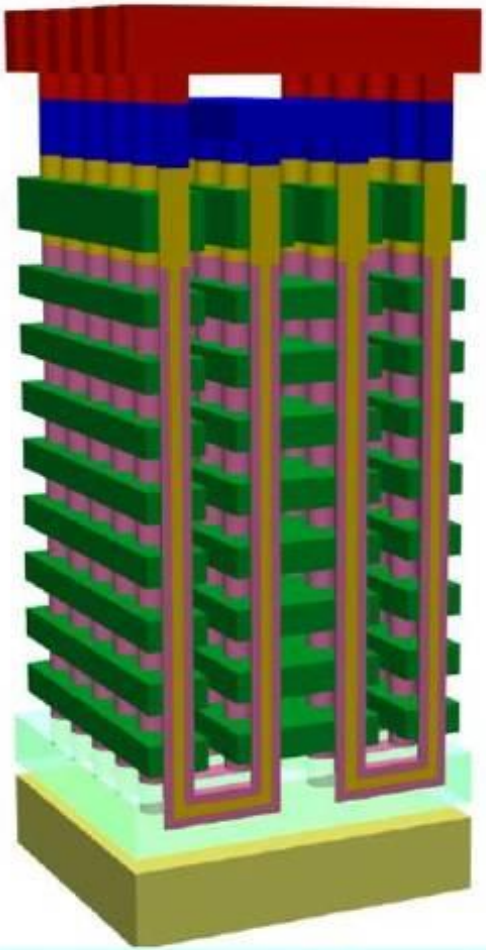
- Single level cell (SLC) → 10^5 writes/cell
- Multi level cell (MLC) → 10^4 writes/cell
- Triple level cell (TLC) → ~300 writes/cell

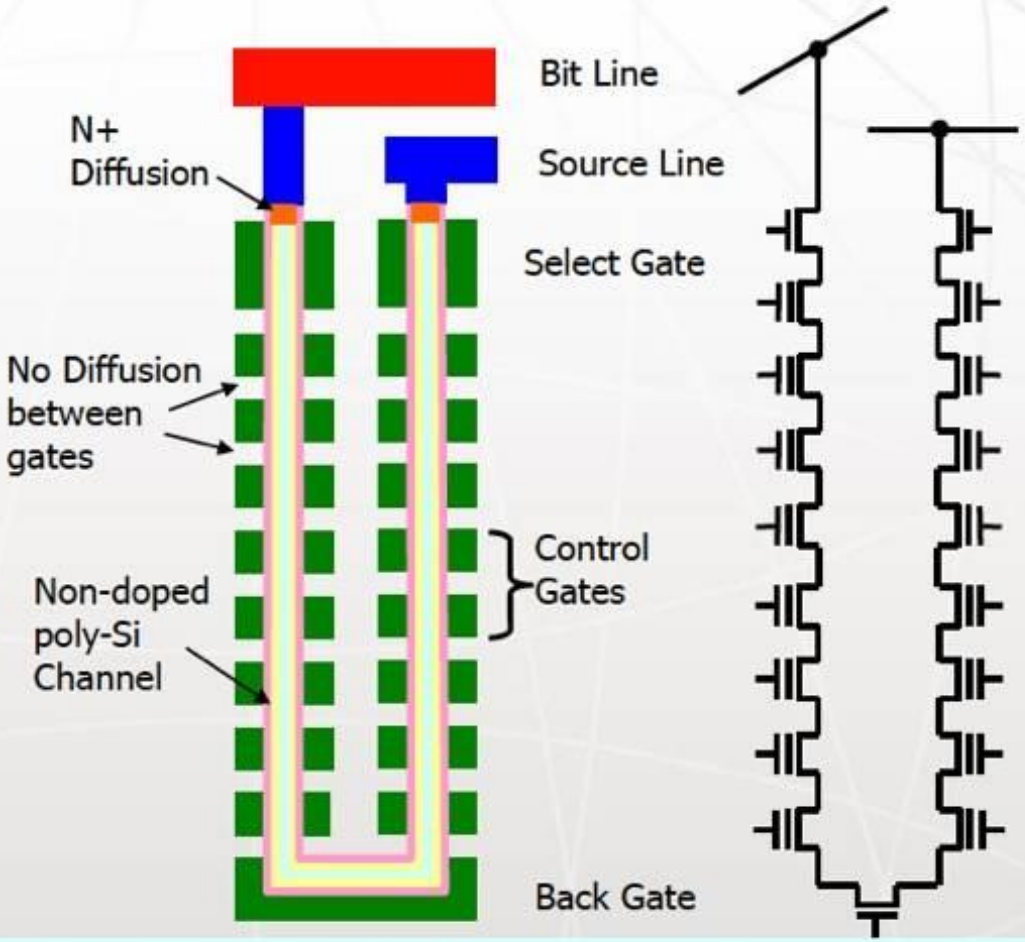
Future outlook

- Scaling focused solely on density
- but **3-D schemes worked!!**



P-BiCS Flash

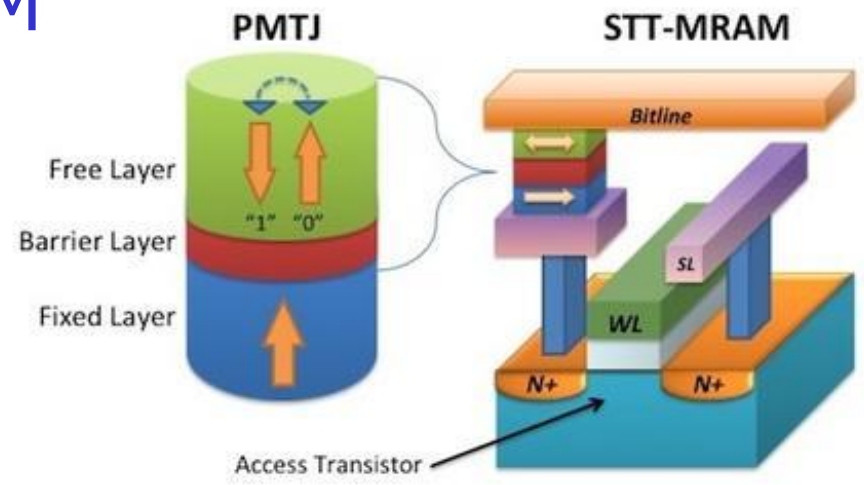




P-BiCS has "U" shaped NAND string with back gate to reduce parasitic resistance of bottom portion. There is no diffusion between CGs. Select gate has asymmetric source and drain structure to reduce off current.

STT (Spin-Torque-Transfer) RAM

- Controlled switching of free magnetic layer in a magnetic tunnel junction using current, leading to two distinct resistance states



Strengths

- Inherently very fast → **almost as fast as DRAM**
- **Much better endurance** than Flash or PCM
- Radiation-tolerant
- Materials are Back-End-Of-the-Line compatible
- Simple cell structure → reduced processing costs

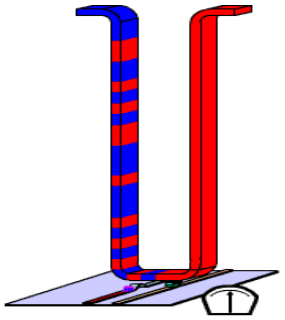
Weaknesses

- Achieving **low switching current/power** is not easy
- BEOL temperatures can affect STT-MRAM device stack
- Resistance contrast is quite low (2-3x) → achieving **tight distributions** is ultra-critical
- High-temperature retention **strongly affected by scaling** below F~50nm
- Tradeoff between fastest switching and switching reliability

Outlook:

Strong outlook for an Embedded Non-Volatile Memory to replace/augment DRAM.

Racetrack Memory offers hope for using STT concepts to create vertical "shift-register" of domain walls → potential densities of 10-100 bits/F^2



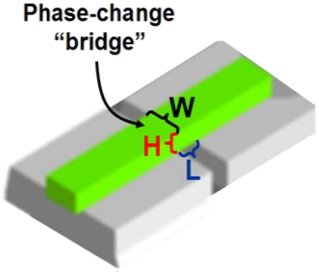
Phase Change Memory at IBM Almaden

2004

IBM/Macronix/ Qimonda Joint Project begins

2005

IEDM 2006

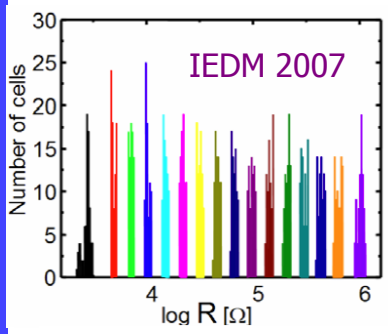


2006

Overview of candidate device technologies for storage-class memory

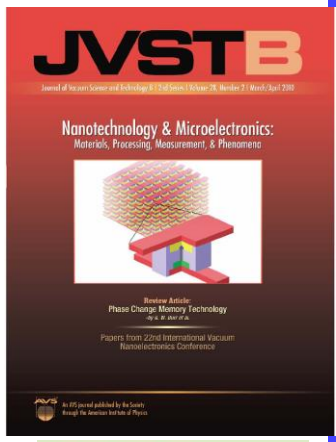
Phase-change random access memory: A scalable technology

2007



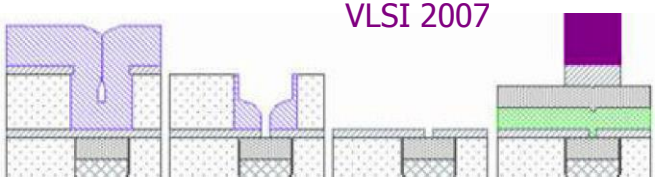
2008

2009



2010

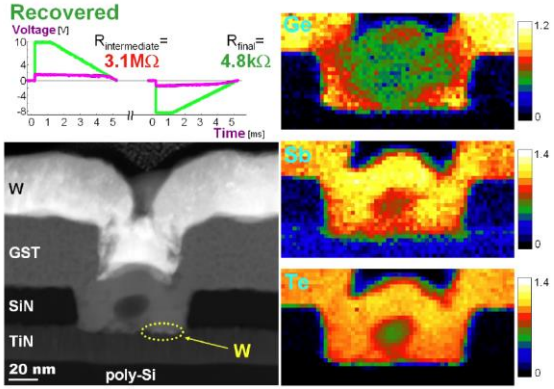
VLSI 2007



2011

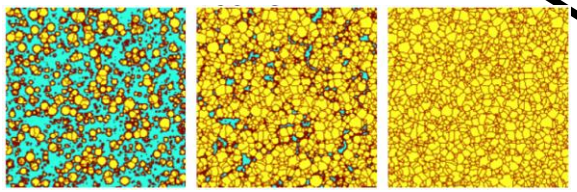
2012

IEDM 2010



2013

Science 2009
JAP 2012
MRS 2013
EPCOS 2013



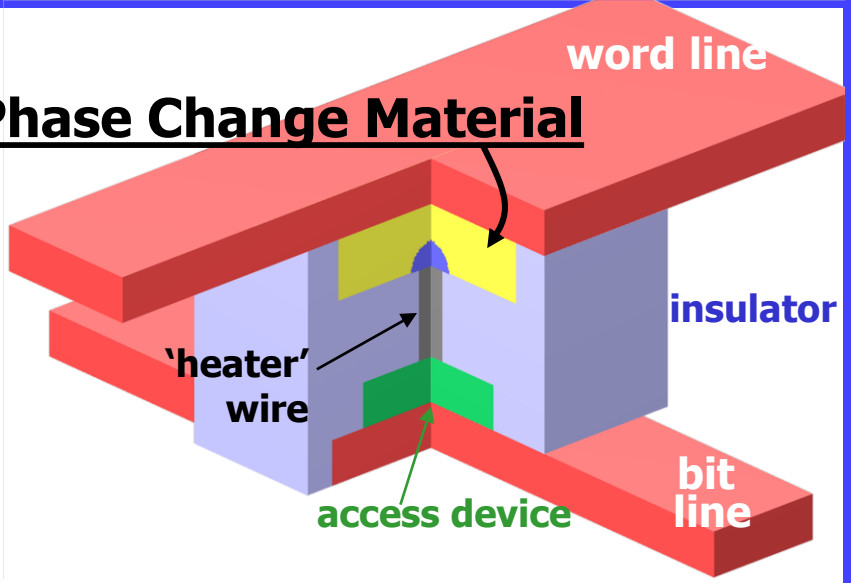
2014

2015

2016

Phase-change RAM

- Switching between **low-resistance** crystalline, and **high-resistance** amorphous phases, controlled through power & duration of electrical pulses



Strengths

- **Mature** (large-scale demos & products)
- Industry consensus on material → GeSbTe or GST
- Large resistance contrast → analog states for **MLC**
- Offers **much better endurance** than Flash
- Shown to be **highly scalable** (still works at ultra-small **F**) and Back-End-Of-the-Line compatible
- Can be very fast (depending on material & doping)
- High-resistance state undergoes **"Ovonic threshold switching"** at reasonable voltages

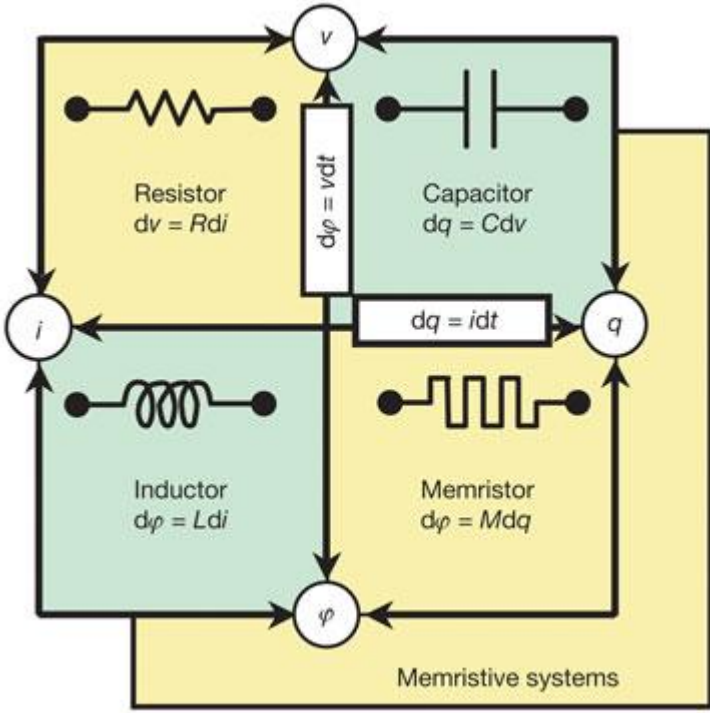
Weaknesses

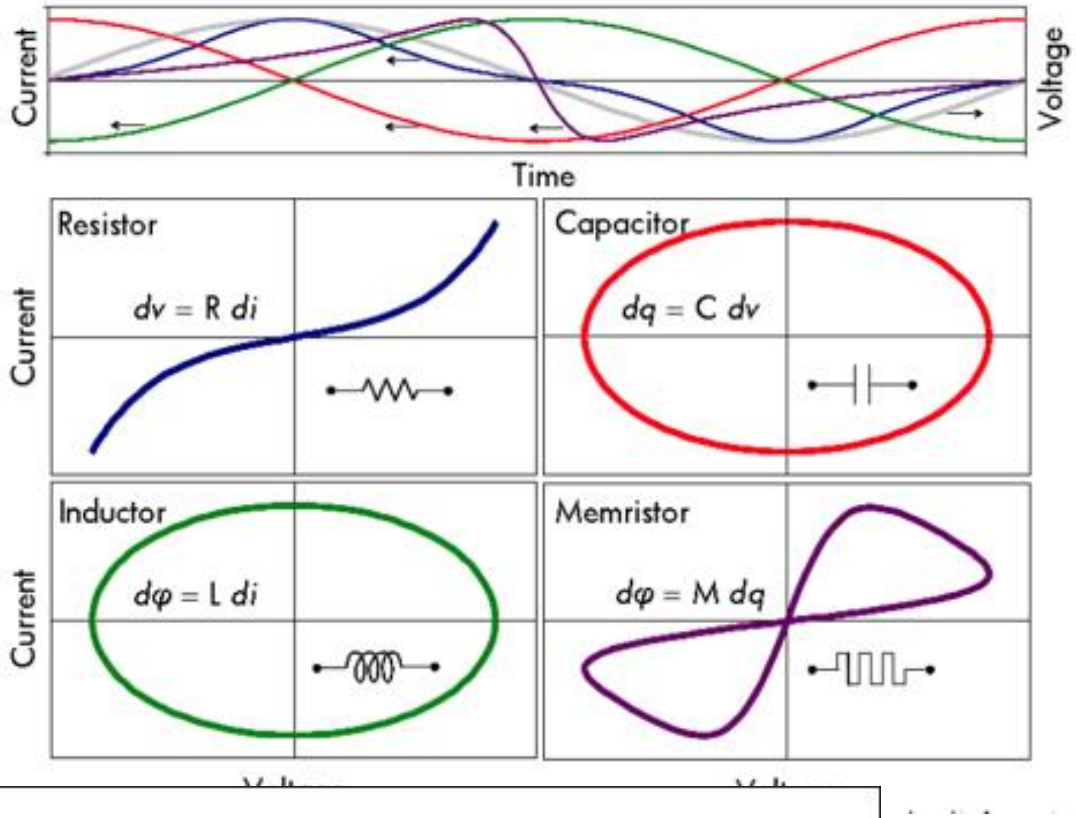
- RESET step to high resistance requires melting → **power-hungry**, thermal crosstalk?
  To keep switching power down → sub-lithographic feature and high-current Access Device
  To fill small feature → ALD or CVD → difficult now to replace GST with a better material
  Variability in small features broadens resistance distributions
- 10-year **retention at elevated temperatures** can be an issue → recrystallization
- Device characteristics change over time due to elemental segregation → device failure
- **MLC** strongly affected by relaxation of amorphous phase → "resistance drift"

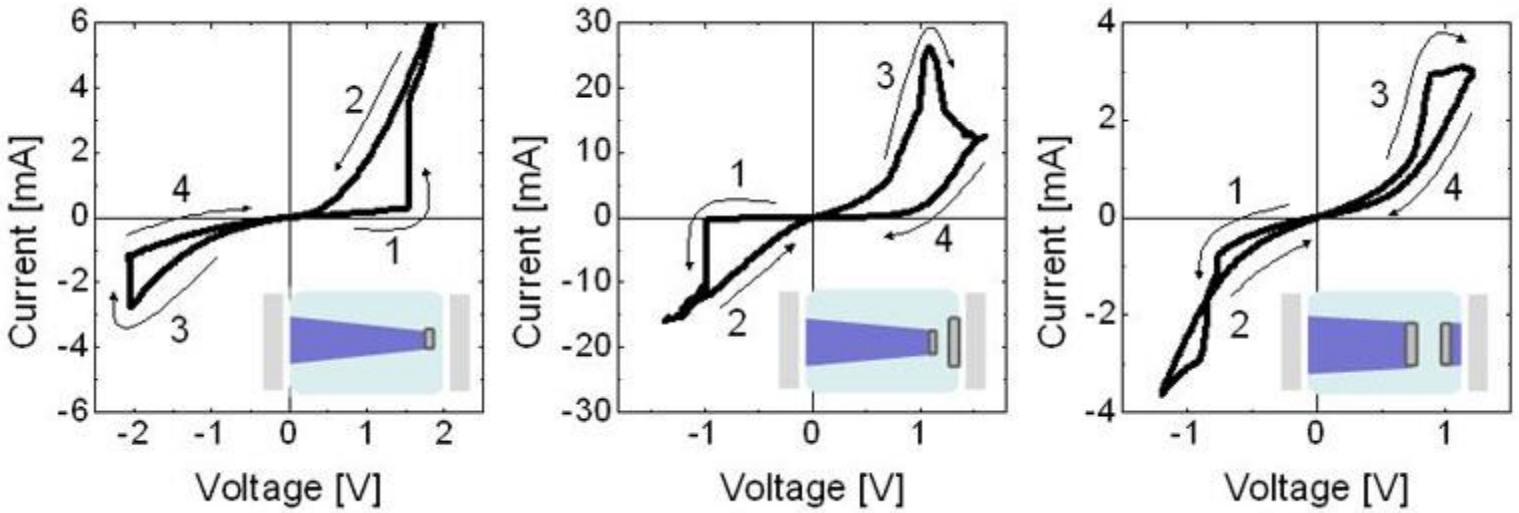
Outlook:

3D Xpoint product now in early customer assessment,
motivated by density (two layers of 3-D Access Devices) –
if successful at S-type SCM → maybe opportunity for M-type SCM as well…

Memristors



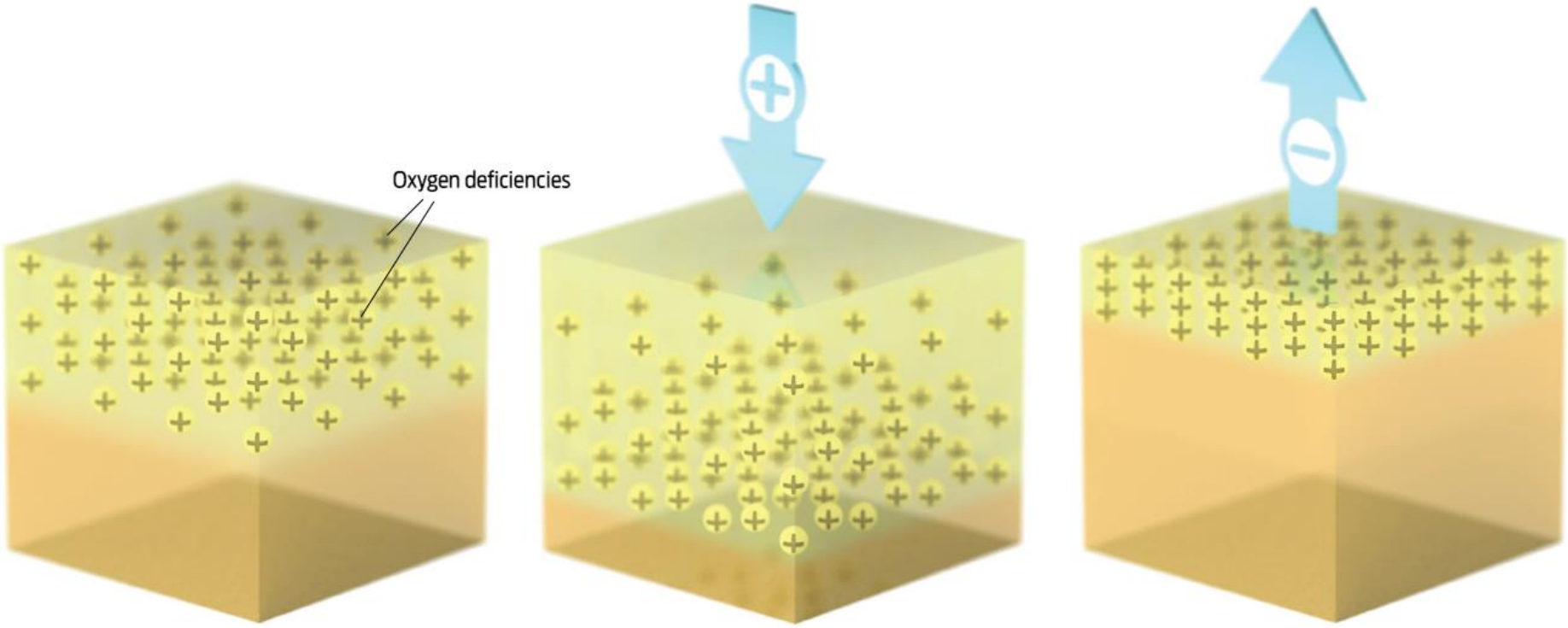




http://nanotechweb.org/cws/article/lab/43477

Memristors

original explanation as a "non-filamentary RRAM"
turned out to be incorrect for HP's material system (TiO_2)



32 NA · IEEE SPECTRUM · DECEMBER 2008

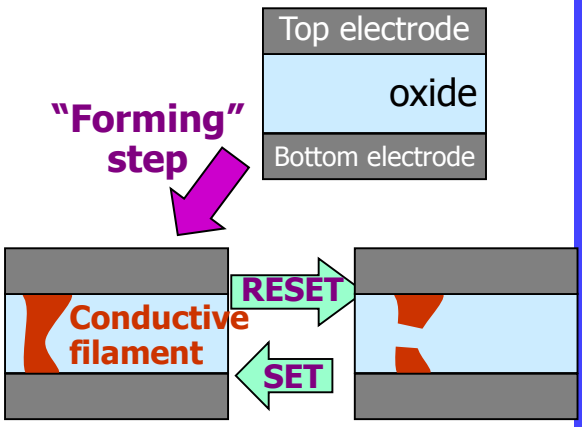
WWW.SPECTRUM.IEEE.ORG

Resistive RAM

Voltage-controlled formation & dissipation of an oxygen-vacancy (or metallic) filament through an otherwise insulating layer

Strengths

- Good retention at elevated-temperatures
- Simple cell structure → reduced processing costs
- Both fast and ultra-low-current switching have been demonstrated
- Some RRAM materials are Back-End-Of-the-Line compatible
- Was a new field → high hopes for improved material concepts
- Less "gating" Intellectual Property to license
- Some RRAM concepts offer co-integrated NVM & Access Device
- Possibility for 3-D silo scheme like 3-D Flash



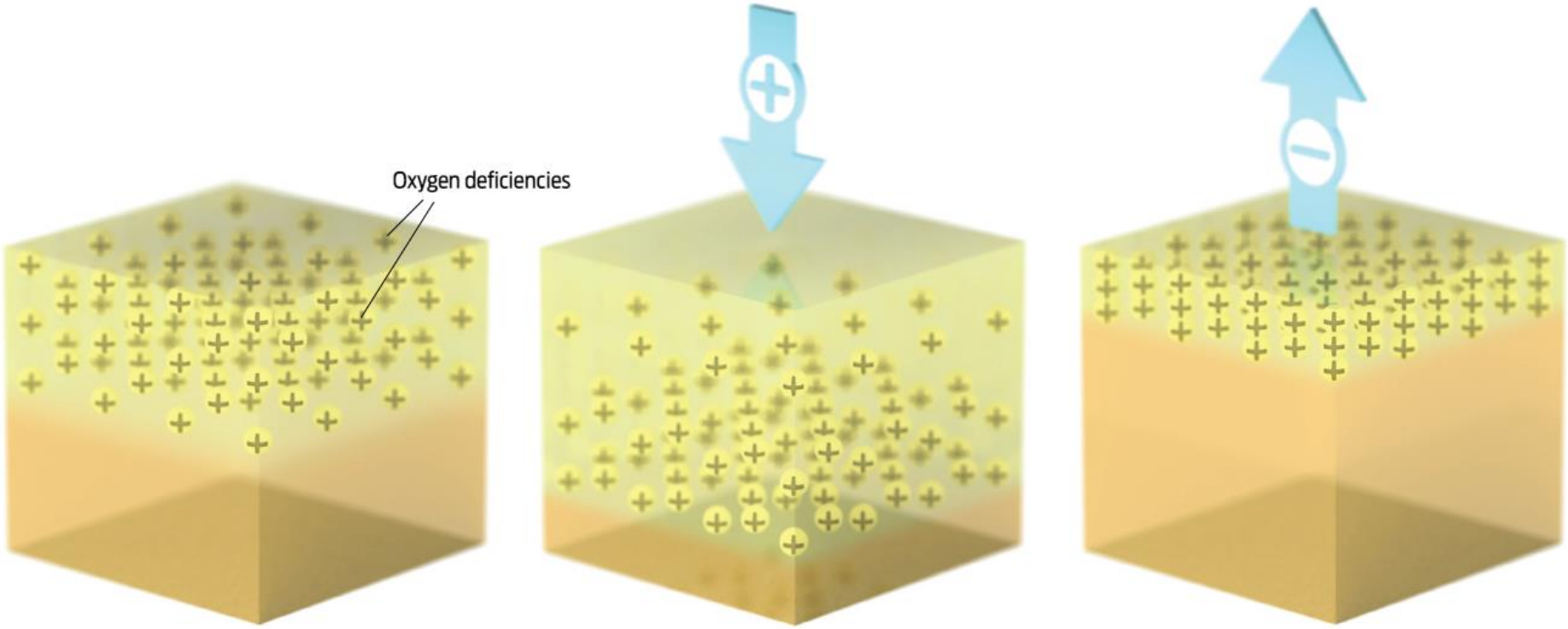
Weaknesses

- Highly immature technology – wide variation in materials hampers cross-industry learning
- Demonstrated endurance is slightly better than Flash, but lower than PCM or STT-RAM
- **Switching reliability** an issue, even within single devices, and read disturb can be an issue
- An initial high-voltage **"forming"** step is often required
- To attain low RESET switching currents, circuit must constrain current during previous SET
- Unipolar and bipolar versions – bipolar typically better in both write margins & endurance, but then requires an unconventional bipolar-capable Access Device (silicon diode is out)
- Intra-device variability **MUCH worse** at low power → **very few atoms** in the filament
- CBRAM (metal atom filament) offers higher signal contrast than RRAM (defect filament)

Outlook:

Outlook is unclear. Clear opportunities for embedded memory. Opportunity for following 3-D Flash into Z dimension attractive, but many uncertainties remain about prospects for reliable storage & memory products.

Non-filamentary RRAM



Make diffusion easy → fast writes… but poor retention.

Make diffusion hard → great retention… but writing is slow.

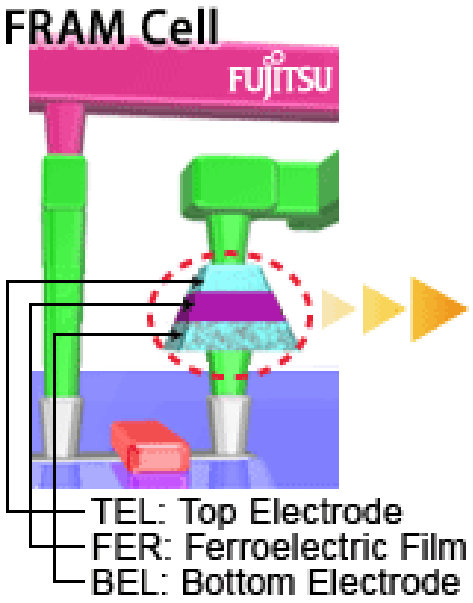
→ "voltage-time" dilemma

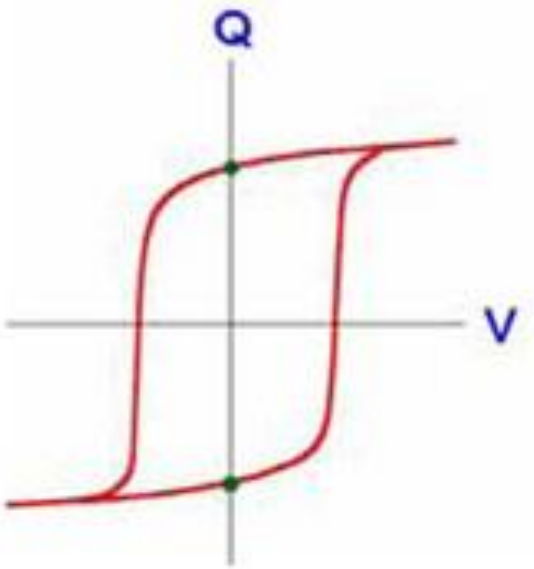
Ferroelectrics

FeRAM

- Fast, high endurance

- Imprint, fatigue
- Destructive read
- Poor density

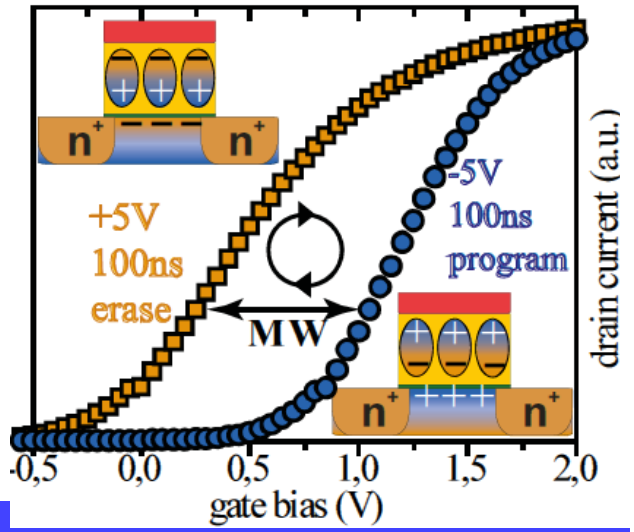




FeFET

- Fast, high endurance

- Imprint, fatigue?
- ~~How to integrate?~~
- ~~Poor density~~
- Also acts as a Flash memory
  → reduced endurance



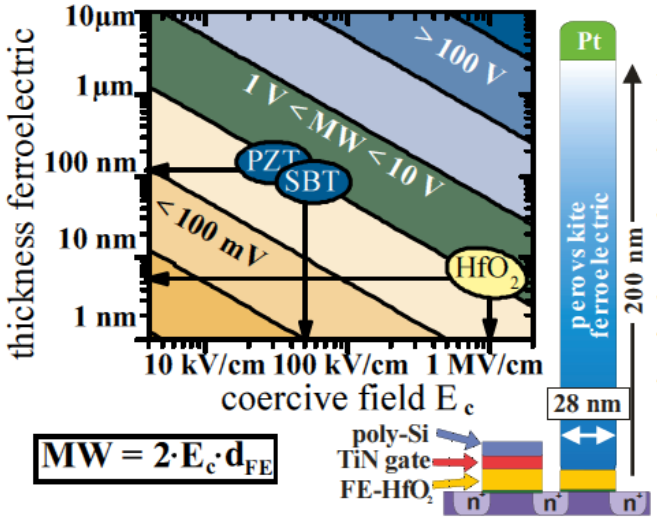


Fig 11 Scaling dilemma of the perovskite-based MFIS-FET. The high E_c and scalable d_{FE} of FE-HfO_2 preserve manufacturability of the gate stack.

Non-volatile memory wish-list

PCM

- Need lower co$t
  - ALD process for a fast, robust material
  - Reliable sub-lithographic patterning

CBRAM

- Still need better reliability

MRAM

- More resistance contrast!
- Solve tradeoffs between retention and switching current
- Solve process difficulties (temperature, etching)

Filamentary-RRAM

- need to solve few-atom switching as only path to low-power

Nonfilamentary-RRAM

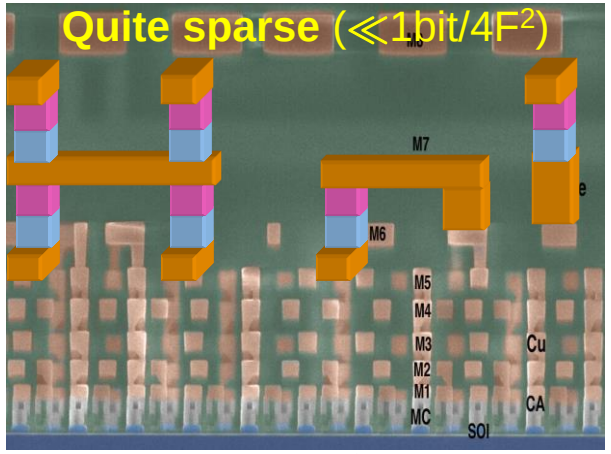
- Voltage-dilemma – how to get fast switching AND good retention?

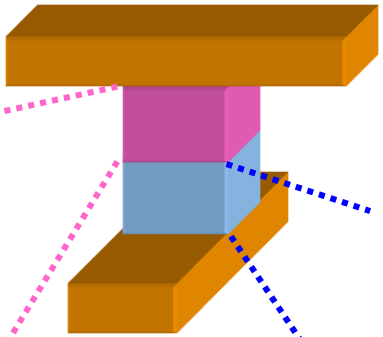
FeRAM

- HfOx results very exciting – materials work needs to lead to more device implementations so learn what the issues might be…

Back-End-Of-the-Line-compatible Non-Volatile Memory:

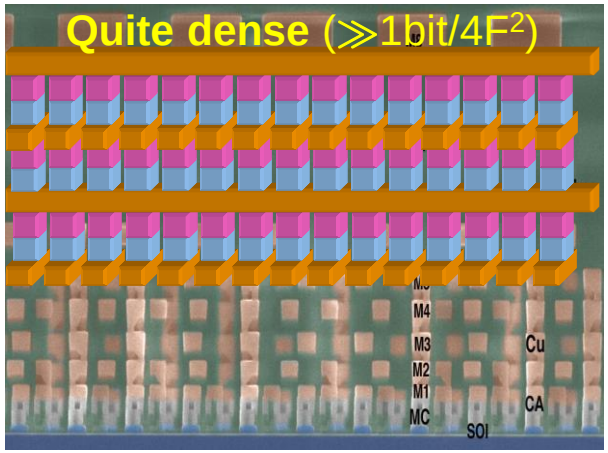
a fundamental "building block"
enabling a range of applications





Access Device (highly nonlinear or transient switch)

NVM (long-term digital storage)



Programmable e-fuses
(FPGAs, reconfigurable computing)

Embedded storage
(Automotive)

Embedded memory
(Low-power, mobile computing)

Standalone M-class SCM
(Hybrid memory)

Computation-in-Memory
(Distributed computing)

Artificial synapses
(Non-VN Computing)

Standalone
S-class Storage Class Memory
(Enhanced Flash)

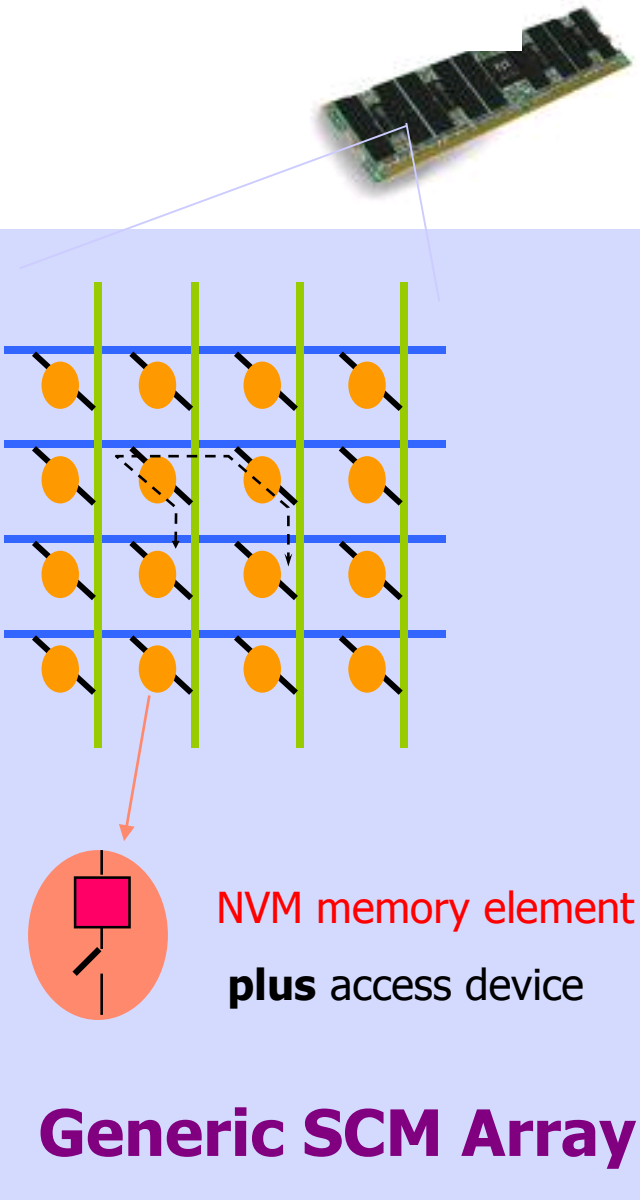
Ingredients of crosspoint memory

1) NVM element

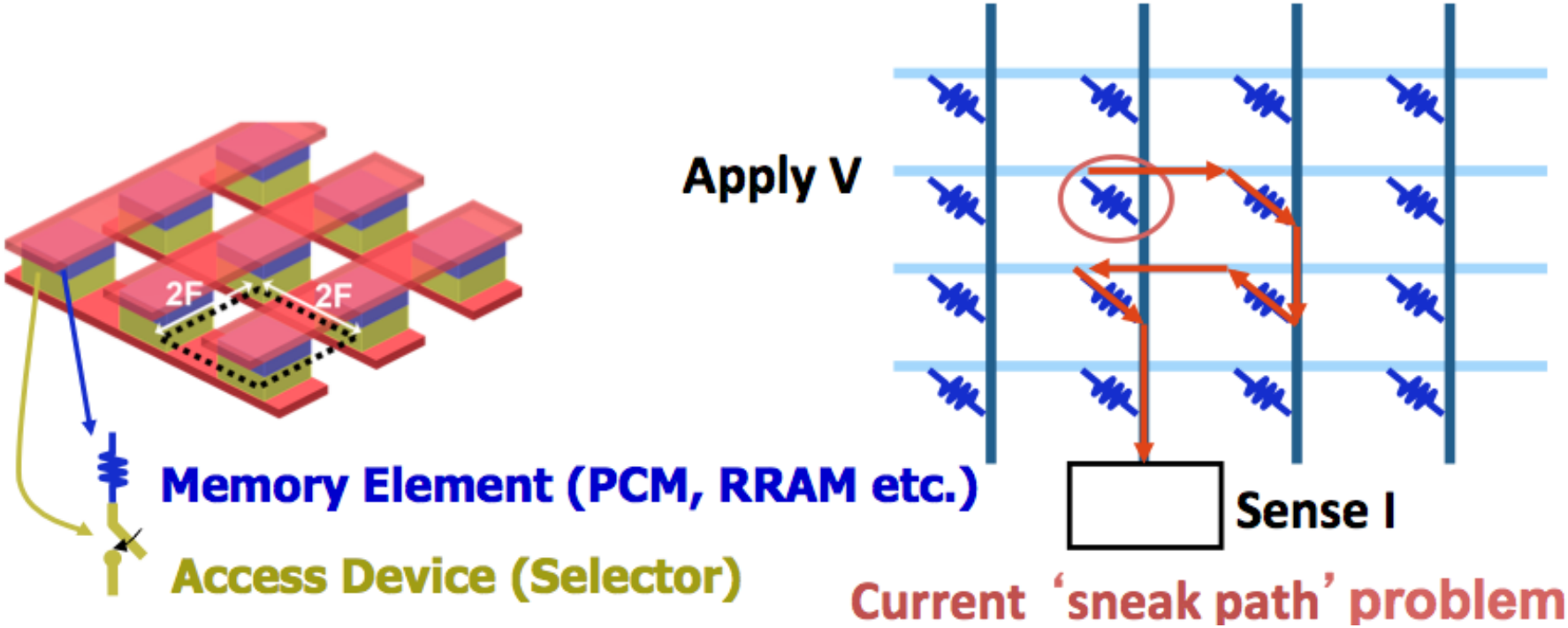
- **Improved FLASH**
- **Magnetic Spin Torque Transfer**
  - → **STT-RAM**
  - → **Magnetic Racetrack**
- **Phase Change RAM**
- **Resistive RAM**

2) **High-density access device (A.D.)**

- **2-D** – silicon transistor or diode
- **3-D** → **higher density per $4F^2$**
  - polysilicon diode (but <400°C processing?)
  - MIEC A.D. (Mixed Ionic-Electronic Conduction)
  - OTS A.D. (Ovonic Threshold Switch)
  - Conductive oxide tunnel barrier A.D.



Need for an Access Device



Access device needed in series with memory element

- Cut off current 'sneak paths'

  that lead to incorrect sensing and wasted power
- Typically diodes used as access devices
- Could also use devices with highly non-linear I-V curves

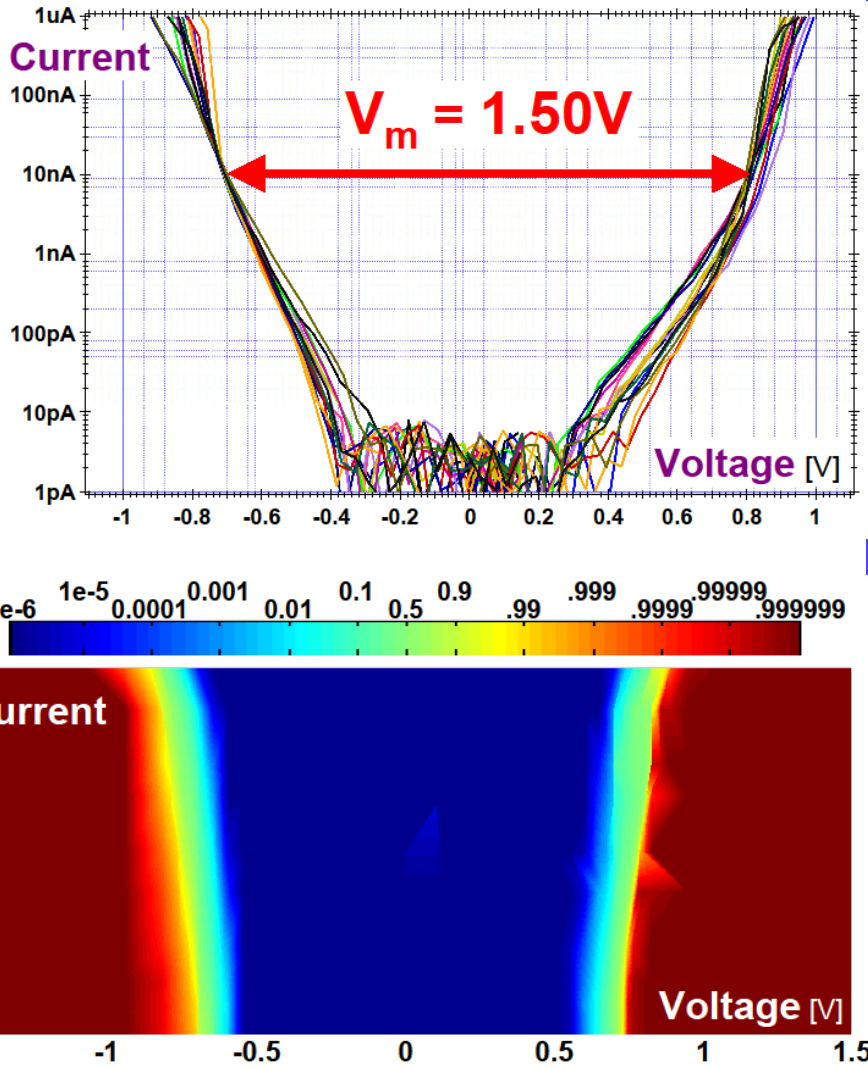
Novel **M**ixed-**I**onic-**E**lectronic-**C**onduction (MIEC) Access Device

Strengths

- **High** enough **ON currents** for PCM – cycling of PCM has been demonstrated
- **Low** enough **OFF current** for large arrays
- Very large (>>1e10) endurance for typical 5uA read currents
- Voltage margins > 1.5V with tight distributions → sufficient for large arrays
- CMP process demonstrated
- 512kBit arrays demonstrated w/ 100% yield
- Scalable to <30nm CD, <12nm thickness
- Capable of 15ns write, 50ns read
- Highly stable in un-/half-select conditions

Weaknesses

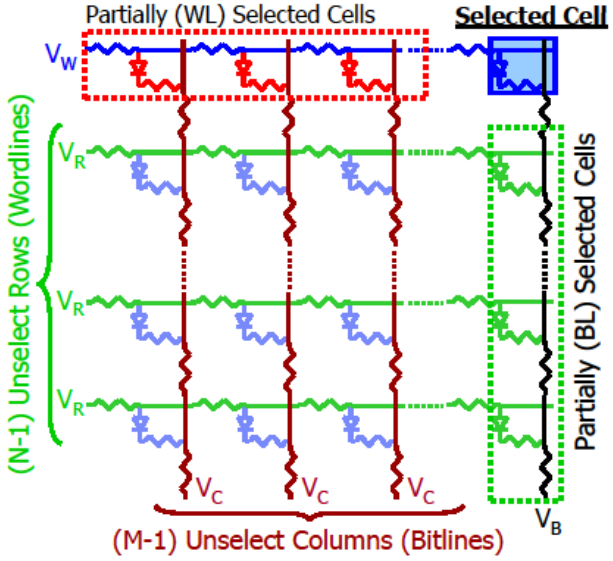
- Maximum voltage across companion NVM during switching must be low (1-2V) → influences half-select condition and thus achievable array size
- **Endurance during NVM programming** is strongly dependent on programming current

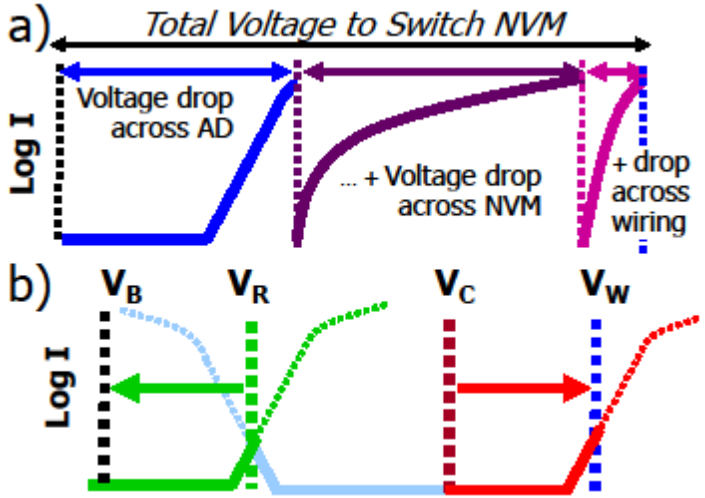


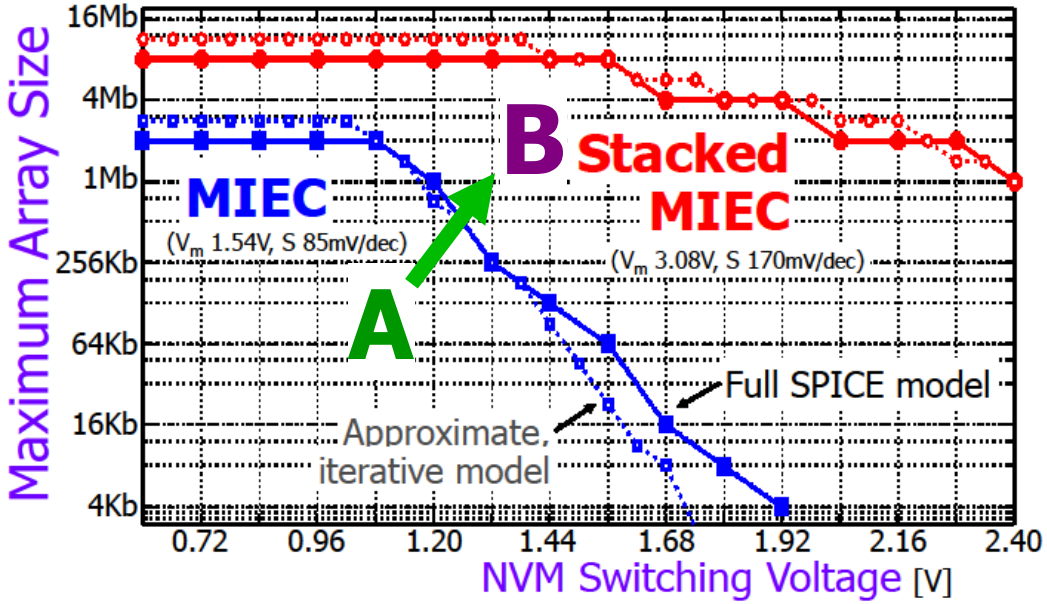
Gopalakrishnan, VLSI 2010
Shenoy, VLSI 2011
Burr, VLSI 2012
Virwani, IEDM 2012
Burr, VLSI 2013

Shenoy, *Semi. Sci. Tech.* **29**/104005 (2014)
Burr, *JVST-B* **32**/040802 (2014)
Narayanan, DRC & IEDM 2014, *J-EDS* **3**/423 (2015), *IEEE J.ESTC&S* (2016)
Padilla, IEEE-TED 62/963 (2015)

DRC 2014 – Crossbar array design using SPICE modeling







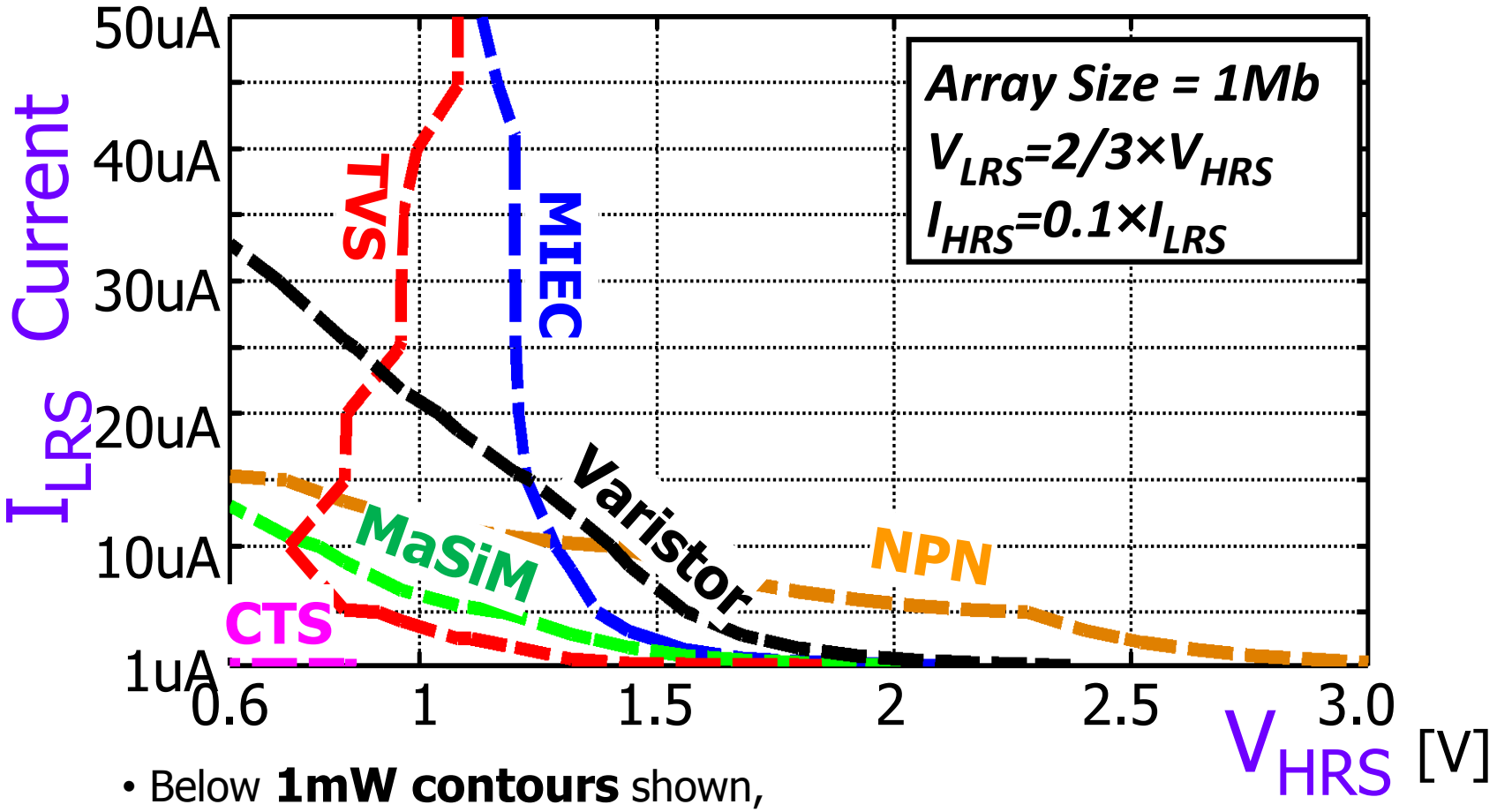
A) Efficient design point:
nearly all injected power delivered to "selected" device(s)

B) Inefficient design point:
much more injected power, which is mostly **dissipated** in "unselected" devices!!

IEDM 2014 paper: compare access devices using SPICE

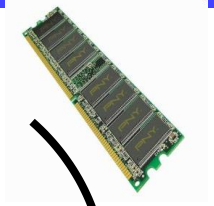
Circuit-Level Benchmarking of Access Devices for Resistive Nonvolatile Memory Arrays

P. Narayanan, G. W. Burr, R. S. Shenoy, K. Virwani, and B. Kurdi



- Below **1mW contours** shown, parallel writes are still a viable option…

Paths towards SCM



Device Availability

Capital investment

Co$t

Applications

NAND — **3-D NAND** → **Future NAND applications** (consumer devices, etc.)

This path no longer possible

1-10us emerging **NVM**

RRAM? PCM? CBRAM?

→ **Embedded** Storage (low density, slower ON-chip storage) → S-type **SCM** (**high-density**, near-ON-line storage)

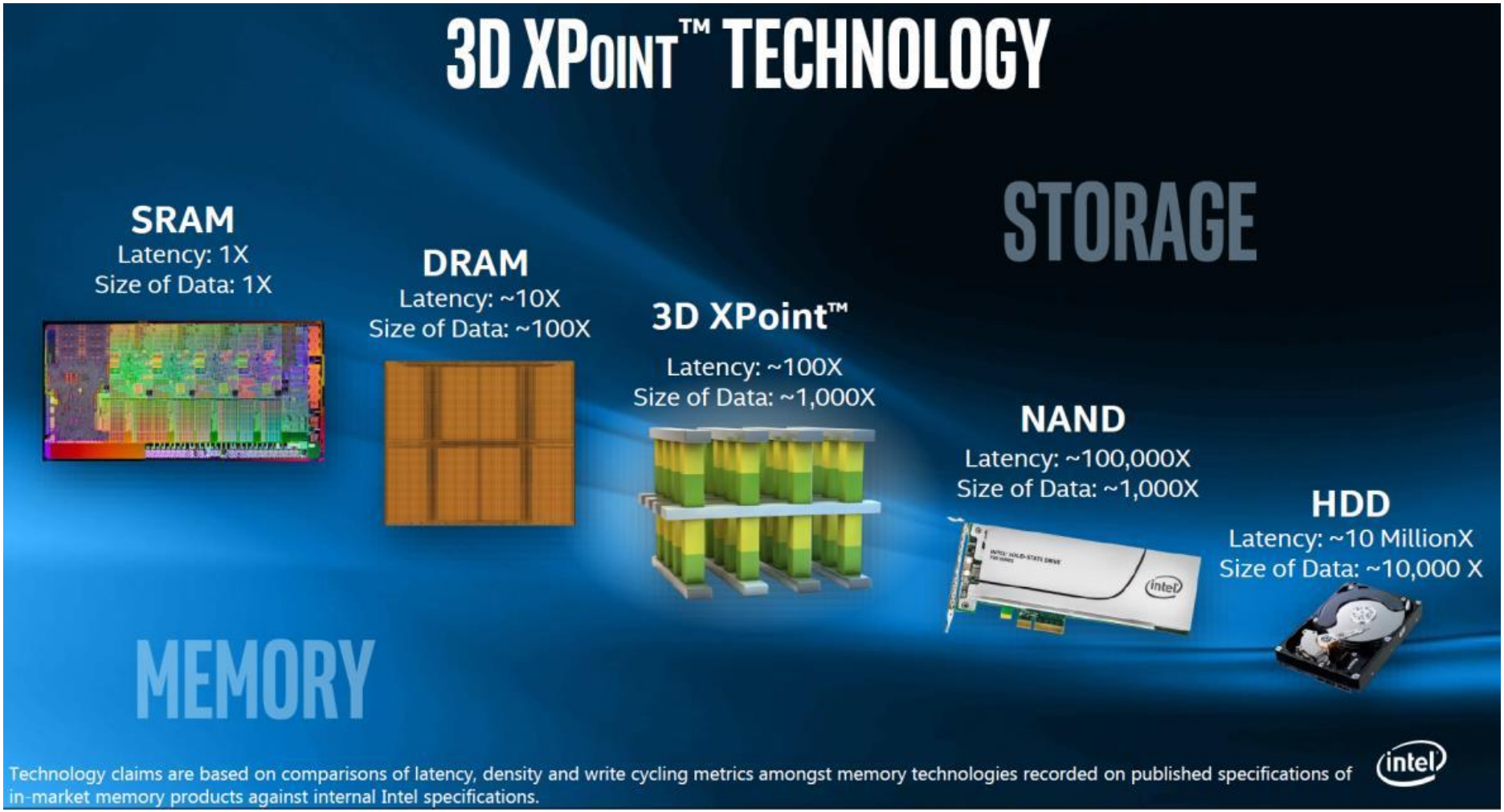
<<1us emerging **NVM**

STT-RAM? CBRAM? PCM??/RRAM??

→ **Embedded** Non-Volatile Memory (low-density, fast ON-chip NVM) → M-type **SCM** (**high-density**, fast OFF-(or ON*) -chip NVM)

* ON-chip using 3-D packaging

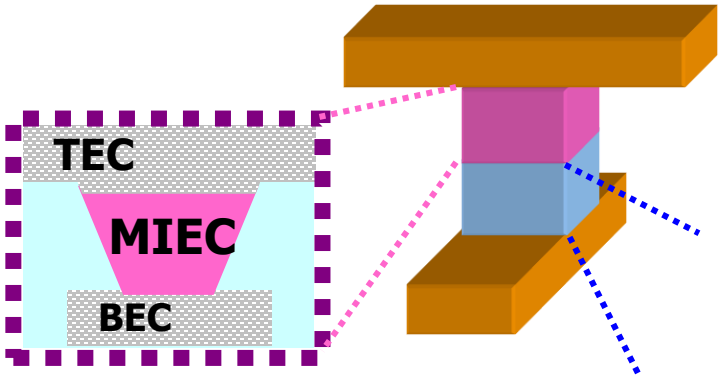
DRAM → **Future DRAM** (working memory, etc.)



3D-XPoint believed to be PCM + OTS

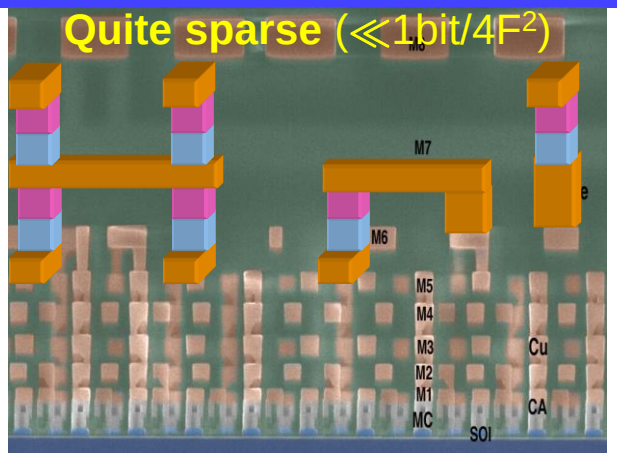
Phase-Change Memory + Ovonic Threshold Switch

MIEC+NVM: a fundamental, BEOL-compatible "building block"



Access Device

PCM
RRAM
CBRAM
MRAM



Programmable e-fuses
(FPGAs, reconfigurable computing)

Embedded storage
(Automotive)

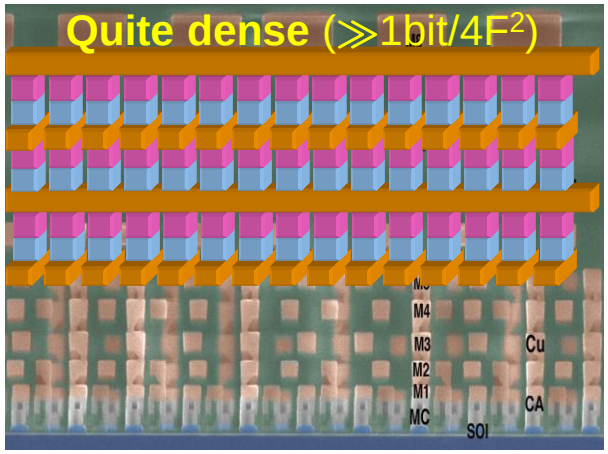
Embedded memory
(Low-power, mobile computing)

Standalone M-class SCM
(Hybrid memory)

Computation-in-Memory
(Distributed computing)

Standalone
S-class SCM
(Enhanced Flash)

Artificial synapses
(Non-VN Computing)

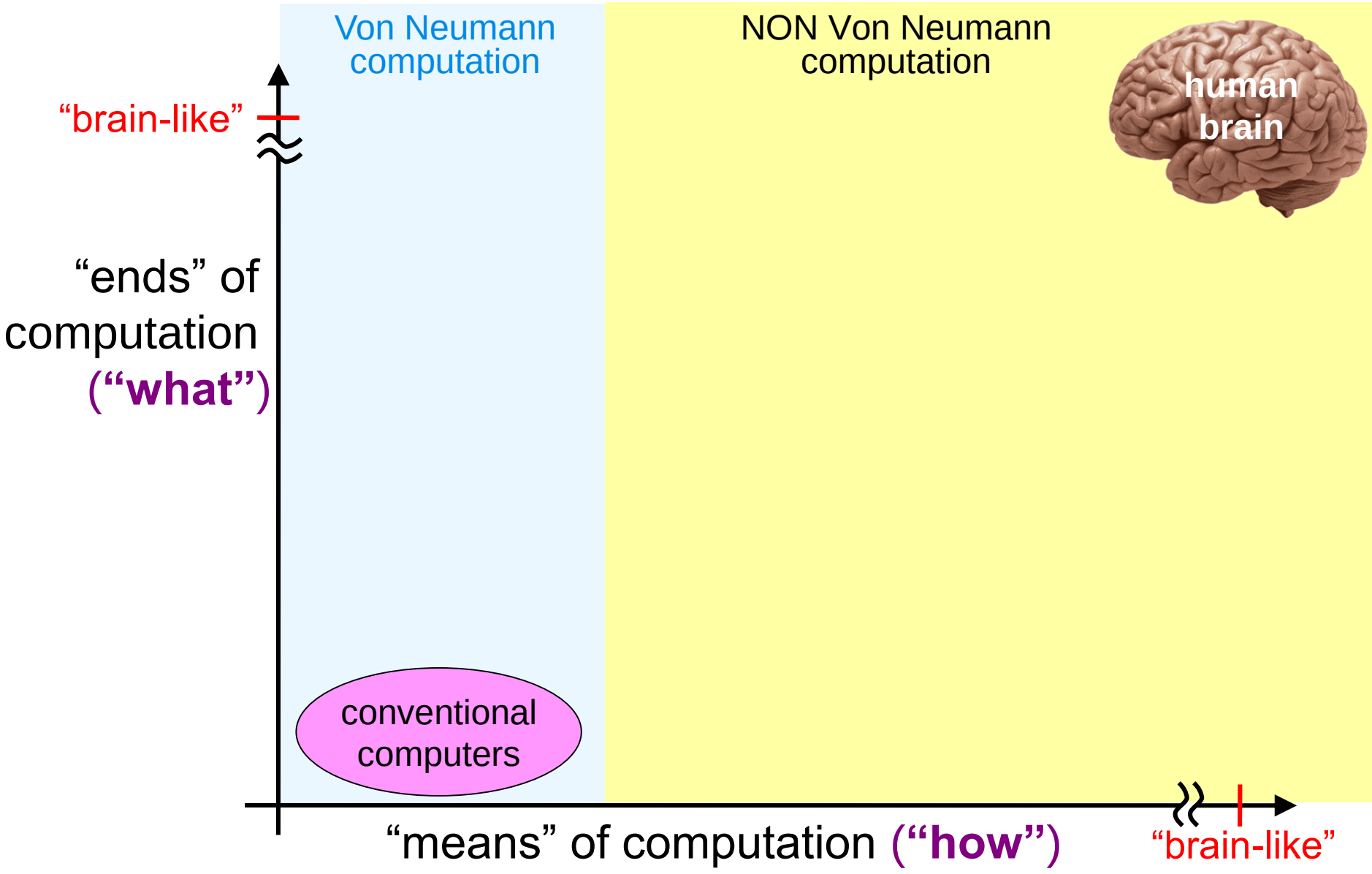


Outline for Part II

- Motivation
  - Look to the brain now that Dennard scaling is exhausted
- The Brain
  - What do we like about how the brain does computing?
  - (What DON'T we like about how the brain does computing?)
- The Computer Scientists
  - What are the computer scientists up to?
  - What might they be missing out on?
- Where can hardware (devices, circuits, systems) play a role, in …
  - ***Accelerating*** Deep Learning
    - TrueNorth, NVM-for-Backprop-Training
  - ***Transcending*** Deep Learning
    - Towards Brain-like energy-efficiency & "Machine Intelligence"
- Applications & outlook

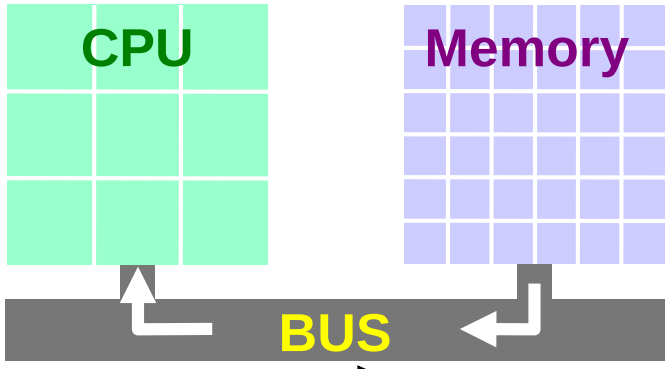
"Motivation:" Brains & light bulbs

Motivation: towards "Brain-like" computation



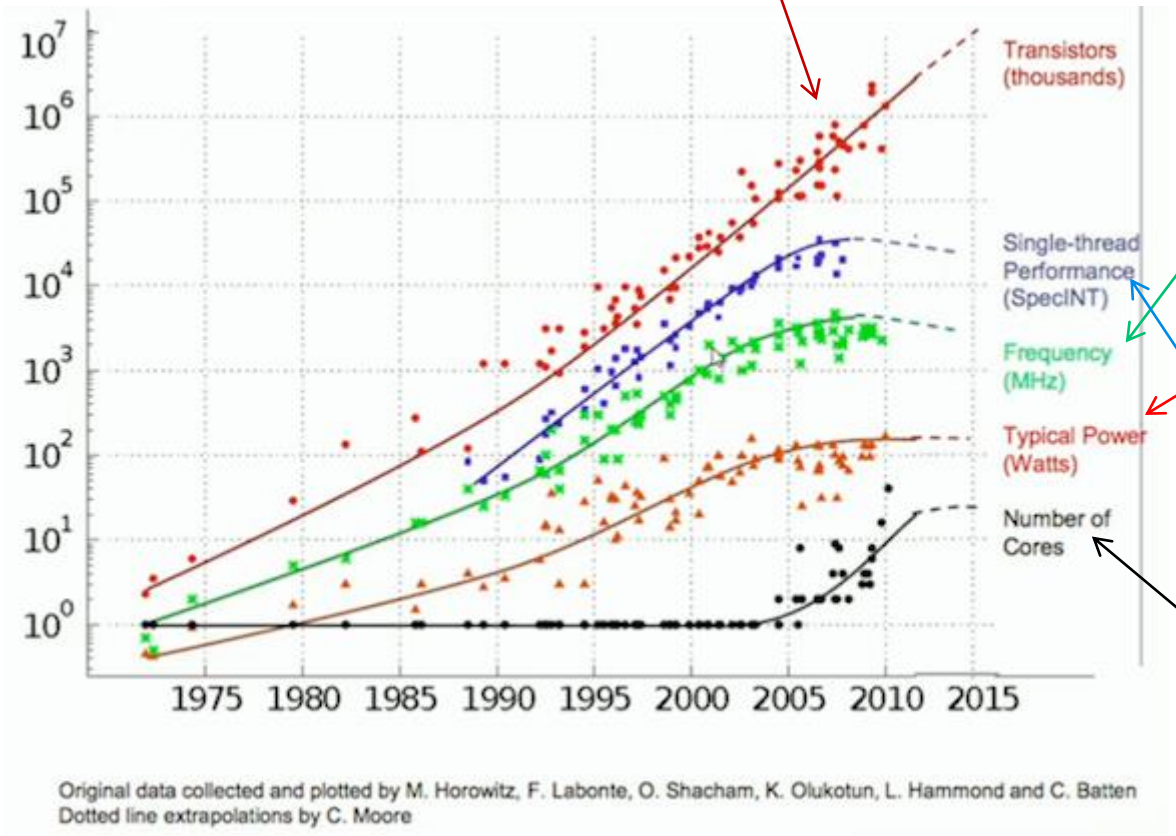
Von Neumann architecture

A "memory" delivers “operations” & “operands”
to a dedicated "central processing unit"



Von Neumann
“Bottleneck”

50 years of amazing progress thanks to
“Moore’s Law” (more transistors per chip)!



www.hpcwire.com/2013/12/11/hpc-progress-free-lunch/#/

But **clock frequencies** are no longer increasing…

… and **Chip power** has hit a ceiling.

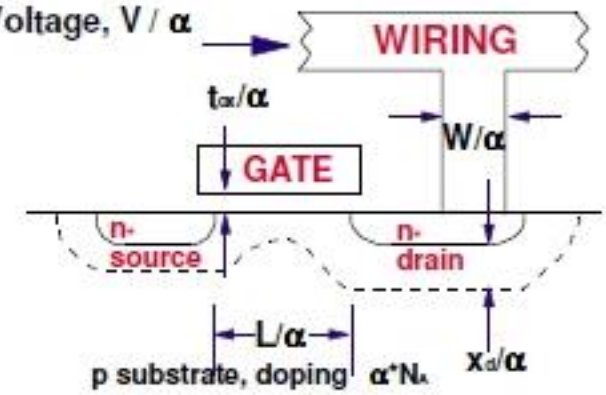
Since this means that **Single-thread performance** is not increasing…

… the only path to better *system* performance is adding more **Cores.**

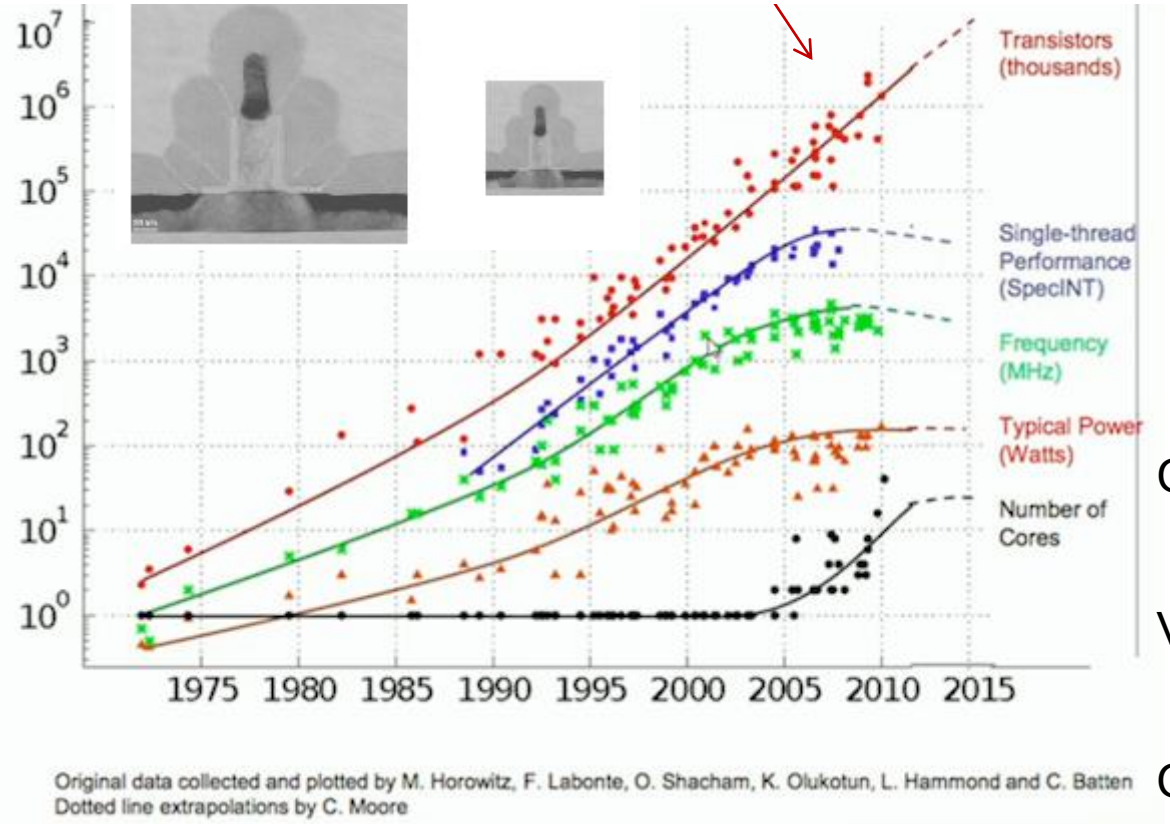
Dennard scaling

The real driver behind **"Moore's Law"**
(more transistors per chip)

Bob Dennard



We added more transistors by making them smaller…



…but those "scaled" transistors were actually **faster** AND required **less power**!! [1]

Unfortunately, Dennard scaling stopped working about 8-10 years ago…

Gate oxide scaling (t_{ox}) problems
→ **"high-K metal gate"**

Voltage scaling problems
→ **"new CMOS switch"**

OFF leakage problems
→ **"dark silicon"**

[1] IEEE Trans. Electr. Dev., **ED-31**(4), 452 (1984)

Adjusting our assumptions about computation

Conventional computing requires that ALL the devices work right.

This is getting difficult to guarantee across **billions** of devices
as **voltages** & **device-sizes** scale down.

Several recent trends in computing try to modify this assumption in different ways…

1) Quantum computing

- much more sensitive to noise (need low temperatures)
- …but much more functionality PER device (qubit)

2) Approximate/Stochastic computing – redundancy through design

3) Brain-inspired computing – redundancy through learning

a. Deep Neural Networks

- Apply "too many" resources to the problem yet get a result
- It can be "OK" if some of the resources are unreliable

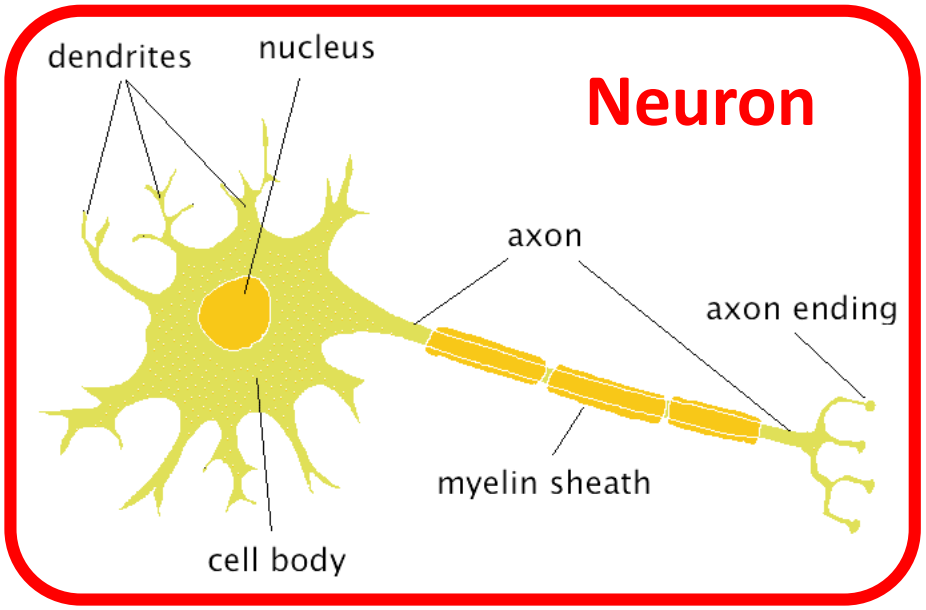
b. Even-more-Neuromorphic computing → "machine intelligence"

- Use **sparsity** in time & space to reduce overall computing power

Brain as an architecture for non-Von Neumann computation

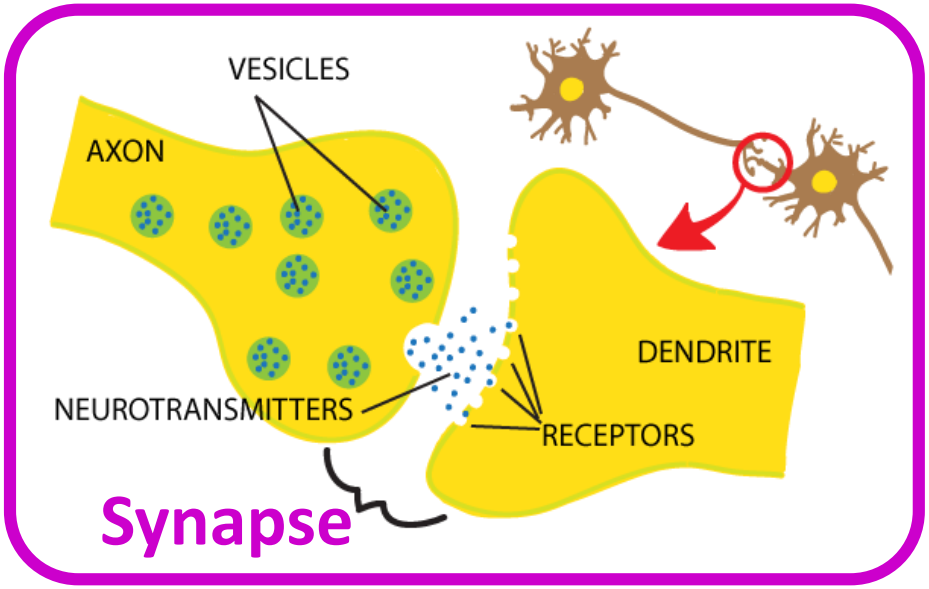
Some of the brain's **energy efficiency** comes from the biochemistry involved in signal transmission.

100 billion **neurons**, interconnected by 100 trillion **synapses**



But the **network architecture** is also incredibly efficient:

only a small fraction of neurons are active at any given instant.



In nature → **complexity** is "free" …

… but **size**, **weight** & **power** are highly constrained

So the distributed computational architecture of the brain *evolved* to **maximize energy efficiency**

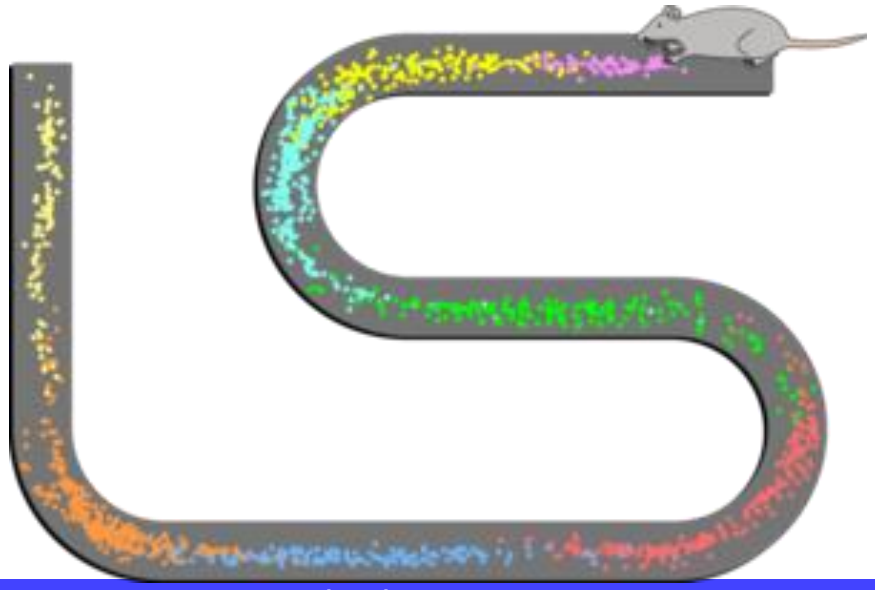
About the brain

- What **do we like** about how the brain does computing?
  - Low power
  - Massive parallelism
    - many processing elements (lots of neurons)
    - massive interconnectivity (large fanout: 1-10k synapses PER neuron)
  - Sparse Distributed Representations → massive capacity

Sparse Distributed Representations

- **Dense Representations**
  Few bits (8-128) Example: ASCII “m” = 01101101
  Efficient but no semantic meaning
- **Sparse Representations**
  Many bits (thousands), few 1’s, mostly 0’s
  Appears inefficient but evolution has picked it!
  Each bit has semantic meaning
- **Example of SDR uses: Union of Properties**
  Color 0000001000100000000110000000000100 (‘red)
  Shape 0000100010001010000010000000000000 (‘sphere’)

  Union 0000101010101010000110000000000100 (‘red sphere’)



Spatial firing patterns of 8 place cells recorded from the CA1 layer of a rat. The rat ran back and forth along an elevated track, stopping at each end to eat a small food reward. Dots indicate positions where action potentials were recorded, with color indicating which neuron emitted that action potential.

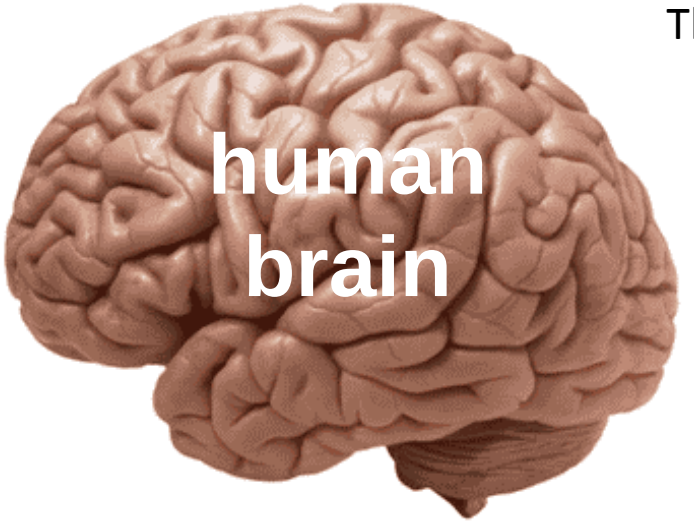
en.wikipedia.org/wiki/Place_cell

About the brain (continued)

- What **do we like** about how the brain does computing?
  - Low power
  - Massive parallelism
    - many processing elements (lots of neurons)
    - massive interconnectivity (large fanout: 1-10k synapses PER neuron)
  - Sparse Distributed Representations → massive capacity
  - Time is really important
  - Computing despite …
    - noise
    - unreliable & stochastic components
  - Makes rapid decisions despite uncertainty & incomplete information

- What **don't we like** about Von-Neumann architecture?
  - Bringing data TO processing is inefficient for data-centric workloads
    - would there be benefits by doing processing AT the data??
  - System has to be perfect (100% yield)
    - more and more difficult as scaling continues
    - are there ways to build systems that are still useful at 90-99% yield??
  - Dependence on software programming – labor-intensive → expensive!

Brain-inspired computing methods



The brain exhibits …

Electrochemical signal transmission

Hierarchy (multiple layers)

Local inhibition

Stochastic behavior

Recurrent connectivity

Local AND remote connectivity

Lots of neurons

Lots of synapses PER neuron

Sparse-distributed representations

Hebbian synaptic plasticity (cells that fire together, wire together)

Integrate-and-fire in space (many spikes arriving in a vicinity)

Integrate-and-fire in time (many spikes over time)

What are the CS folks doing/ignoring?

- What are the computer scientists up to?
  - Design-based Big Data Analytics – not “neural”
  - Machine Learning
    - Deep Neural Networks
      - pros & cons here
    - Transcending DNN – LSTM, Recurrent NN, Reinforcement Learning
  - Machine Intelligence

A simple taxonomy of Cognitive Computing

Other Machine Learning
(Support Vector Machines, k-means, knowledge-graphs, etc.)

Deep-Neural Networks
(Deep Machine Learning w/ backpropagation)

Machine Intelligence

Machine Learning: solving a specific task on **labeled** data by defining & optimizing an objective function

Brain-inspired Computing

mature, not as scalable.

mature, scalable.

not yet mature.

same

requires **lots** of **labelled, static** data.

learn **efficiently** w/ **unlabeled, time-dependent** data

more **explainable & user-adjustable.**

… can't readily **explain its decisions.**

IBM Watson

Combination of **natural language processing** and "design-based" **Big Data analytics**

Jan 2011 – win at "Jeopardy!"

2014 – IBM Watson group

Engagement Advisor:
- more meaningful interactions with customers

Explorer:
- make sense of big data, providing **context, trends** and **relationships**

Discovery Advisor:
- accelerate research

Health care: Helping doctors **identify treatment options**

Finance: Helping planners **recommend better investments**
Transform customer experiences, financial analysis, risk management & compliance

Retail: Helping retailers **transform customer relationships**
Transform the shopping experience, merchandising and supply networks, sales operations

Public sector: Helping government **help** its **citizens**
Citizens' experience, policy & performance, public security

www.ibm.com/smarterplanet/us/en/ibmwatson/index.html

Cognitive Computing based on Deep Neural Networks

Systems that learn at scale, reason with purpose, and interact with humans naturally.

Image recognition:

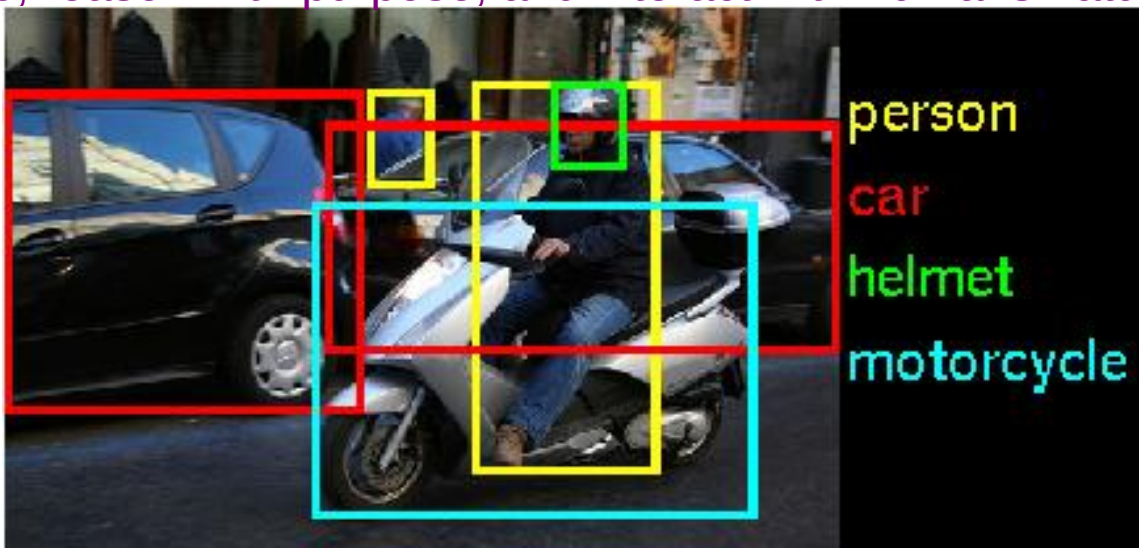


image-net.org/challenges/LSVRC/2014/

Speech recognition:

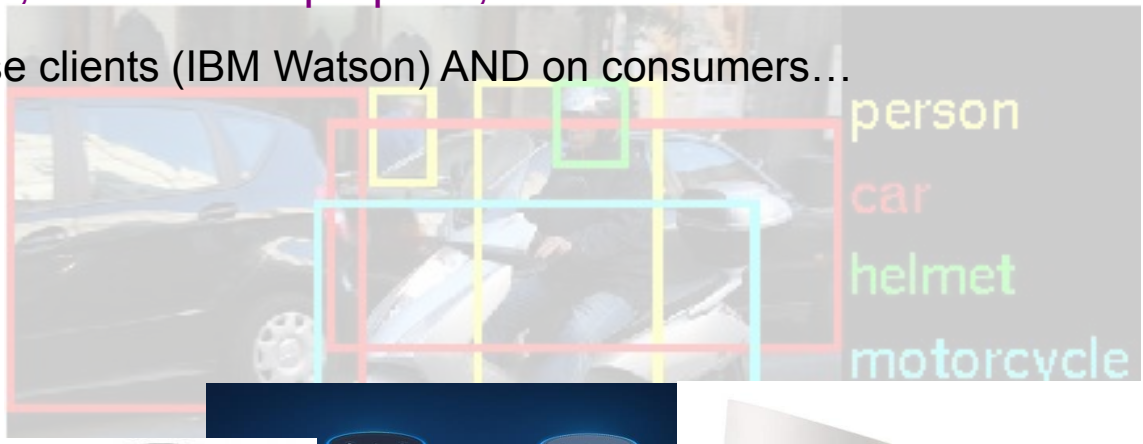
Machine translation:

Cognitive Computing based on Deep Neural Networks

Systems that learn at scale, reason with purpose, and interact with humans naturally.

→ Impact on enterprise clients (IBM Watson) AND on consumers…

Image recognition:



Speech recognition:

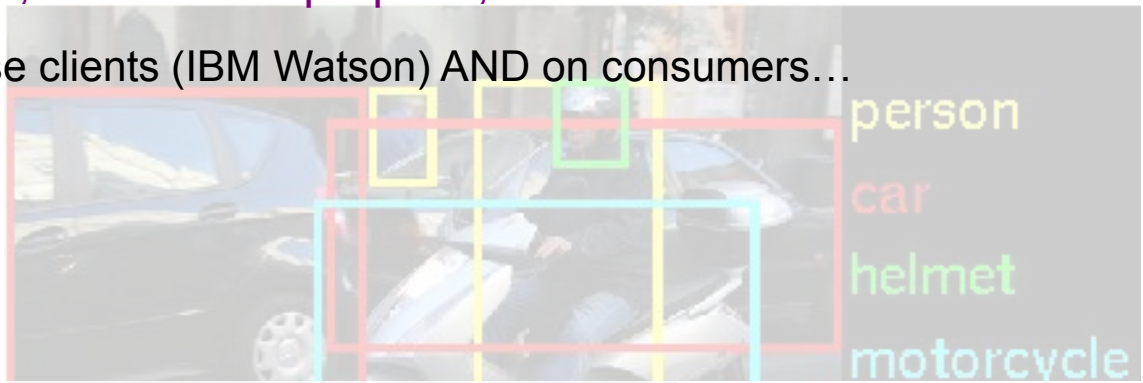
Machine translation:

Cognitive Computing based on Deep Neural Networks

Systems that learn at scale, reason with purpose, and interact with humans naturally.

→ Impact on enterprise clients (IBM Watson) AND on consumers…

Image recognition:



Speech recognition:

Machine translation:

Pre-2016: "One is not what is for what he writes, but for what he has read."

Uno no es lo que es por lo que escribe, sino por lo que ha leído

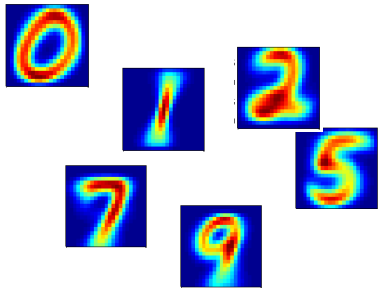
You are not what you write, but what you have read

당신은 당신이 쓰는 것이 아니라 당신이 읽은 것입니다.

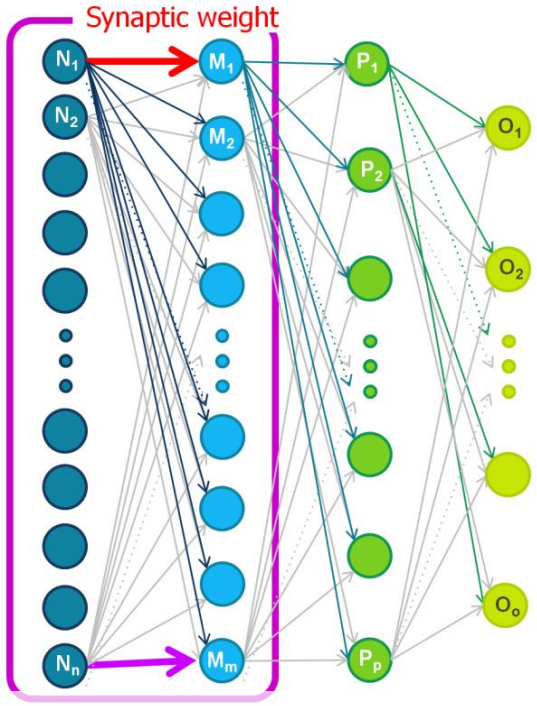
www.nytimes.com/2016/12/14/magazine/the-great-ai-awakening.html

Deep Neural Networks

1) Input data (images, raw speech data, etc.)
input to neural network



"MNIST" database
~1998
→ check-reading ATMs



A Deep Neural Network contains
multiple **layers,** …
each layer containing many **neurons,** …
each neuron driven through many
synaptic **weight** connections
from other neurons.

2) classification results compared to labels

3) corrections "backpropagated" & all weights updated

Forward inference:

Fully trained network

Hardware opportunity: Efficient, **low-power** deplo

Training:

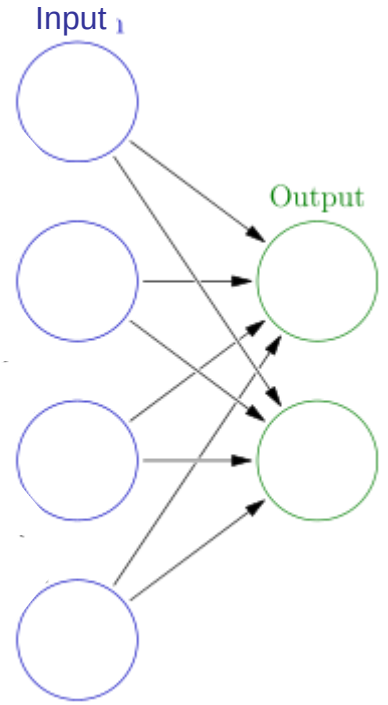
UN-trained network

"This is a **seven**."

Problem: It can take WEEKS to train these networks, even with many GPUs.

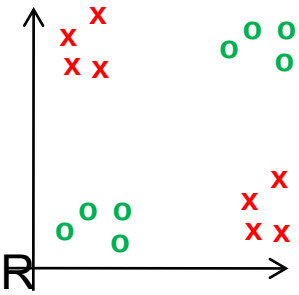
Hardware opportunity: Train big networks FASTER and at LOWER POWER.

Artificial Neural Networks
A brief history:



1943 – McCullough & Pitts
→ mathematical representation of a neural network

late 1940s – **Hebbian** learning
→ "cells that fire together, wire together"



1958 – **Perceptron** network

1969 – Minsky & Pappert paper
→ 1-layer perceptron can't solve XOR

1986 – Rumelhart & Hinton popularize **backpropagation**

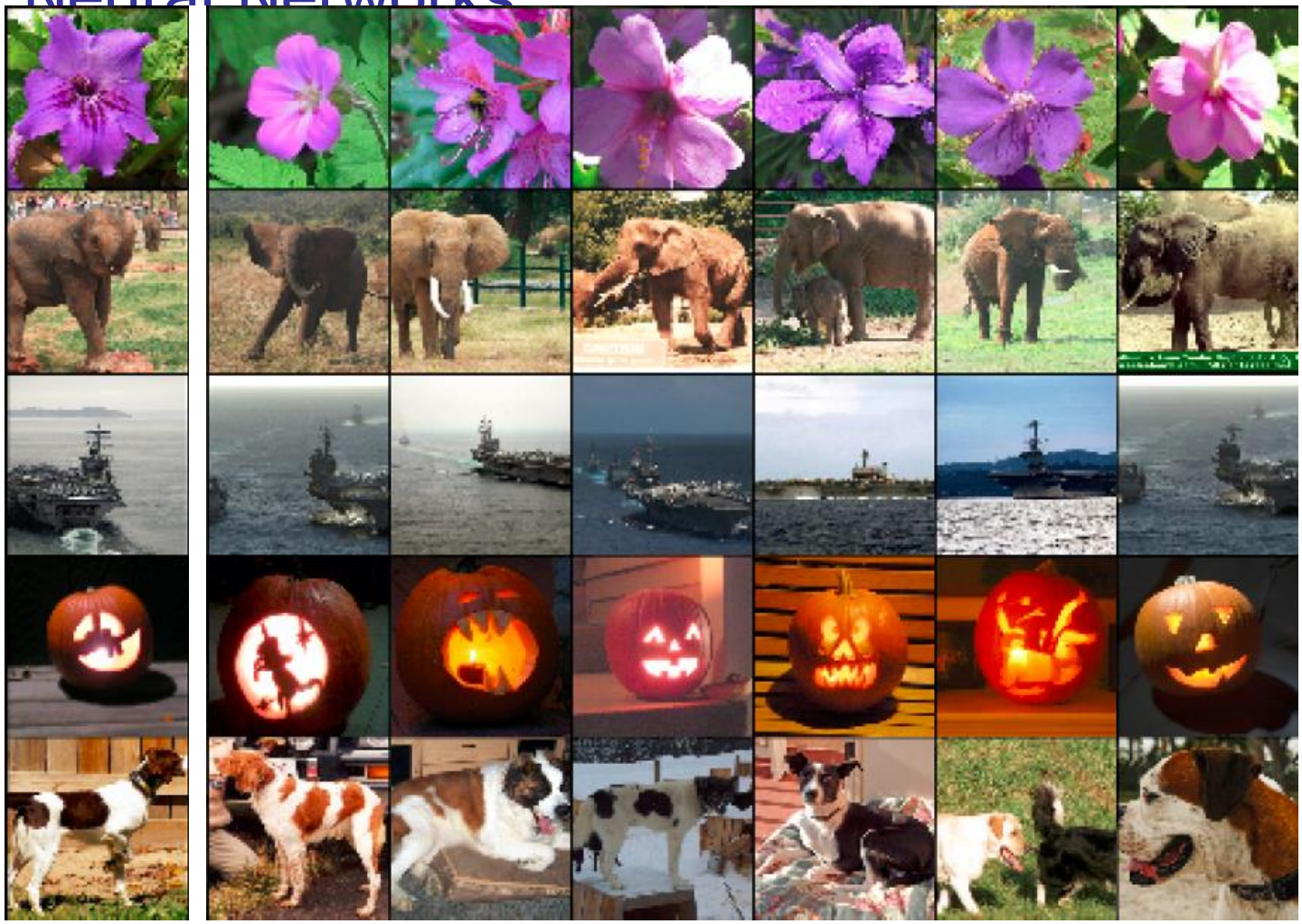
1990's – multilayer backprop. is too slow
→ rise of **statistical machine learning** (support vector machines, etc.)

2006 – "**deep" neural networks** – layer-by-layer "greedy" training

late 2000's – availabilty of **powerful GPUs** (Graphics Processing Unit)

2012 – "**ImageNet Classification** with Deep Convolutional Neural Networks,"
Krizhevsky, Sutskever, and Hinton, NIPS 2012

Artificial Neural Networks



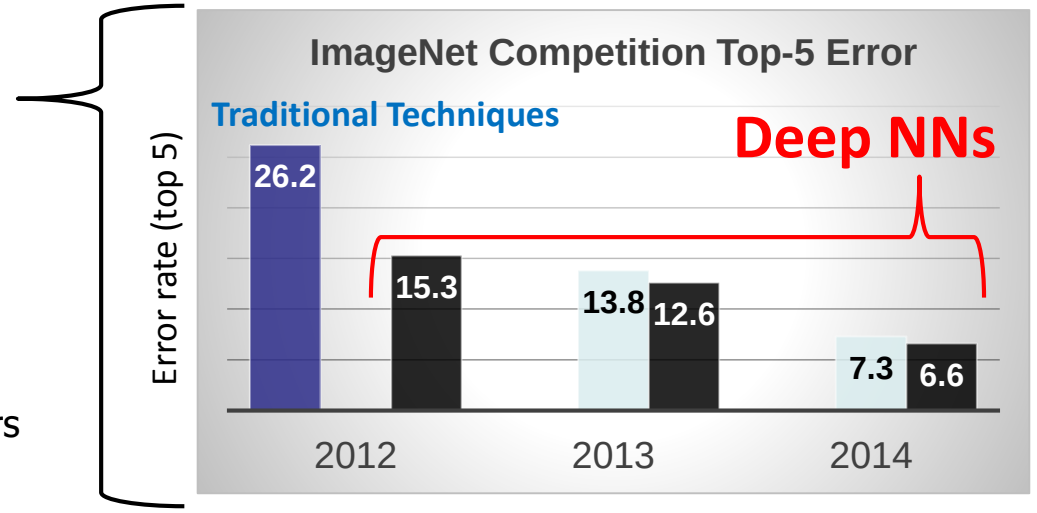
2006 – "**deep" neural networks** – layer-by-layer "greedy" training

late 2000's – availabilty of **powerful GPUs** (Graphics Processing Unit)

2012 – "**ImageNet Classification** with Deep Convolutional Neural Networks,"
Krizhevsky, Sutskever, and Hinton, NIPS 2012

Deep Neural Networks

- Strong impact on applications such as …
  - **Image recognition**
    - 1 billion parameters
  - **Speech recognition**
    - 40 million parameters
  - **(**compare **MNIST dataset)**
    - 100 thousand parameters



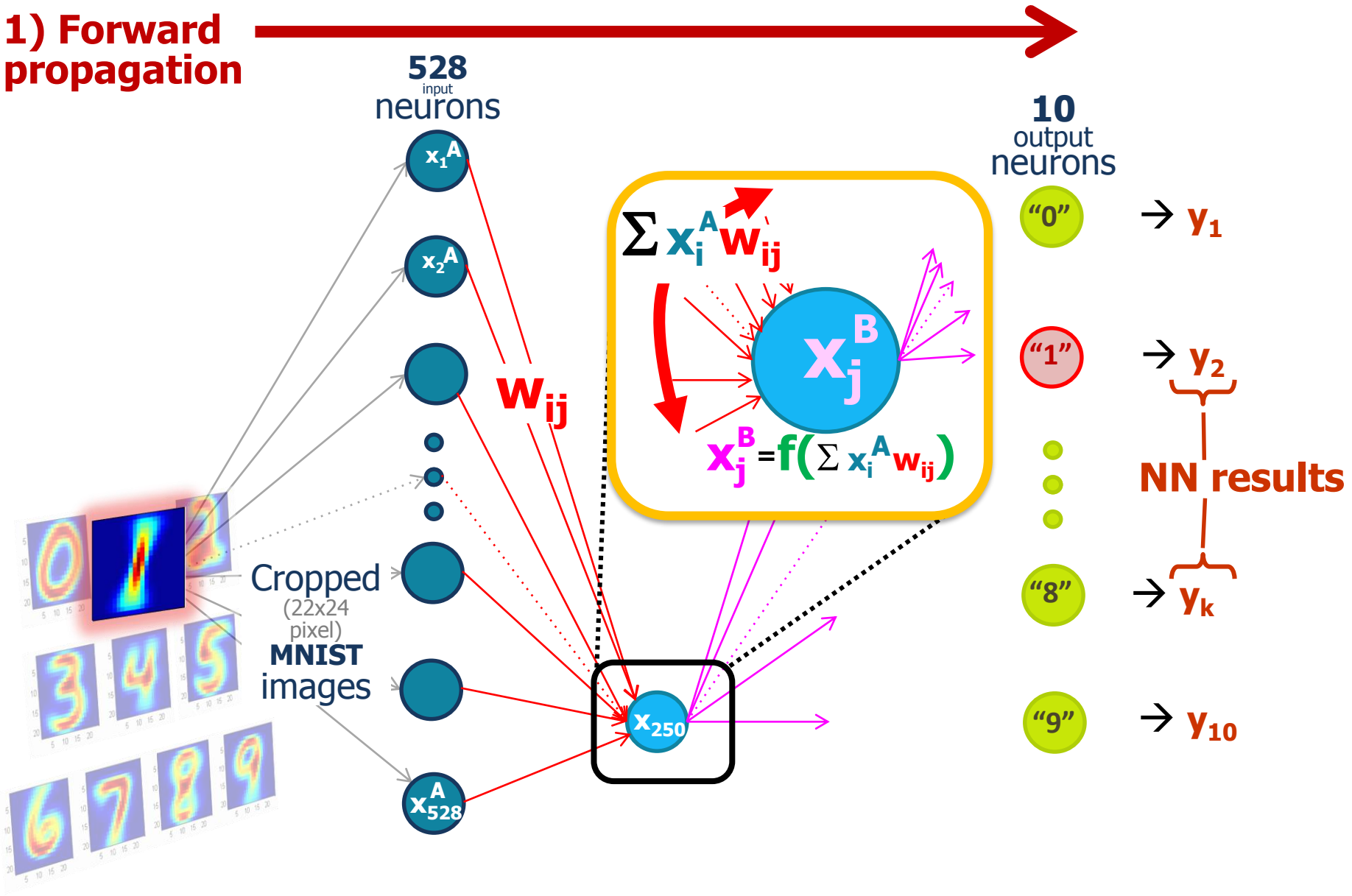
Step 1 - Training

- **Difficult** task (*Optimization*)
- Modify **adjustable parameters** (weights) in the model to match the input-output pairs for the training data.
- Takes **weeks** on many cores

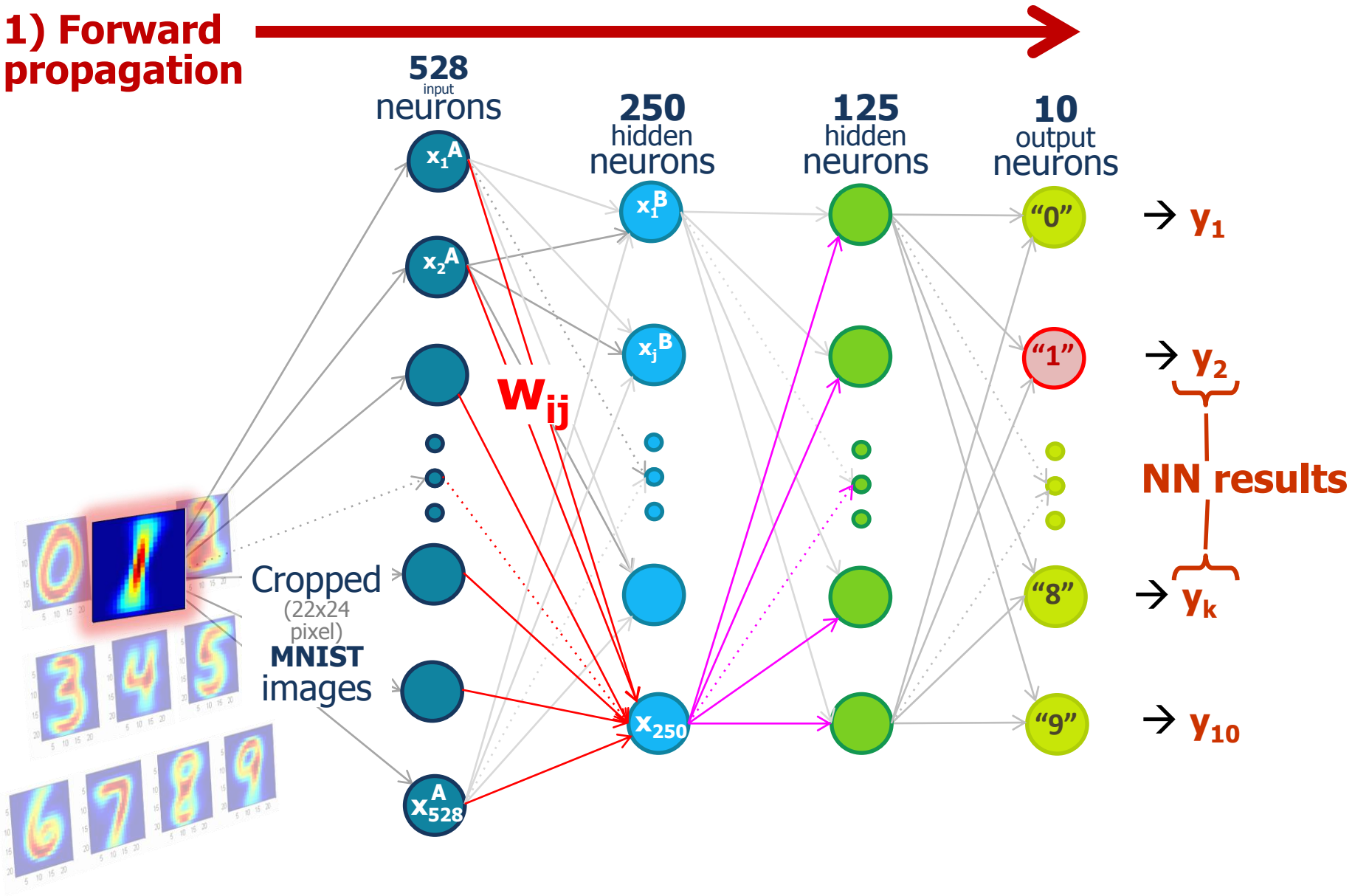
Step 2 - Execution

- Easier task (*Forward evaluation*)
- Given the input, generate the output (e.g., "**classify it**") using the trained model parameters (weights).
- Takes **milliseconds** on a single core

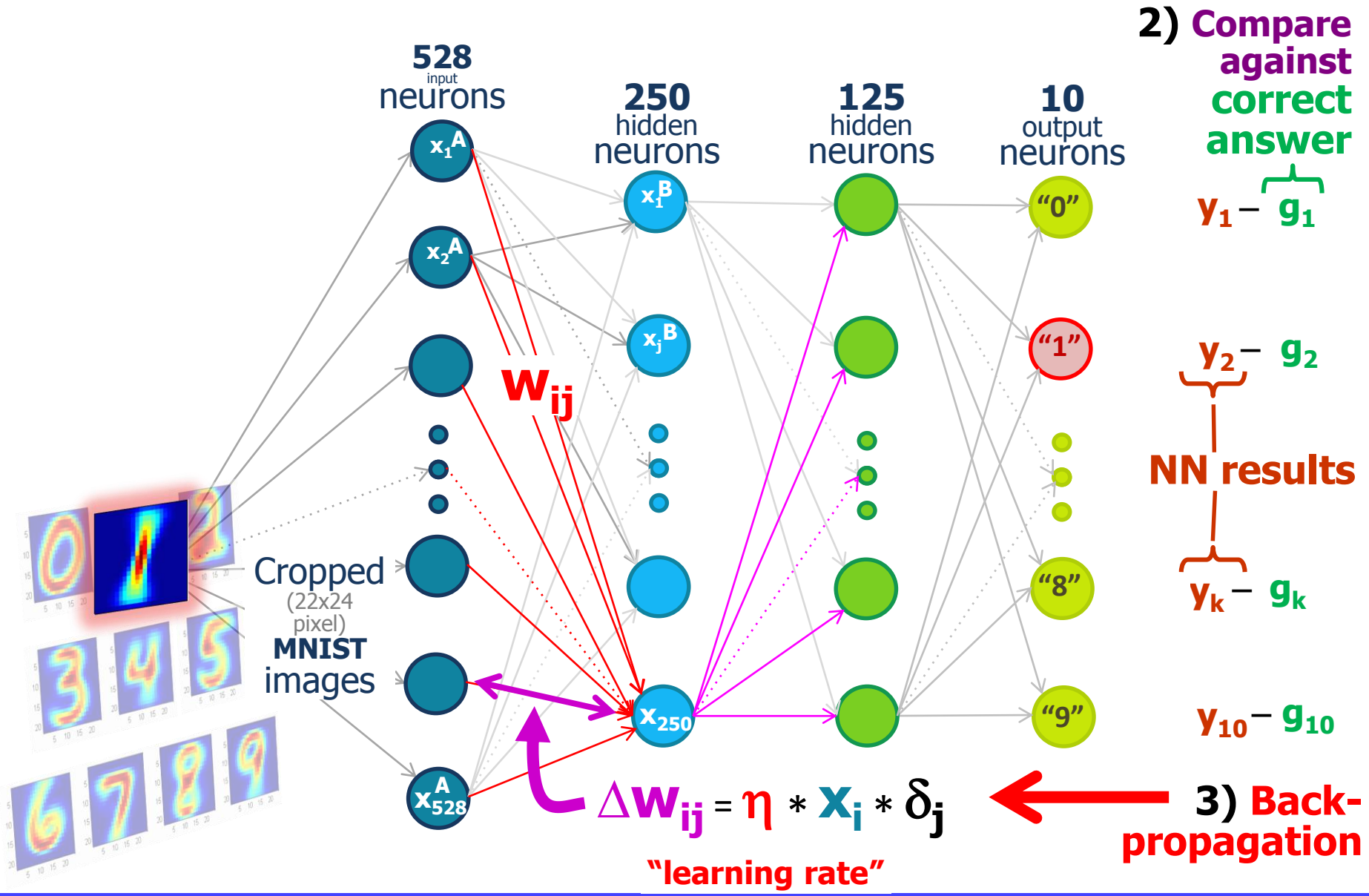
What is backpropagation?



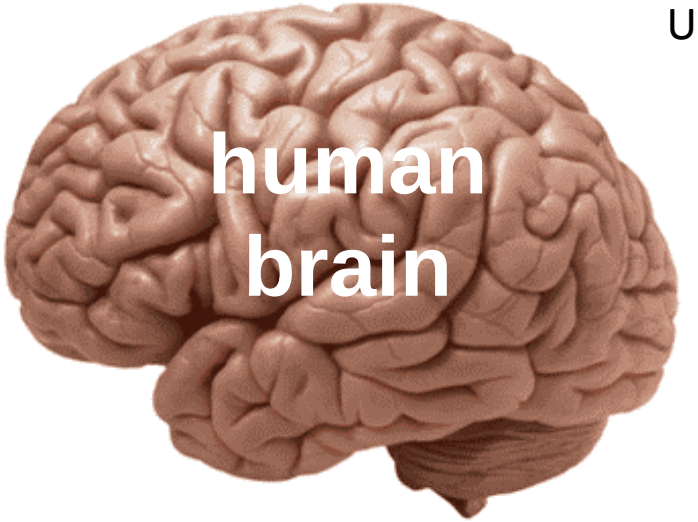
What is backpropagation?



What is backpropagation?



Deep Neural Networks



Uses PART of what the brain exhibits …

~~Electrochemical signal transmission~~

Hierarchy (multiple layers)

~~Local inhibition~~

Stochastic behavior (RBMs)

Recurrent connectivity (LSTMs, etc.)

Lots of neurons

Local ~~AND remote~~ connectivity

Lots of synapses PER neuron

~~Sparse distributed representations~~

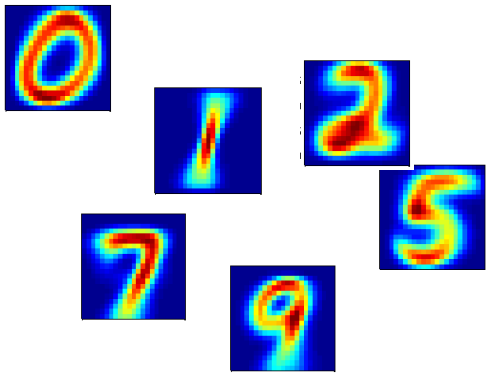
~~Hebbian~~ synaptic plasticity (~~cells that fire together, wire together~~) **Backprop**

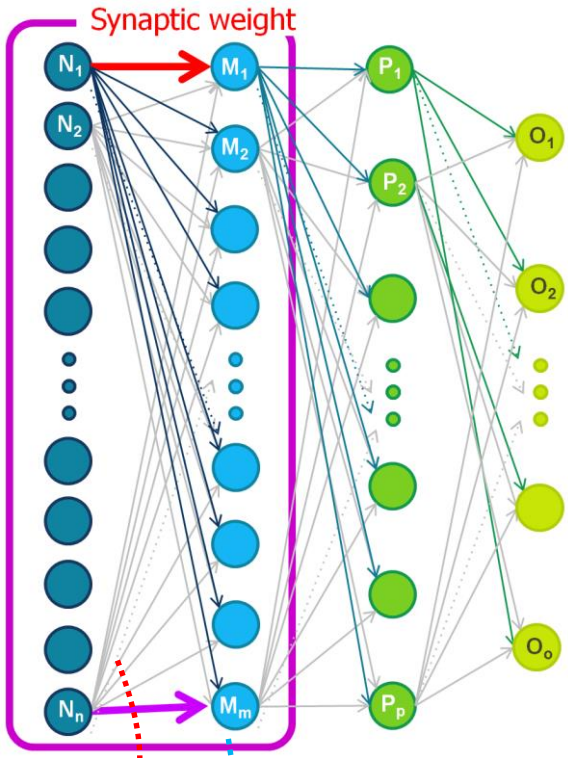
Integrate-and-fire in space (~~many spikes arriving in a vicinity~~)

~~Integrate-and-fire in time~~ (~~many spikes over time~~)

"Deep Learning" on GPUs

1) Input data (images, raw speech data, etc.) input to neural network





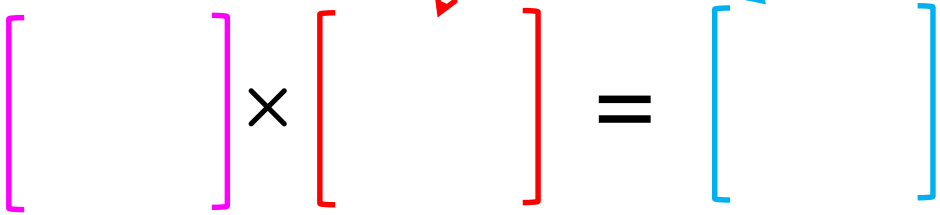
2) classification results compared to labels

3) corrections "backpropagated" & all weights updated

Combine 100-1000 input **vectors** into an input **matrix** ***("mini-batch")***

… multiply by current **weight matrix,**

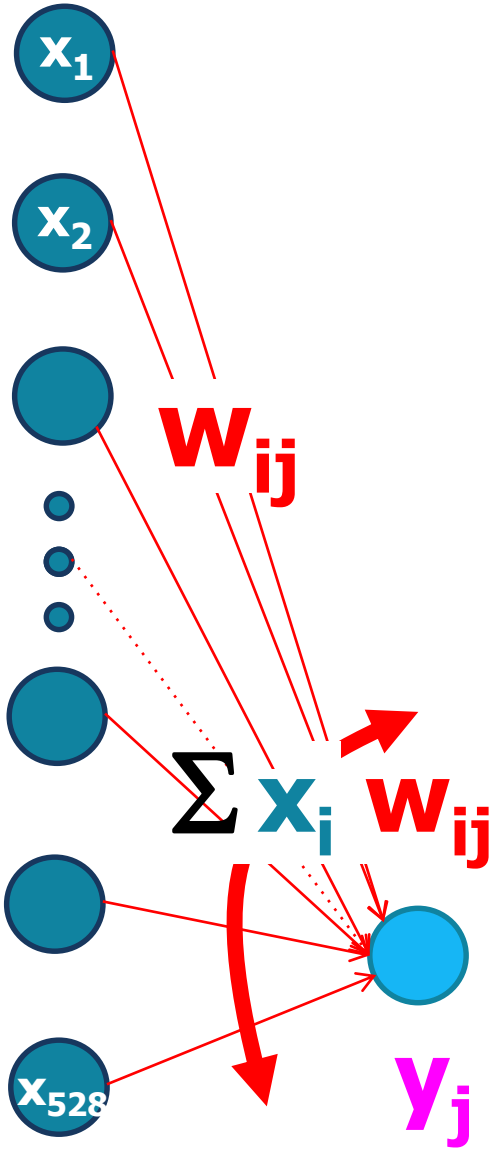
→ excitation into next **hidden neurons**



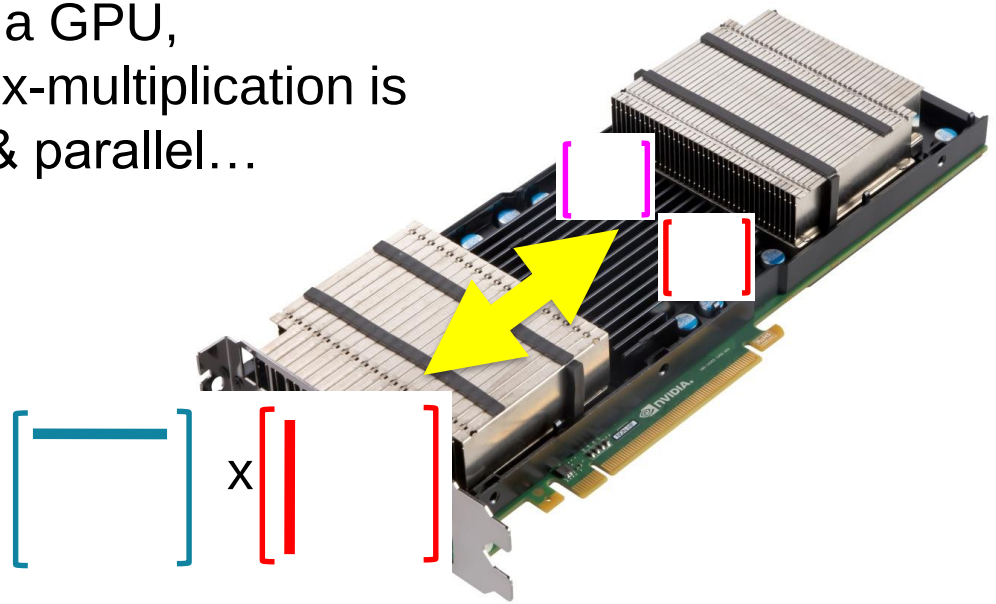
All steps can be mapped to matrix multiplications

→ can run very fast on GPUs

Computation needed: "Multiply-accumulate"



With a GPU,
matrix-multiplication is
fast & parallel…

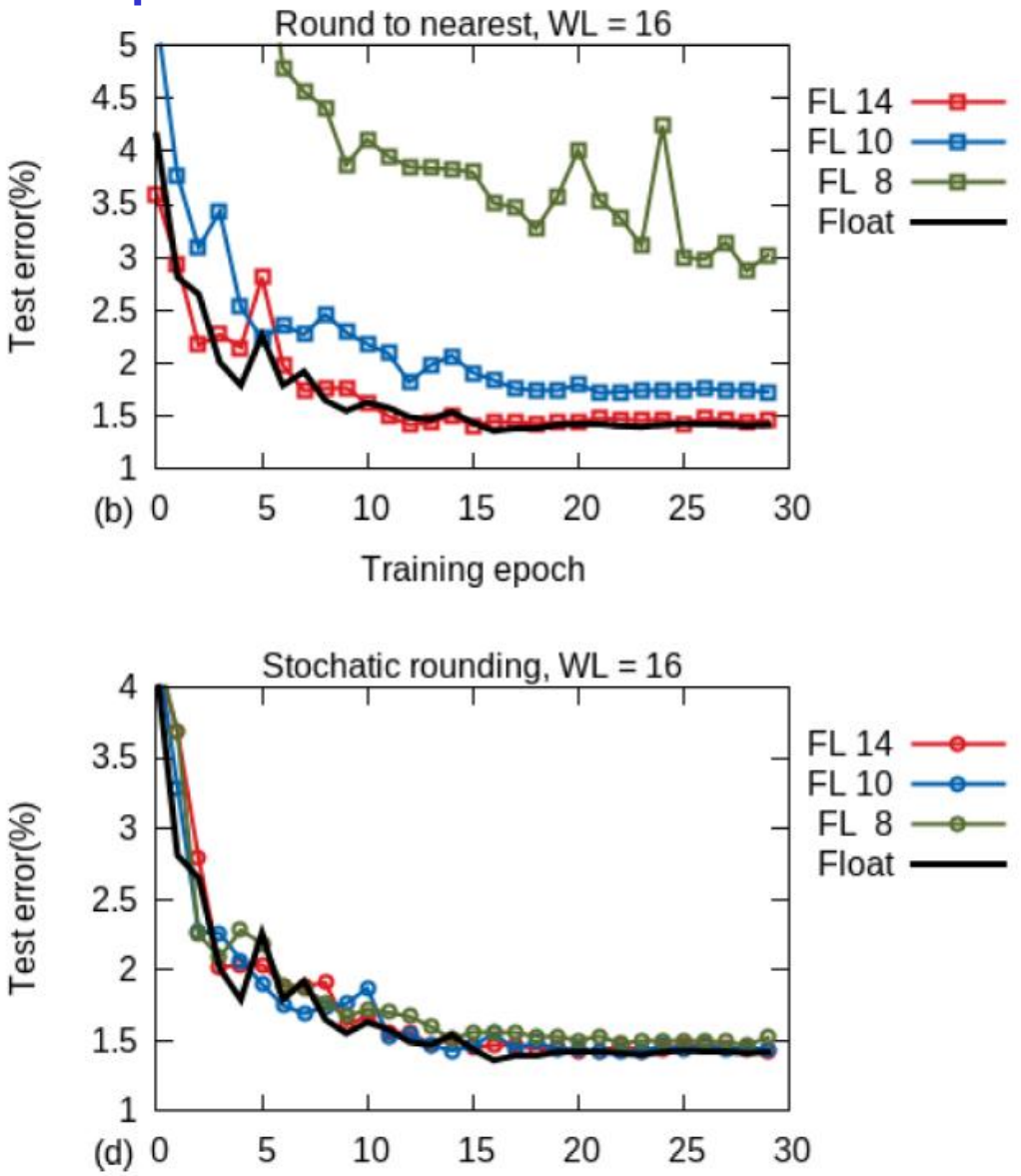


… but **x** and **w** values must arrive from DRAM,
and new **y** values sent back to DRAM

Reduced precision for Deep Neural Networks

• If you simply clip all the numbers during ANN training, you will lose out because your weight updates get smaller than the LSB you're using

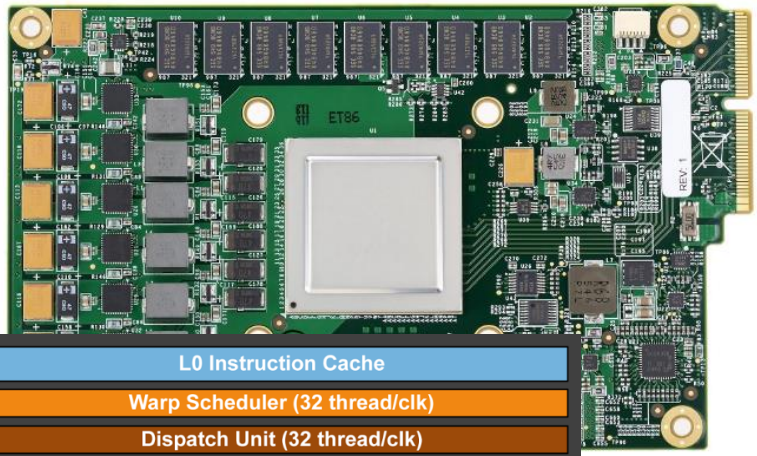
• Simple stochastic rounding at this LSB during ANN training can retain **ALL the performance** but with **many fewer bits of precision**



S. Gupta et al. "Deep learning with limited numerical precision." arXiv:1502.02551 (2015).

Reduced precision for Deep Neural Networks

Google TPU



rted in the slot
e Gen3 x16.

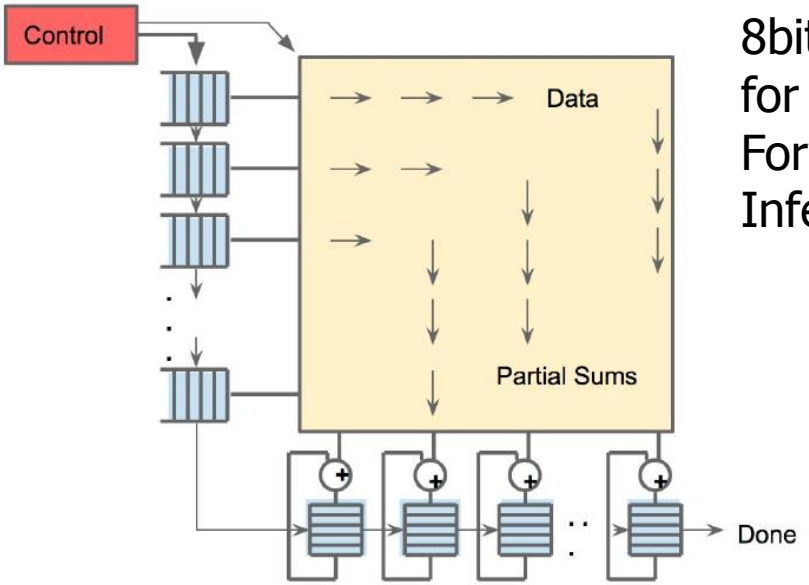


Figure 4. Systolic data flow of the Matrix Multiply Unit. Software has the illusion that each 256B input is read at once, and they instantly update one location of each of 256 accumulator RAMs.

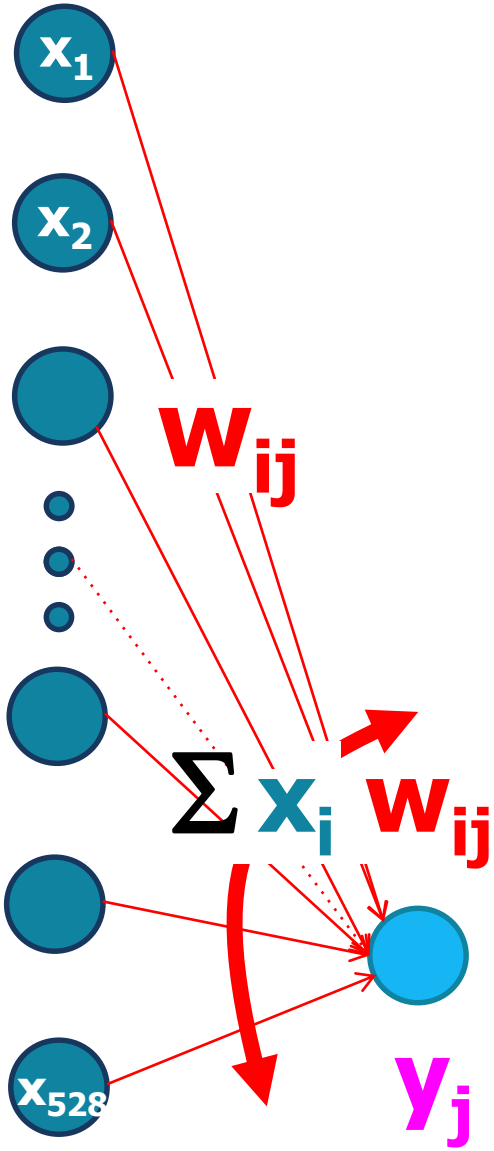
arxiv.org/abs/1704.04760

8bit INTs for Forward Inference

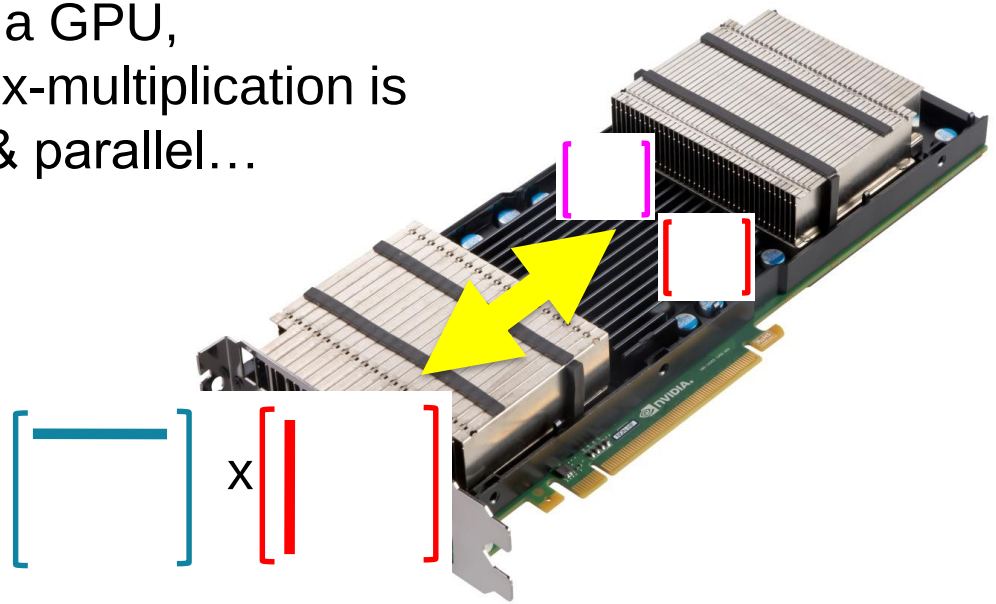
Nvidia Volta GPU

wccftech.com/nvidia-volta-gv100-gpu-tesla-v100-architecture-specifications-deep-dive/

Computation needed: "Multiply-accumulate"



With a GPU,
matrix-multiplication is
fast & parallel…

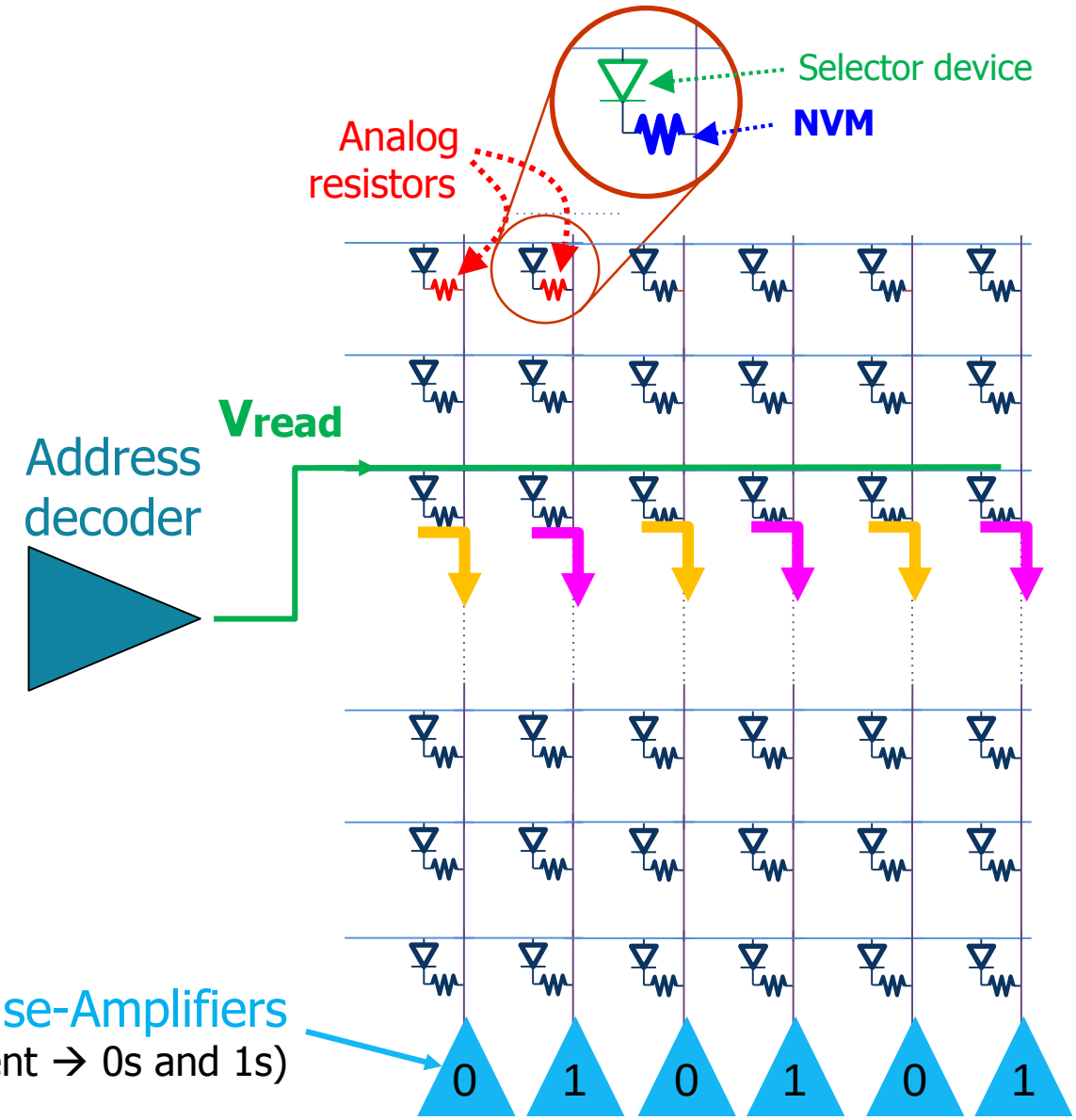


… but **x** and **w** values must arrive from DRAM,
and new **y** values sent back to DRAM

NVM (Non-Volatile Memory): usually for storing digital data (0s and 1s)

NVM technologies include:
MRAM (Magnetic RAM)
PCM (Phase-Change Memory)
RRAM (Resistance RAM)

Like conventional memory (SRAM/DRAM/Flash), an NVM is addressed one row at a time, to retrieve previously-stored digital data.



Multiply-accumulate with NVM: computed at the data, by physics

1) Different peripheral circuitry

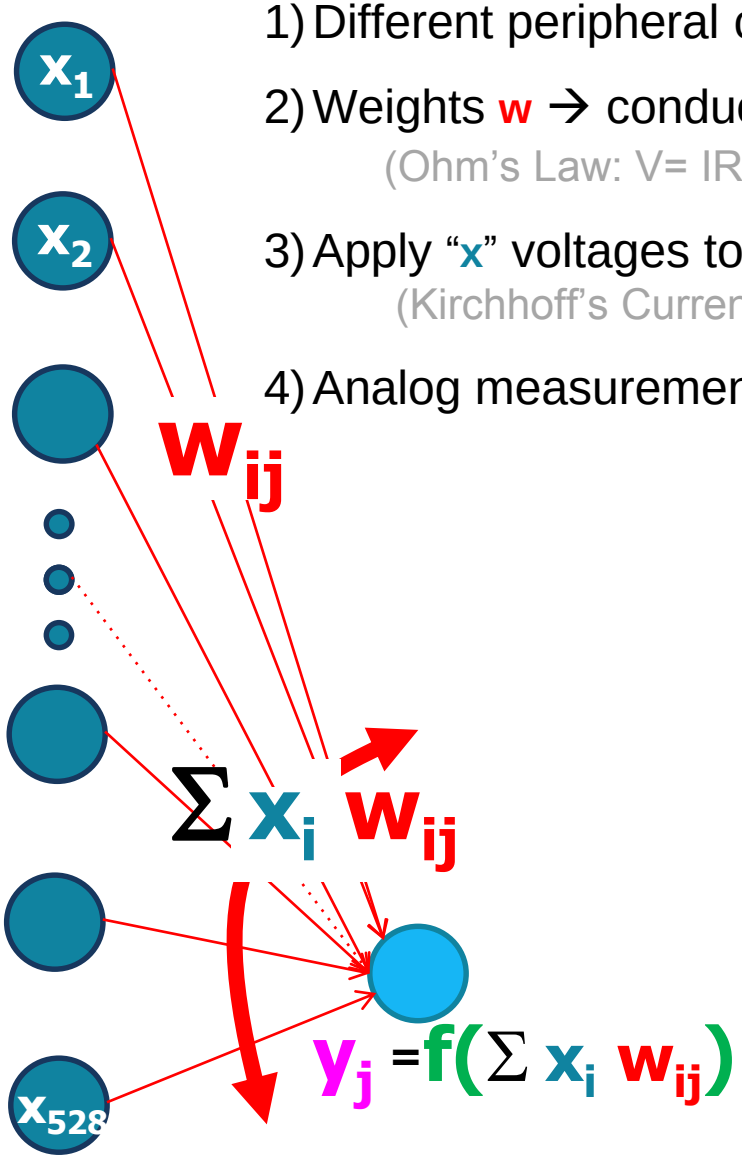
2) Weights **w** → conductances G^+, G^-

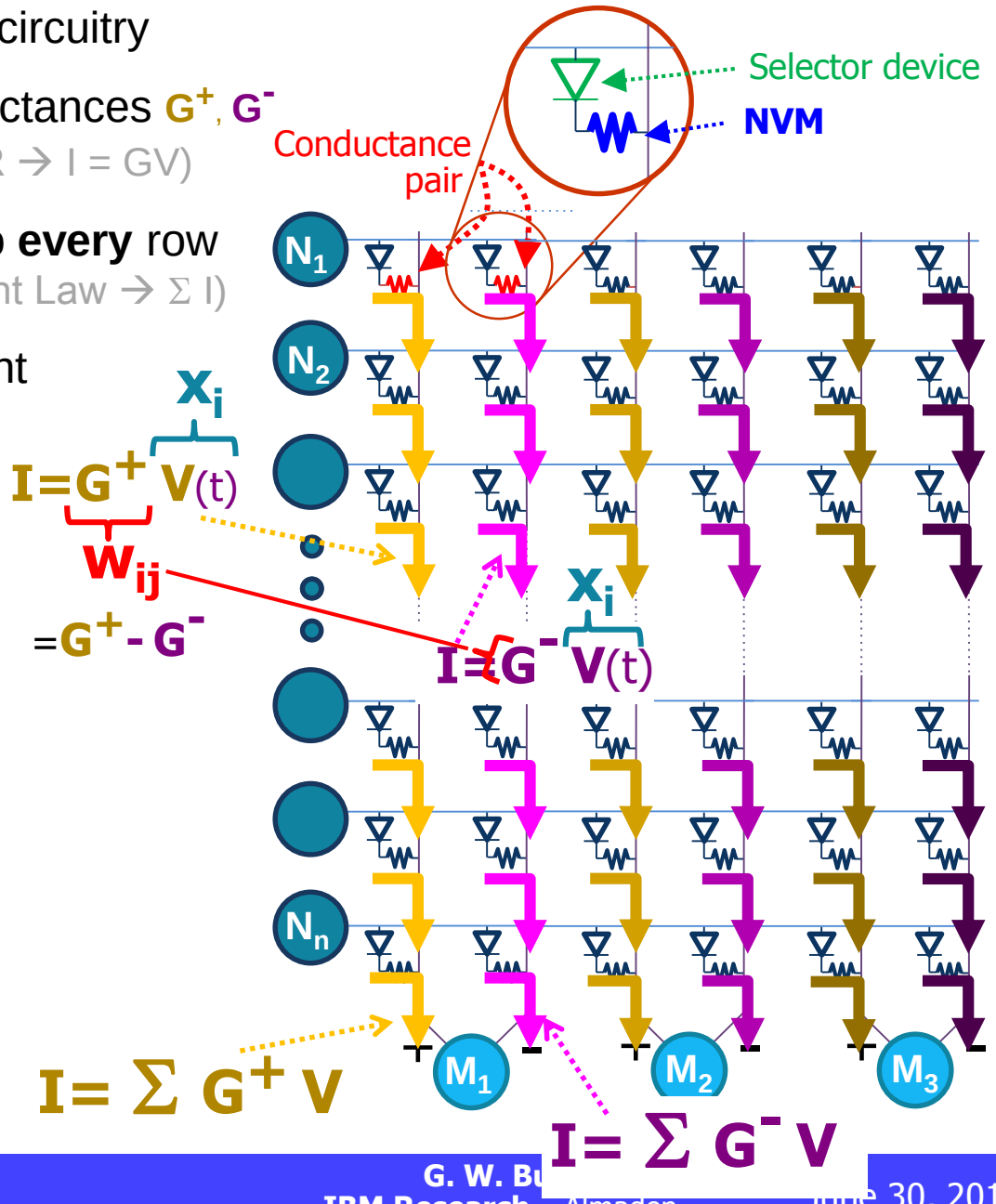
(Ohm's Law: V= IR → I = GV)

3) Apply "x" voltages to **every** row

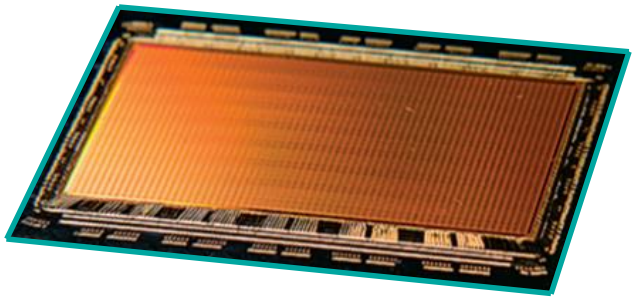
(Kirchhoff's Current Law → Σ I)

4) Analog measurement



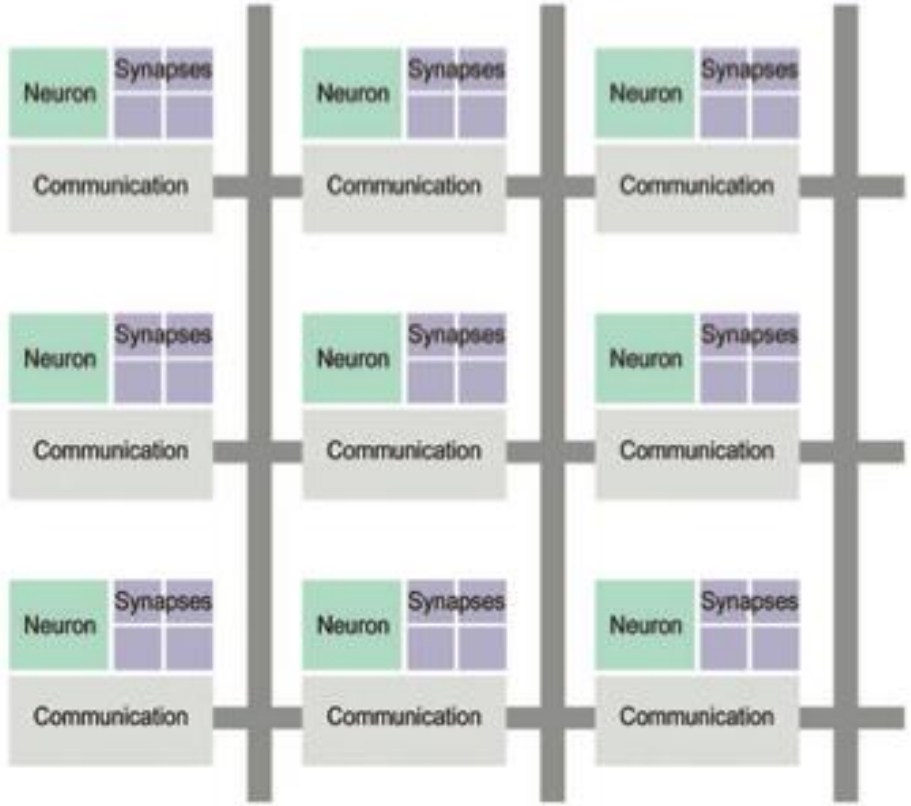


SyNAPSE project – the TrueNorth chip



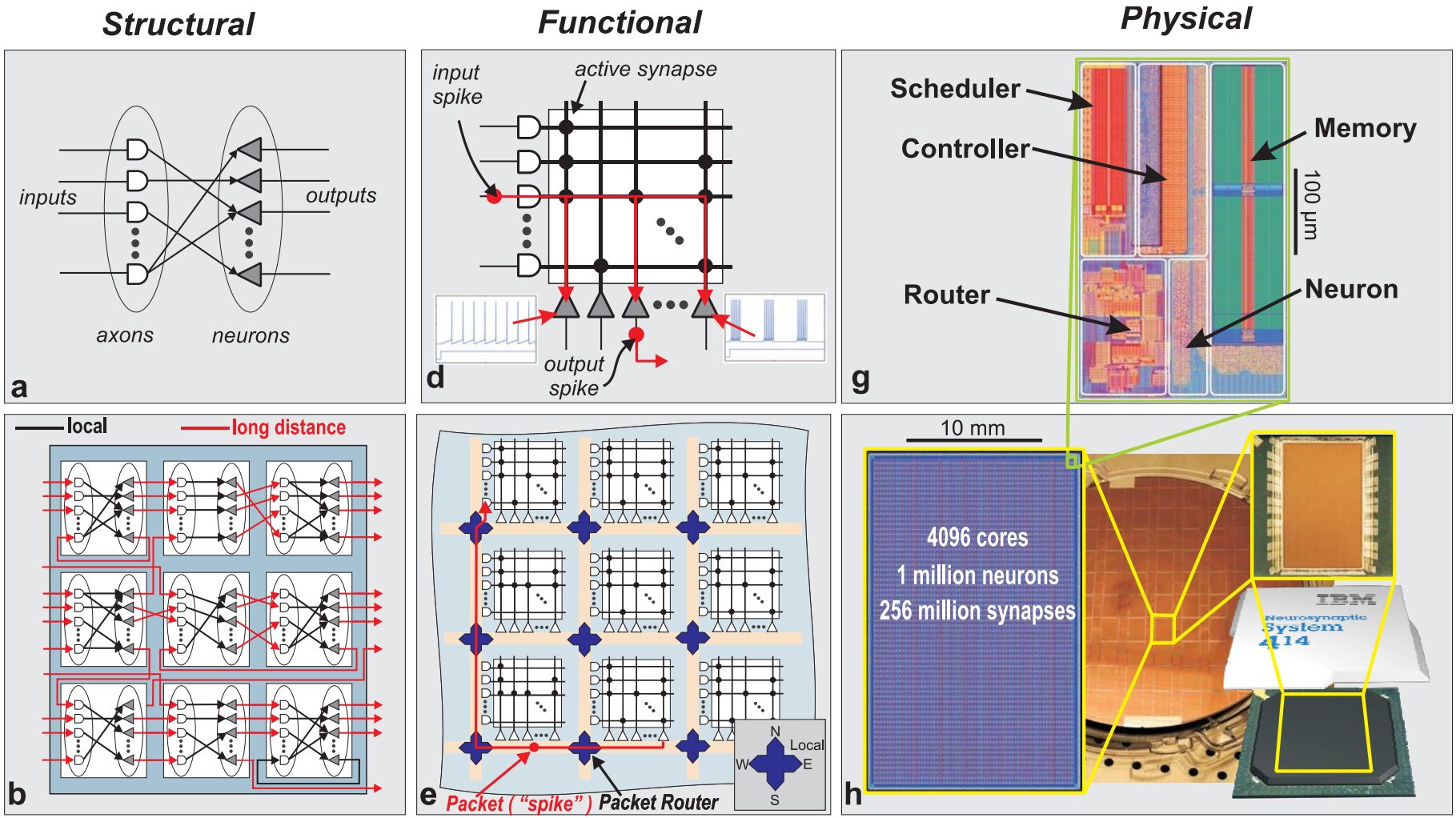
Modular network of lightweight cores
→ co-located computation, memory, & communication

Ultra-low-power execution of
pre-trained neural networks



Merolla et al., *Science*, **345**(6197), 668 (2014).

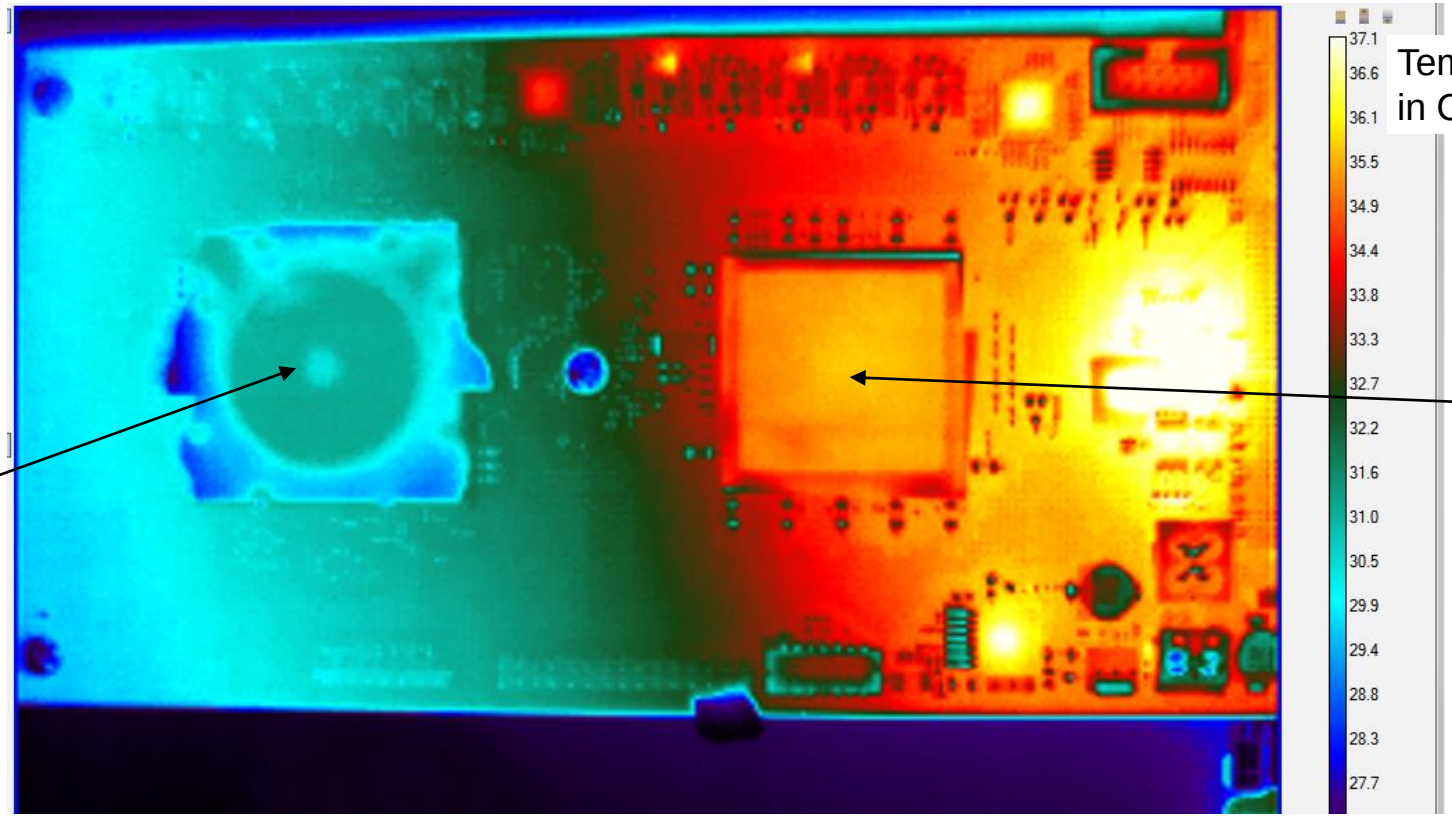
SyNAPSE project – the TrueNorth chip



~5,400,000,000 Transistors

Merolla et al., *Science*, **345**(6197), 668 (2014).

SyNAPSE project – power results



TrueNorth: 70mW total chip power, running a typical recurrent network at real-time

→ **26pJ energy per synaptic event** (128 synapses per neuron spiking at 20Hz)

- modern general-purpose microprocessor, optimized simulator (Compass) running the same network: consumes **176,000x more energy per event**
- state-of-the-art multiprocessor neuromorphic system (SpiNNaker), 48 chips each w/18 microprocessors, similar network: consumes **769x more energy per event**, requires **11.4x more silicon area**

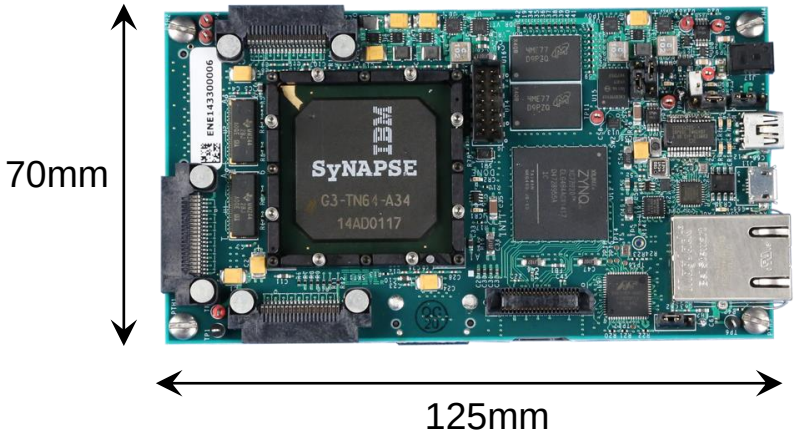
Merolla et al., *Science*, **345**(6197), 668 (2014).

SyNAPSE project – system variants

Mobile development platform

Miniature Form-Factor
Low Power, Low Weight
Real Time, User Friendly

< 100g



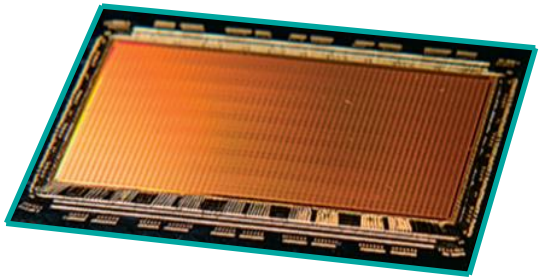
"SyNAPSE University"

1 week, on-site hands-on training

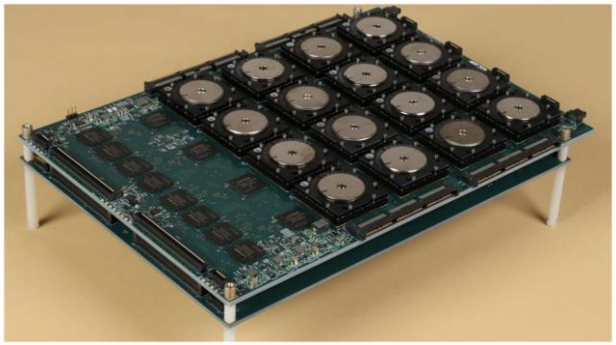
Ben Shaw
shawbe@us.ibm.com

IBM TrueNorth/SyNAPSE chip

Single Chip



16 Chip Board



SyNAPSE Supercomputer

Each **TrueNorth Chip**

- 4096 **"cores"**
  (of 256 axons x 256 neurons)
- 1 million neurons
- 256 million synapses
- 5.4 billion transistors
- **70mW power**

Very power efficient!!

Board with 16 chips

- 65536 **cores**
- 16 million neurons
- 4 billion synapses

Rack with 16 boards

- 1 million **cores**
- 256 million neurons
- 64 billion synapses

Only performs
forward evaluation of ANNs –
not ***training*****.**

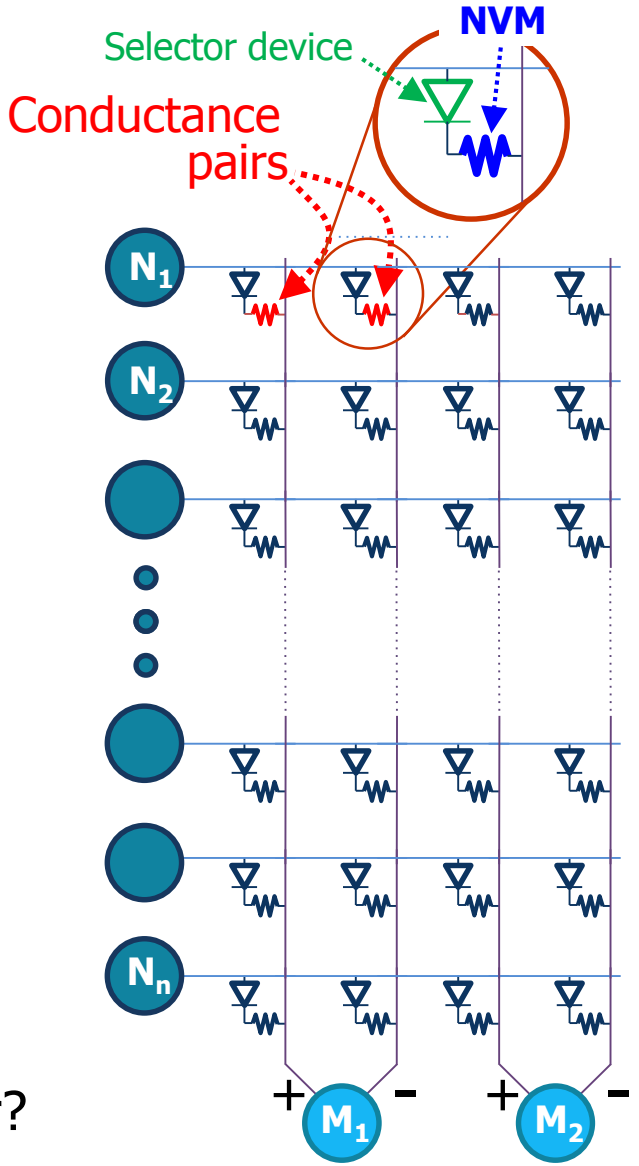
NVM-for-Machine-Learning

Like TrueNorth: compute **AT** the weight data

Unlike TrueNorth: learning performed **on-chip**

For TrueNorth, **power** is everything

For NVM-for-ML, need **speed-up** over GPUs



Research challenges

1) What do we really need from the NVM devices?
   - Recap of our *IEDM2014*, *IEEE-TED2015* work
     → Need competitive ML performance

2) What are the potential benefits, in speed & power?
   - Speed → Parallelism → Area-efficient circuits

Published work on "what do we need from the NVM?"

[1] **IEDM 2014**

Experimental demonstration and tolerancing of a large-scale neural network (165,000 synapses), using phase-change memory as the synaptic weight element

G. W. Burr, R. M. Shelby, C. di Nolfo, J. W. Jang‡, R. S. Shenoy, P. Narayanan, K. Virwani, E. U. Giacometti, B. Kurdi, and H. Hwang‡

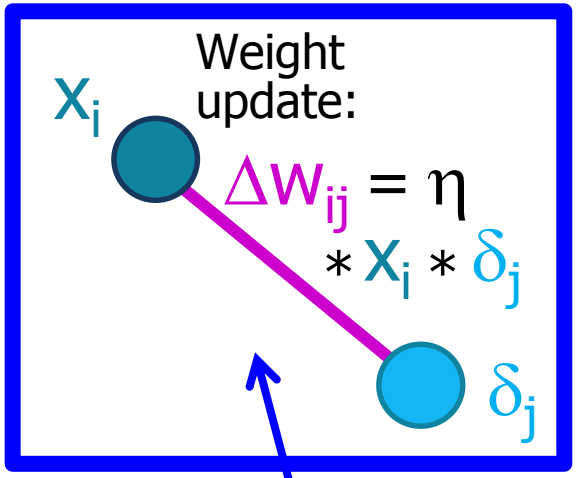
→ First large-scale mixed hardware-software demonstration + tolerancing

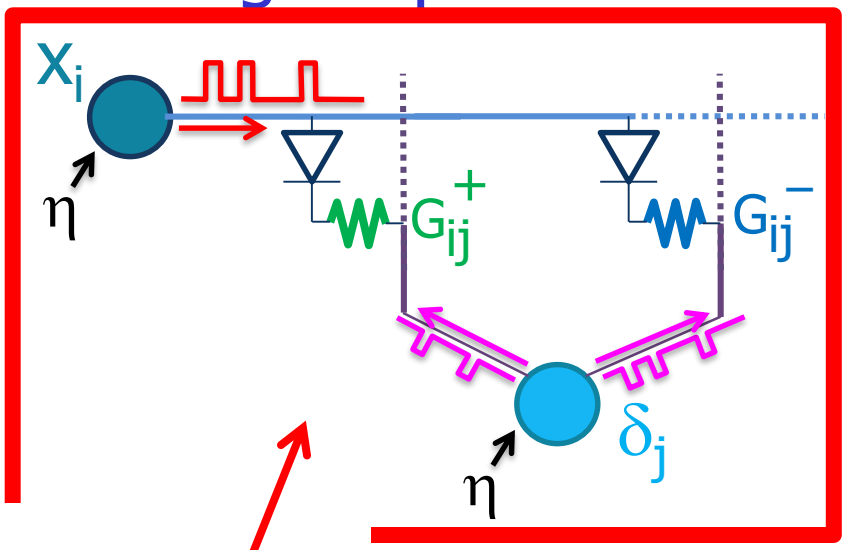
→ ~82% accuracy on MNIST with 5000 examples

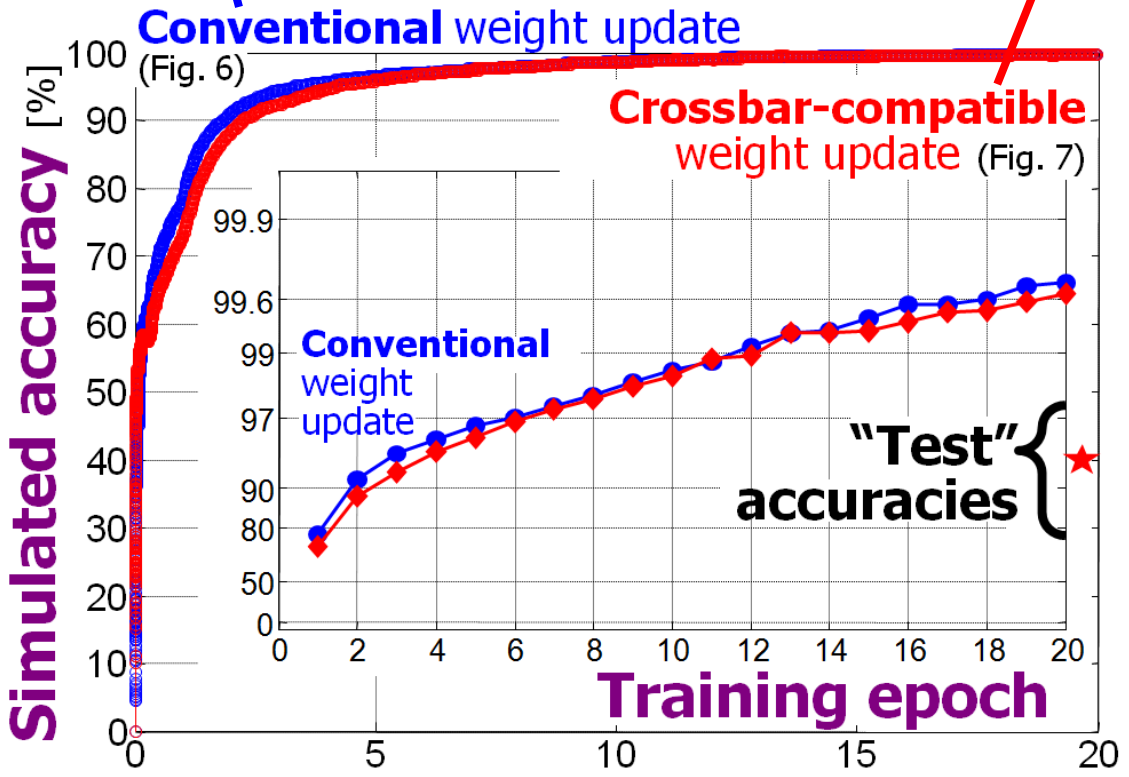
Introduced…

- **"crossbar-compatible"** weight-update
- **"G-diamonds"** – represent distribution of synaptic-states **graphically**

IEDM 2014: xbar-compatible weight-update



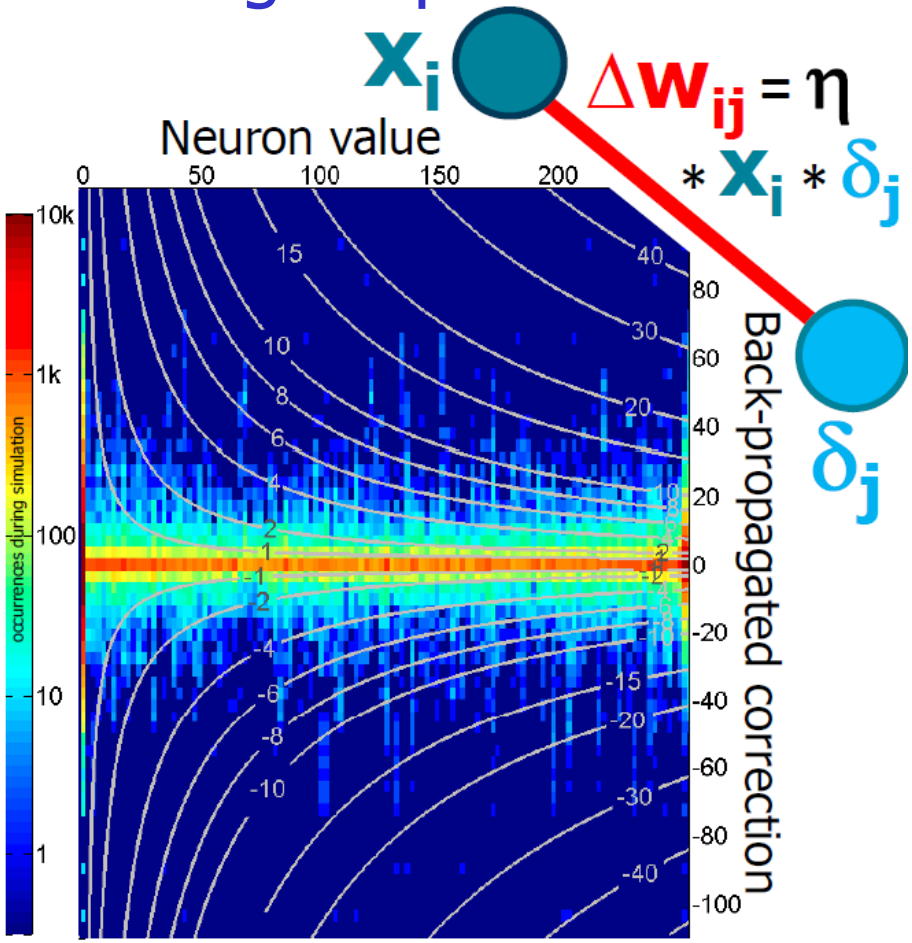




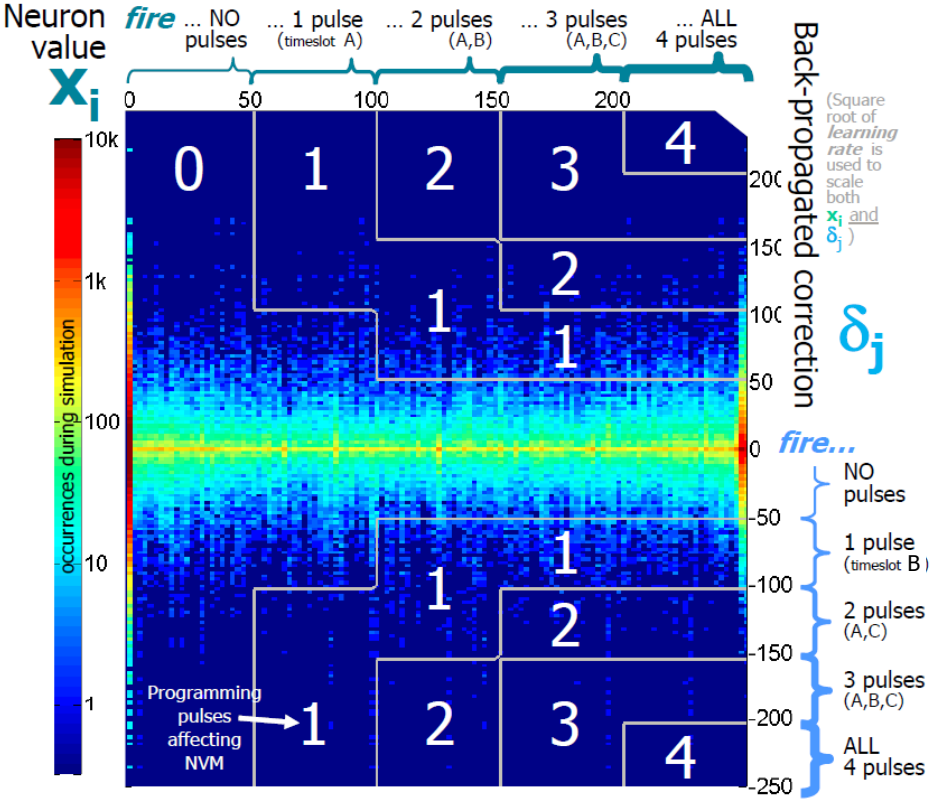
Weight update rule → not a problem!

[1] G. W. Burr, R. M. Shelby, et al., *IEDM Technical Digest,* 29.5, (2014).

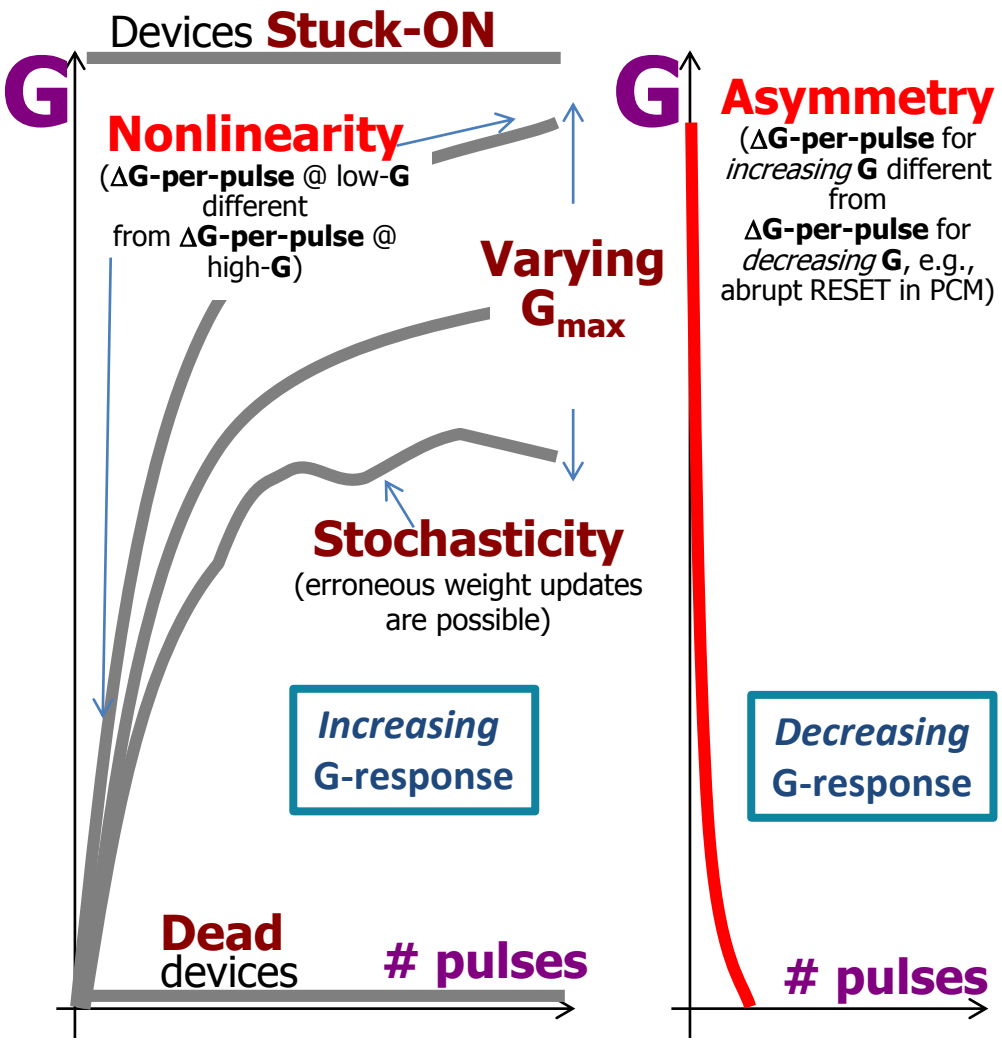
Conventional ("CS professor") weight update



Crossbar-compatible weight update



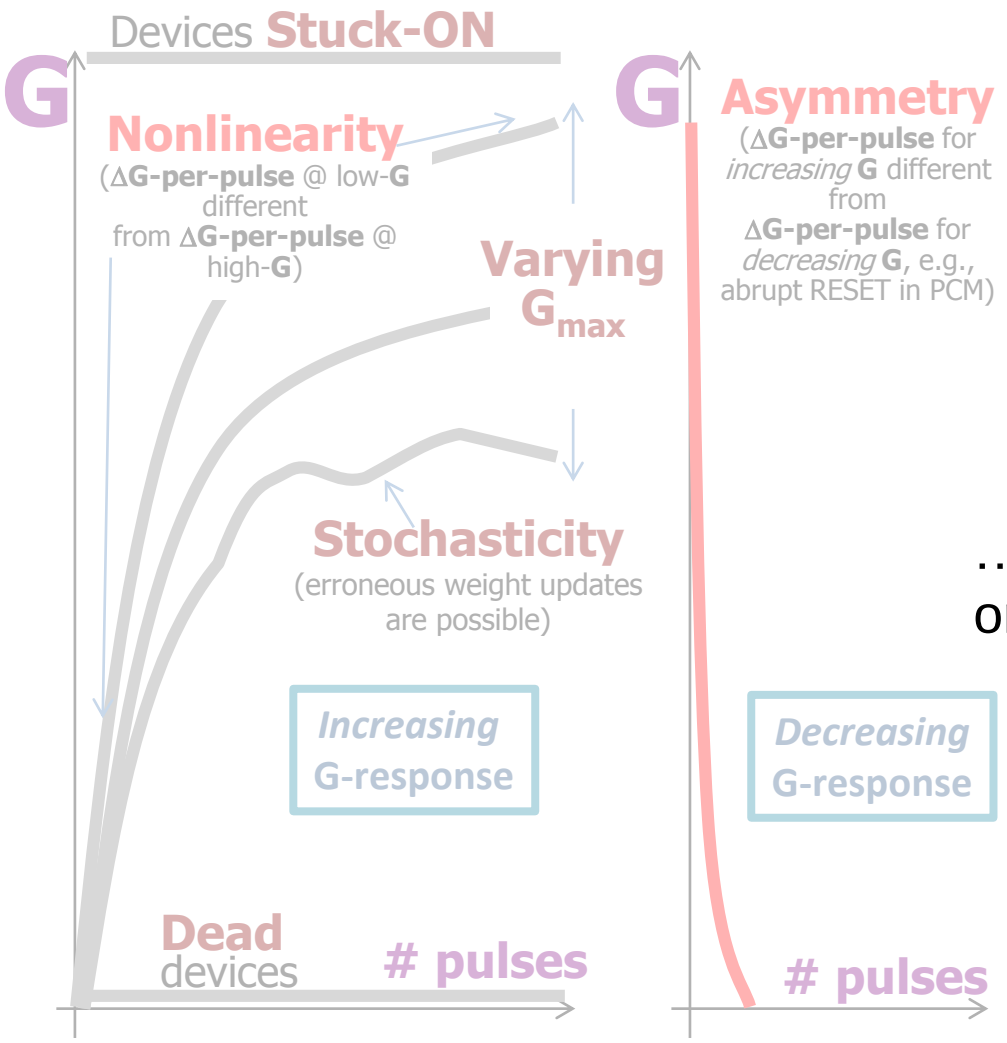
NVM imperfections



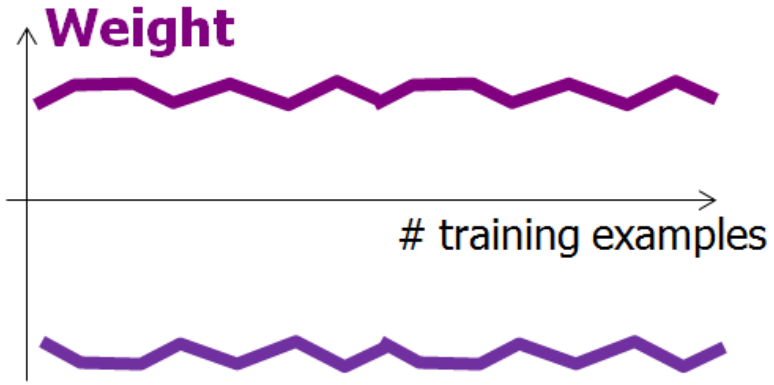
Abrupt RESET in Phase-Change Memory

→ significant asymmetry

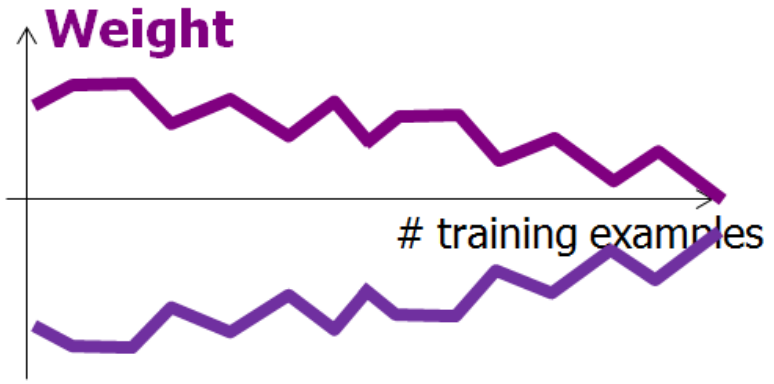
How do NVM imperfections cause trouble?



Any NN tends to "dither" many of its weights, with no long-term effect …

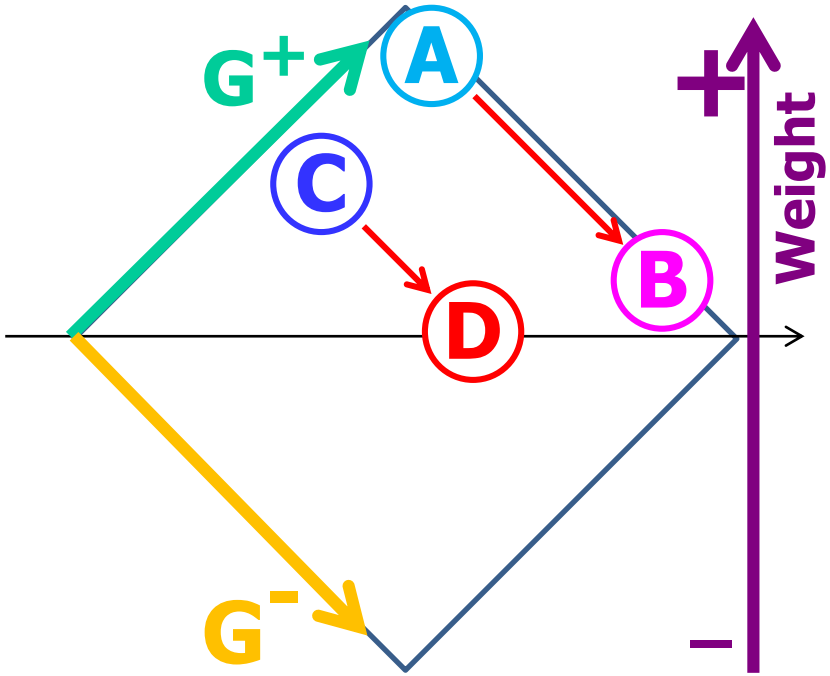


…**UNLESS** change is much "easier" in one direction (often → **low weights**)

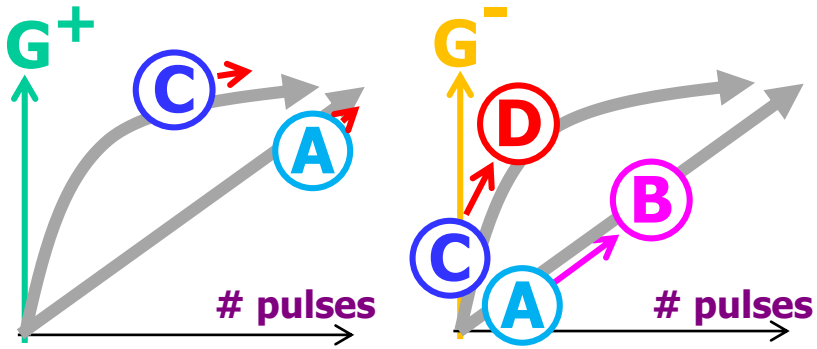


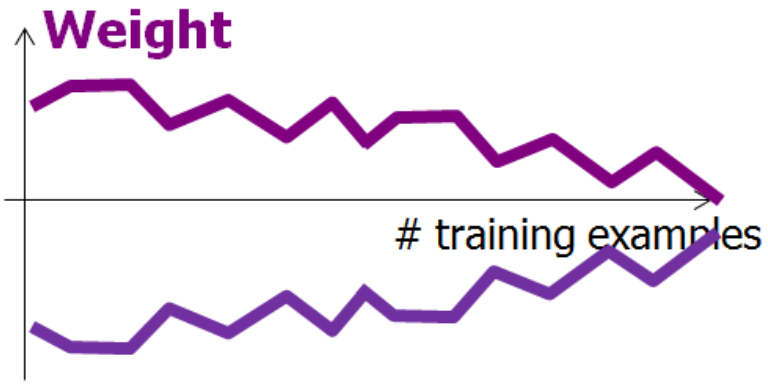
Low weights → small δ_j corrections → NO weights get updated ("freeze-out")

IEDM 2014: "G-diamond" concept



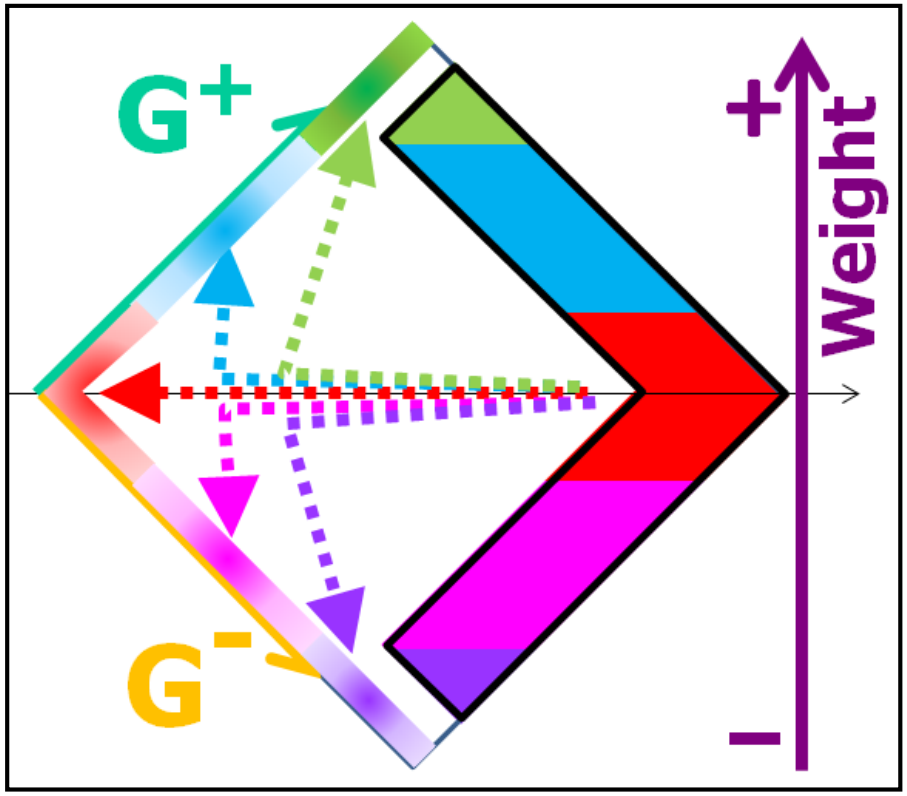
- Graphical method for understanding issues introduced by **nonlinearity** & **asymmetry**





[1] G. W. Burr, R. M. Shelby, et al., *IEDM Technical Digest*, 29.5, (2014).

IEDM 2014: Occasional RESET for PCM



- requires serial scan of conductance values

→ best to RESET only infrequently

- requires 2 Full RESETs followed by iterative SET

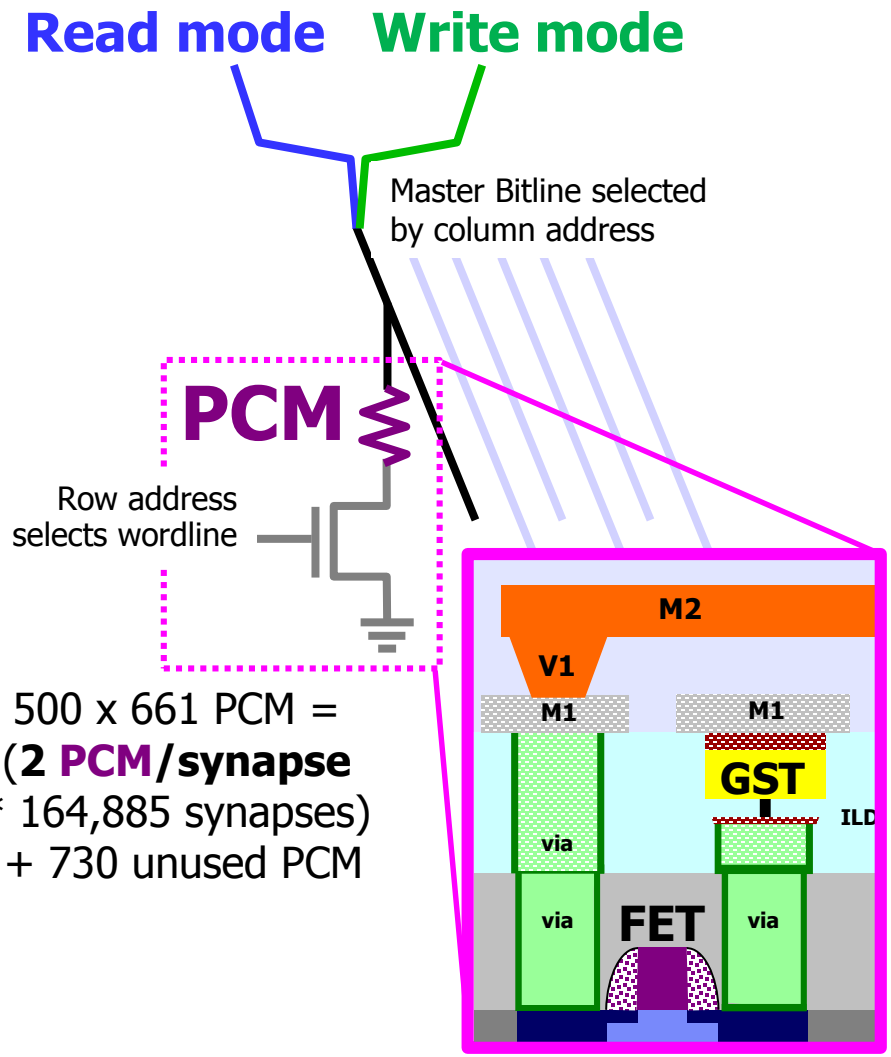
- synaptic weight after RESET NOT same as before

→ inherently inaccurate

[1] G. W. Burr, R. M. Shelby, et al., *IEDM Technical Digest*, 29.5, (2014).

Experimental NN implementation using PCM

"**Disruptor**" Array Diagnostic Monitor
→ 512 x 1024 array

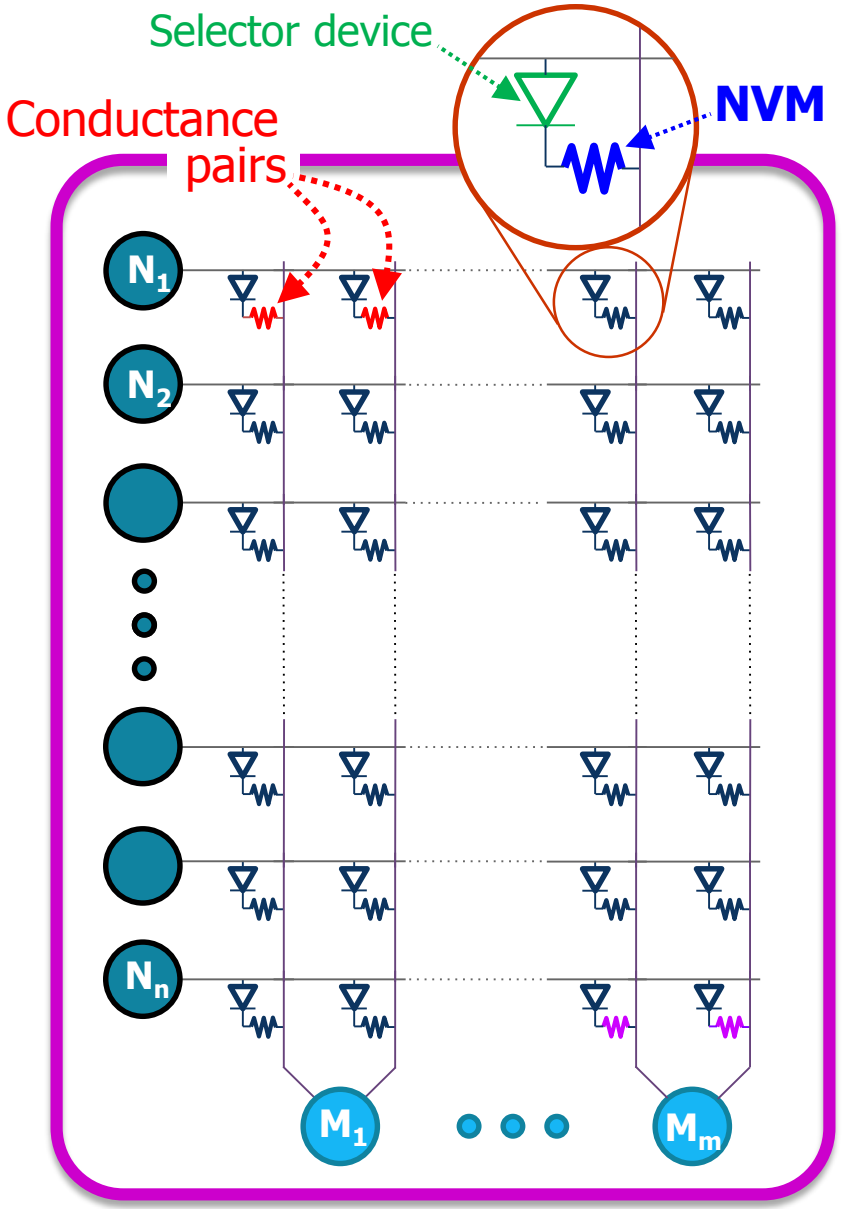


500 x 661 PCM =
(**2 PCM/synapse**
* 164,885 synapses)
+ 730 unused PCM

Hardware we **had** …

IBM/Macronix PCRAM project:
Chung Lam,
Matt BrightSky

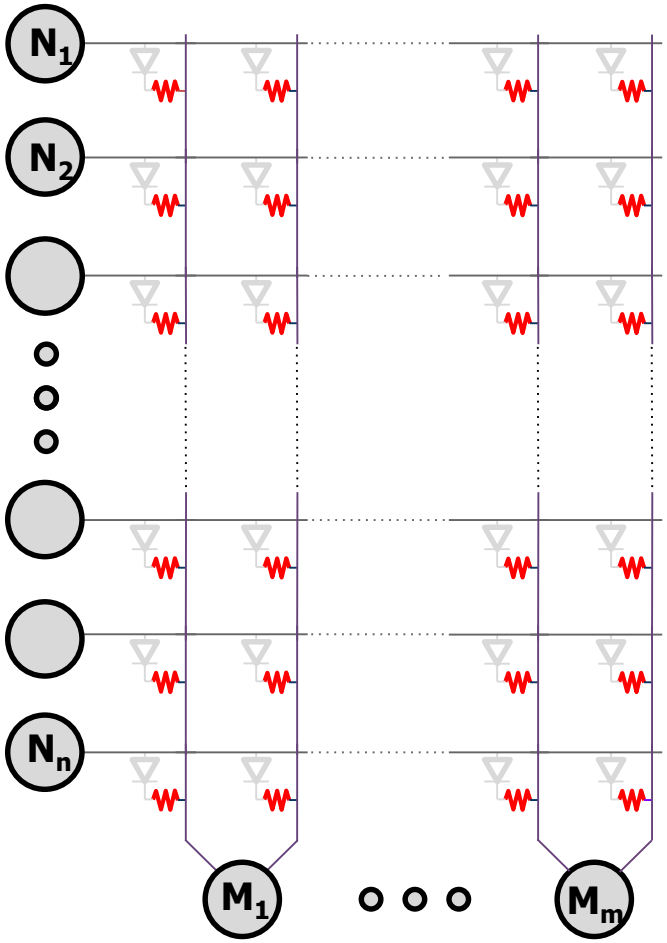
Experimental NN implementation using PCM

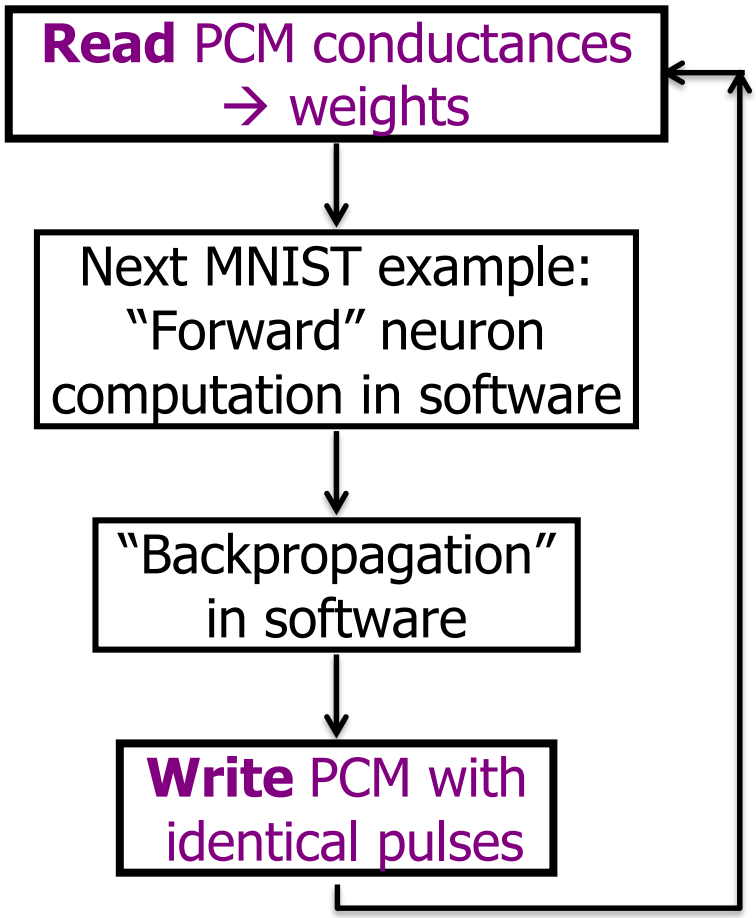


… was not the same as the hardware we **wanted** …

Experimental NN implementation using PCM

… but we wanted to do an experiment that told us **what performance we migh expect** with this target hardware…

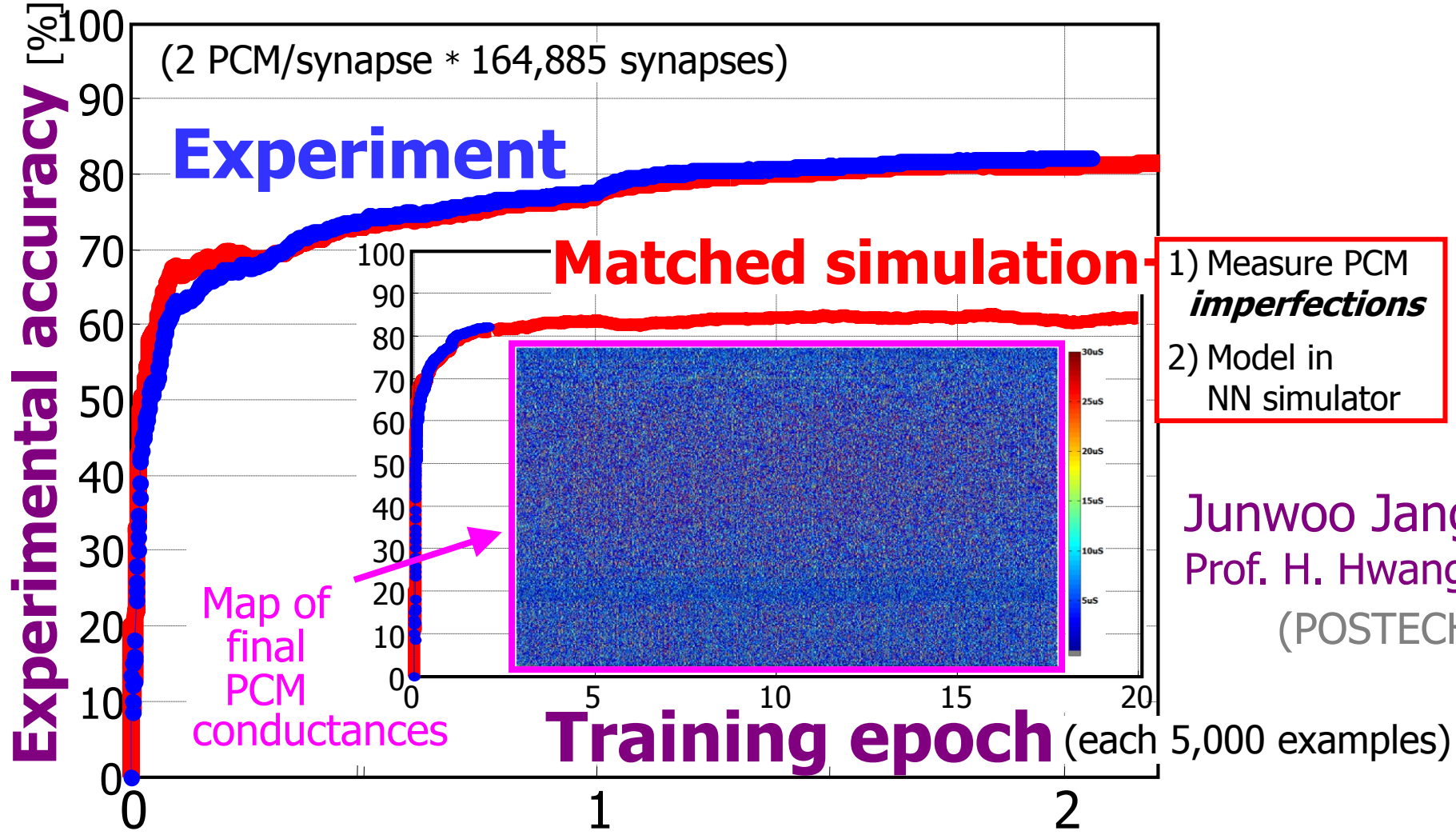




All weight operations **on PCM**

Accuracy results from IEDM 2014

- Reasonably good accuracy using PCM (**82.9% generalization** on "unseen" MNIST test set)



Junwoo Jang
Prof. H. Hwang
(POSTECH)

[1] G. W. Burr, R. M. Shelby, et al., *IEDM Technical Digest*, 29.5, (2014).

IEDM 2014: overall results

✓ Reasonably good accuracy on MNIST dataset using PCM
(with 5,000 examples → **82.9% generalization** on "unseen" test set)

✓ Crossbar-compatible **weight-update rule**

✓ "**G diamond**" graphical concept

✓ Doing inaccurate & infrequent "Occasional RESET" should work

- Extensive tolerancing enabled by **matching simulation** to experiment

- In general, we found NVM-based NN to be ...
- highly **resilient** to **random effects** (NVM variability, yield, and stochasticity)
- highly **sensitive** to **"gradient" effects** that act to steer all synaptic weights

- Low "learning-rate" → **high accuracy** & **low training energy**

[1] G. W. Burr, R. M. Shelby, et al., *IEDM Technical Digest*, 29.5, (2014).

IEDM 2014: overall results

✓ Reasonably good accuracy on MNIST dataset using PCM
(with 5,000 examples → **82.9% generalization** on "unseen" test set)

✓

Problem!!!

MNIST dates from ~1998
→ it's now considered a "minor-league" ML problem!

We MUST get to competitive performance numbers:

~94% w/ 5,000 examples

97-98% w/ 60,000 examples

- Low "learning-rate" → **high accuracy** & **low training energy**

[1] G. W. Burr, R. M. Shelby, et al., *IEDM Technical Digest*, 29.5, (2014).

Published work on "what do we need from the NVM?"

[1] **IEDM 2014**

Experimental demonstration and tolerancing of a large-scale neural network (165,000 synapses), using phase-change memory as the synaptic weight element

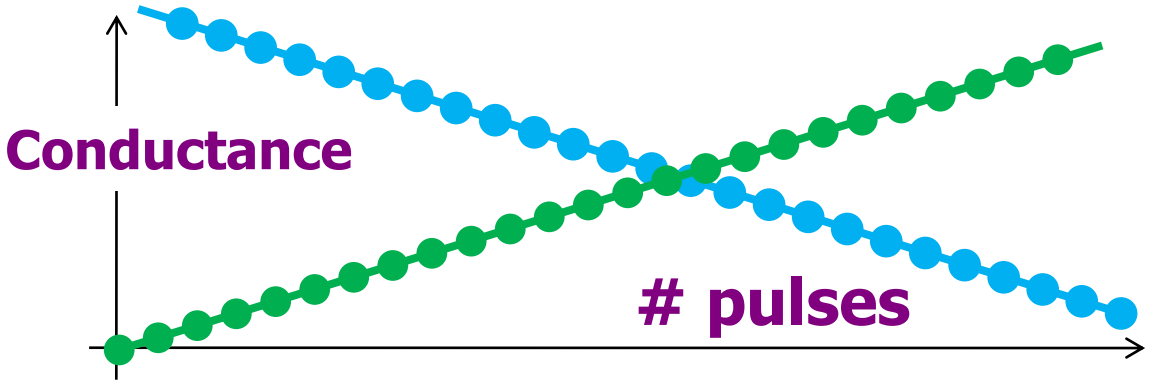
G. W. Burr, R. M. Shelby, C. di Nolfo, J. W. Jang‡, R. S. Shenoy, P. Narayanan, K. Virwani, E. U. Giacometti, B. Kurdi, and H. Hwang‡

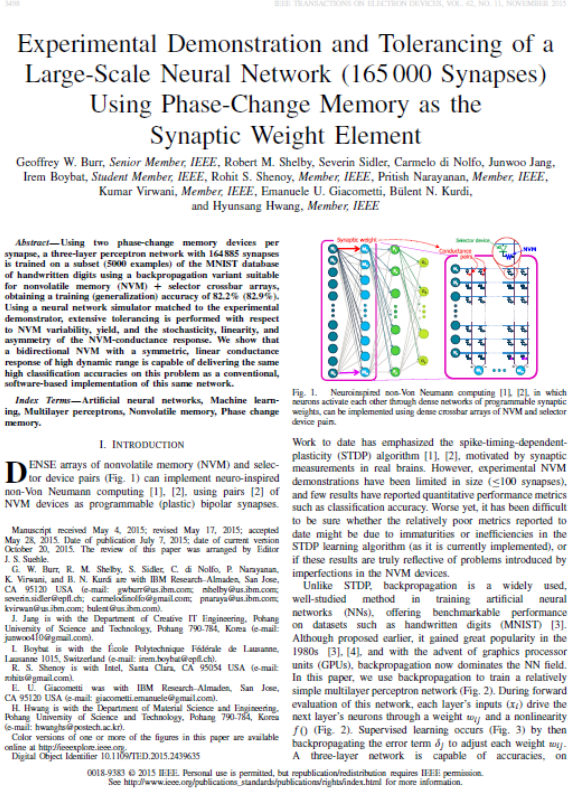
→ First large-scale mixed hardware-software demonstration + tolerancing

→ ~82% accuracy on MNIST with 5000 examples

[2] Invited paper in **IEEE-TED** (v**62**(11), 3498 (2015).)

→ Showed that high accuracy
(~94% w/ 5,000 examples,
97-98% w/ 60,000 examples)
is possible – NVM just needs a **linear** conductance response w/ **small** steps





Experimental Demonstration and Tolerancing of a Large-Scale Neural Network (165 000 Synapses) Using Phase-Change Memory as the Synaptic Weight Element

Geoffrey W. Burr, *Senior Member, IEEE*, Robert M. Shelby, Severin Sidler, Carmelo di Nolfo, Junwoo Jang, Irem Boybat, *Student Member, IEEE*, Rohit S. Shenoy, *Member, IEEE*, Pritish Narayanan, *Member, IEEE*, Kumar Virwani, *Member, IEEE*, Emanuele U. Giacometti, Bülent N. Kurdi, and Hyunsang Hwang, *Member, IEEE*

***Abstract*—Using two phase-change memory devices per synapse, a three-layer perceptron network with 164 885 synapses is trained on a subset (5000 examples) of the MNIST database of handwritten digits using a backpropagation variant suitable for nonvolatile memory (NVM) + selector crossbar arrays, obtaining a training (generalization) accuracy of 82.2% (82.9%). Using a neural network simulator matched to the experimental demonstrator, extensive tolerancing is performed with respect to NVM variability, yield, and the stochasticity, linearity, and asymmetry of the NVM-conductance response. We show that a bidirectional NVM with a symmetric, linear conductance response of high dynamic range is capable of delivering the same high classification accuracies on this problem as a conventional, software-based implementation of this same network.**

***Index Terms*—Artificial neural networks, Machine learning, Multilayer perceptrons, Nonvolatile memory, Phase change memory.**

I. Introduction

DENSE arrays of nonvolatile memory (NVM) and selector device pairs (Fig. 1) can implement neuro-inspired non-Von Neumann computing [1], [2], using pairs [2] of NVM devices as programmable (plastic) bipolar synapses.

Manuscript received May 4, 2015; revised May 17, 2015; accepted May 28, 2015. Date of publication July 7, 2015; date of current version October 20, 2015. The review of this paper was arranged by Editor J. S. Suehle.

G. W. Burr, R. M. Shelby, S. Sidler, C. di Nolfo, P. Narayanan, K. Virwani, and B. N. Kurdi are with IBM Research–Almaden, San Jose, CA 95120 USA (e-mail: gwburr@us.ibm.com; rmshelby@us.ibm.com; severin.sidler@epfl.ch; carmelodinolfo@gmail.com; pnaraya@us.ibm.com; kvirwan@us.ibm.com; bulent@us.ibm.com).

J. Jang is with the Department of Creative IT Engineering, Pohang University of Science and Technology, Pohang 790-784, Korea (e-mail: junwoo410@gmail.com).

I. Boybat is with the École Polytechnique Fédérale de Lausanne, Lausanne 1015, Switzerland (e-mail: irem.boybat@epfl.ch).

R. S. Shenoy is with Intel, Santa Clara, CA 95054 USA (e-mail: rohits@gmail.com).

E. U. Giacometti was with IBM Research–Almaden, San Jose, CA 95120 USA (e-mail: giacometti.emanuele@gmail.com).

H. Hwang is with the Department of Material Science and Engineering, Pohang University of Science and Technology, Pohang 790-784, Korea (e-mail: hwanghs@postech.ac.kr).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TED.2015.2439635

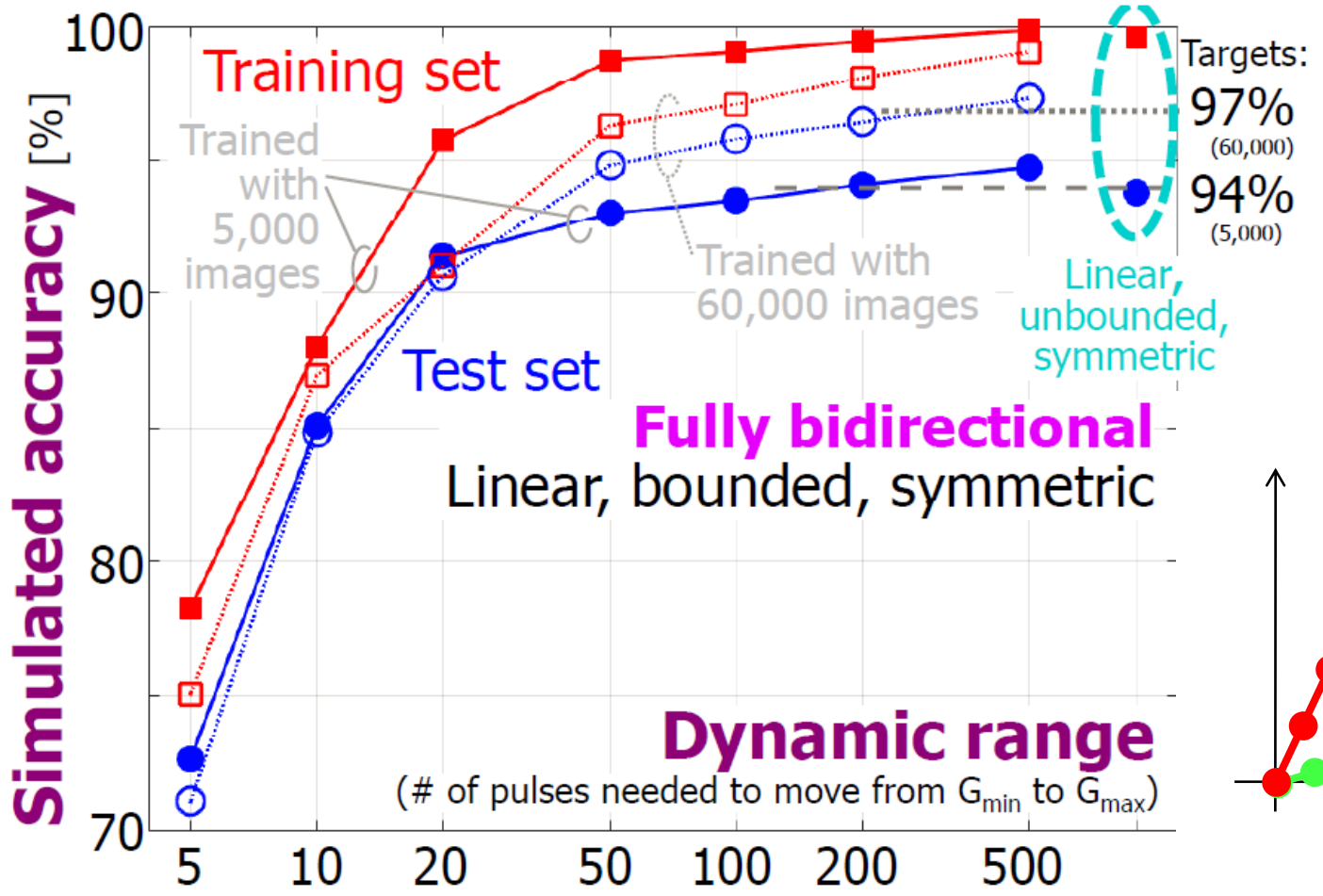
Fig. 1. Neuroinspired non-Von Neumann computing [1], [2], in which neurons activate each other through dense networks of programmable synaptic weights, can be implemented using dense crossbar arrays of NVM and selector device pairs.

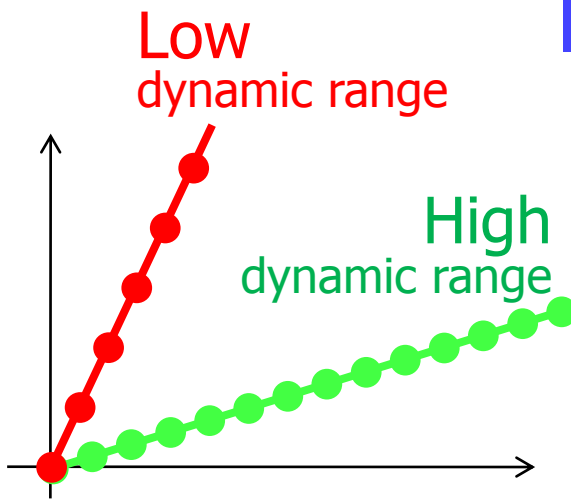
Work to date has emphasized the spike-timing-dependent-plasticity (STDP) algorithm [1], [2], motivated by synaptic measurements in real brains. However, experimental NVM demonstrations have been limited in size (≤100 synapses), and few results have reported quantitative performance metrics such as classification accuracy. Worse yet, it has been difficult to be sure whether the relatively poor metrics reported to date might be due to immaturities or inefficiencies in the STDP learning algorithm (as it is currently implemented), or if these results are truly reflective of problems introduced by imperfections in the NVM devices.

Unlike STDP, backpropagation is a widely used, well-studied method in training artificial neural networks (NNs), offering benchmarkable performance on datasets such as handwritten digits (MNIST) [3]. Although proposed earlier, it gained great popularity in the 1980s [3], [4], and with the advent of graphics processor units (GPUs), backpropagation now dominates the NN field. In this paper, we use backpropagation to train a relatively simple multilayer perceptron network (Fig. 2). During forward evaluation of this network, each layer's inputs (x_i) drive the next layer's neurons through a weight w_{ij} and a nonlinearity $f()$ (Fig. 2). Supervised learning occurs (Fig. 3) by then backpropagating the error term δ_j to adjust each weight w_{ij}. A three-layer network is capable of accuracies, on

IEEE-TED 2015: what we need from NVM

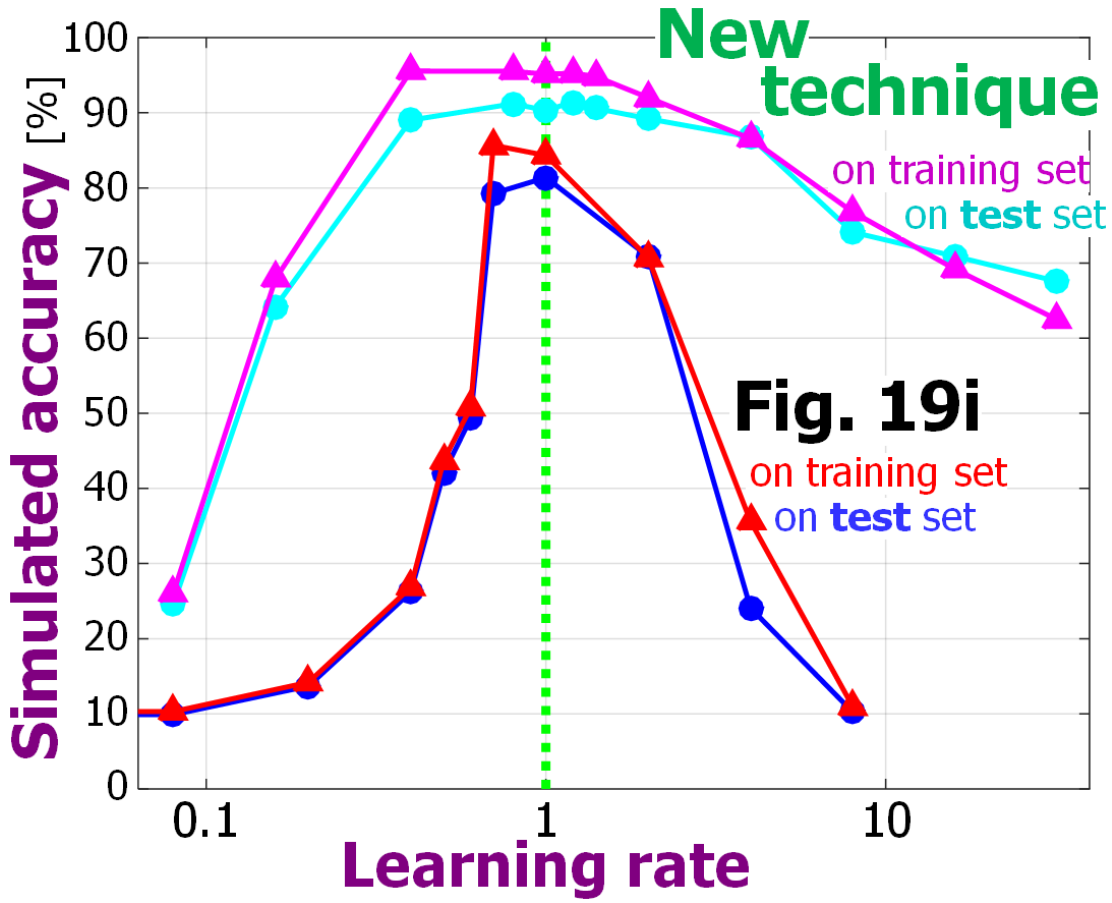
We showed that an "ideal" bi-directional NVM with a linear G-response of high dynamic range *can* provide **the full performance** available from the algorithm





[2] G. W. Burr, R. M. Shelby, et al., *IEEE Trans. Electr. Dev.,* 62(11), 3498-3507 (2015).

"Local Gains" technique for non-ideal-NVM devices



Carmelo di Nolfo
Irem Boybat

This technique…

- reduces need to tune "learning rate" precisely
- improves performance (suppresses synapses that "dither")
- reduces power consumption

Published work on "what do we need from the NVM?"

[1] **IEDM 2014**

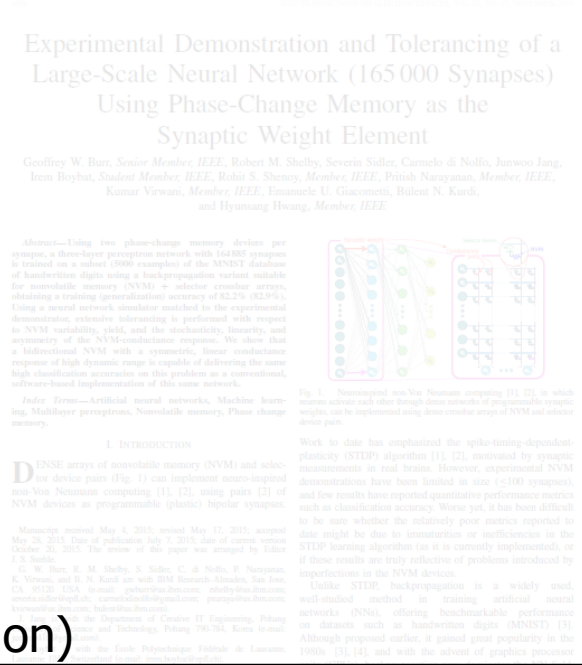
Experimental demonstration and tolerancing of a large-scale neural network (165,000 synapses), using phase-change memory as the synaptic weight element

G. W. Burr, R. M. Shelby, C. di Nolfo, J. W. Jang‡, R. S. Shenoy, P. Narayanan, K. Virwani, E. U. Giacometti, B. Kurdi, and H. Hwang‡

→ First large-scale mixed hardware-software demonstration + tolerancing

→ ~82% accuracy on MNIST with 5000 examples

[2] Invited paper in **IEEE-TED** (v**62**(11), 3498 (2015).)


Experimental Demonstration and Tolerancing of a Large-Scale Neural Network (165 000 Synapses) Using Phase-Change Memory as the Synaptic Weight Element

→ Showed that high accuracy (~94% w/ 5,000 examples, 97-98% w/ 60,000 examples) **is** possible – NVM just needs a **linear** conductance response w/ **small** steps

Conductance

[3] Invited talk **@IEDM 2015** (Neuromorphic Focus Session)

Large-scale neural networks implemented with non-volatile memory as the synaptic weight element: comparative performance analysis (accuracy, speed, and power)

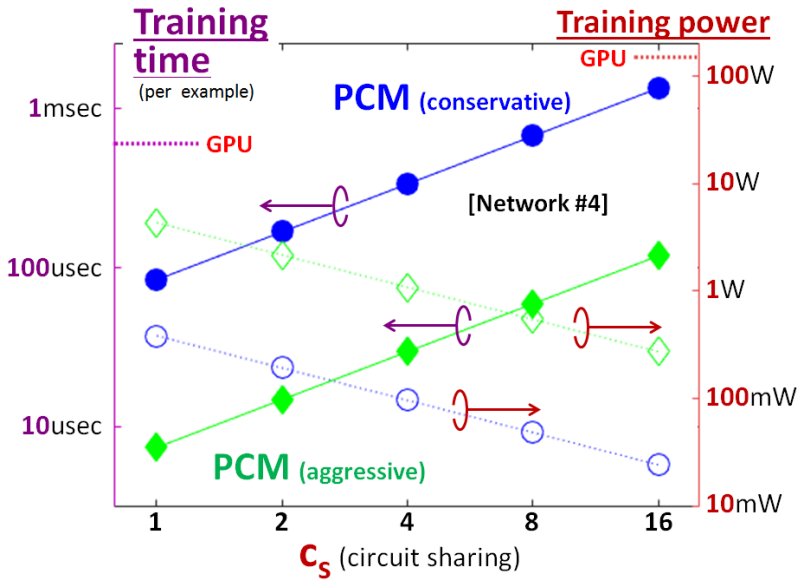
G. W. Burr, P. Narayanan, R. M. Shelby, S. Sidler, I. Boybat, C. di Nolfo, and Y. Leblebici†

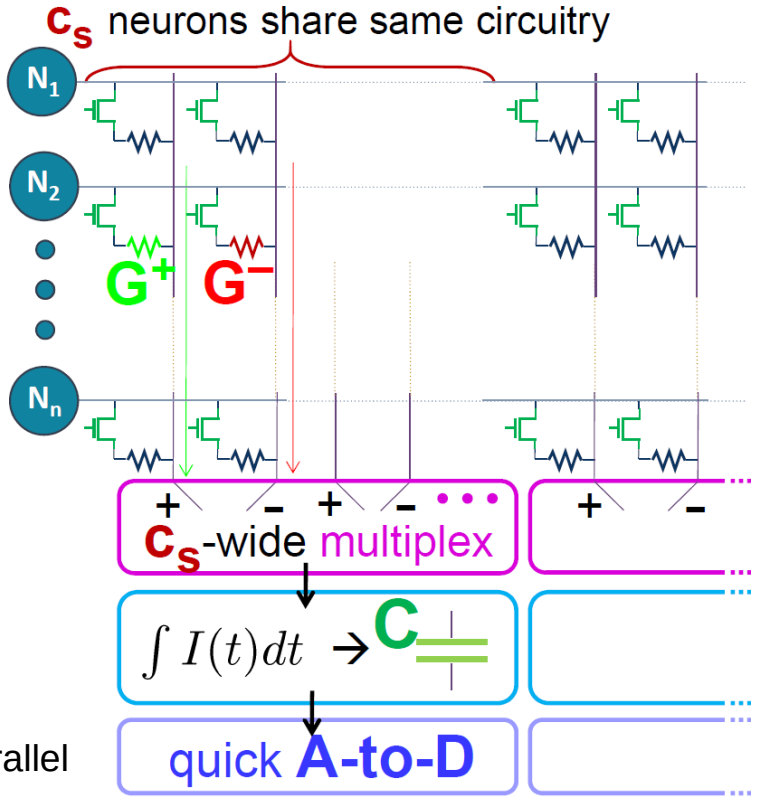
→ showed prospects for speedup (up to 25x) and lower power (100x to 3000x)

Motivation: Need to Minimize Circuit Sharing (IEDM15)

Shown prospects for **2-3 orders-of-magnitude speedup**
and 120-2850x lower power

These **speed benefits** require **minimal** circuit sharing, c_s





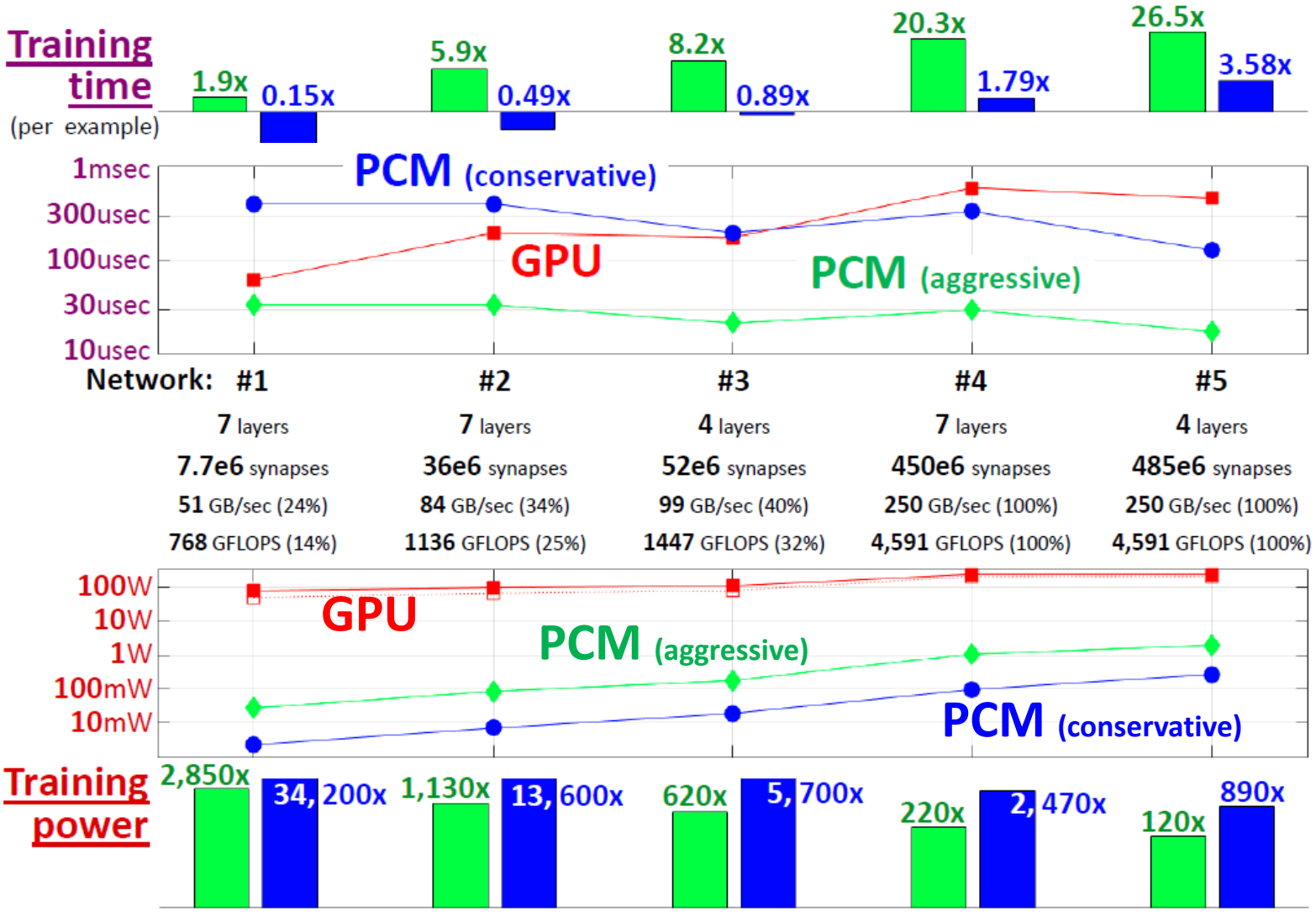
Design of **highly area-efficient circuits** is essential
→ read and write of many synaptic bitlines (& wordlines) in parallel

Tradeoff Circuit Complexity against Device and Algorithm Requirements

ISCAS 2017: Reducing Circuit Design Complexity for Neuromorphic Machine Learning Systems based on Non-Volatile Memory Arrays

NVM benefits in speed & power vs. GPU

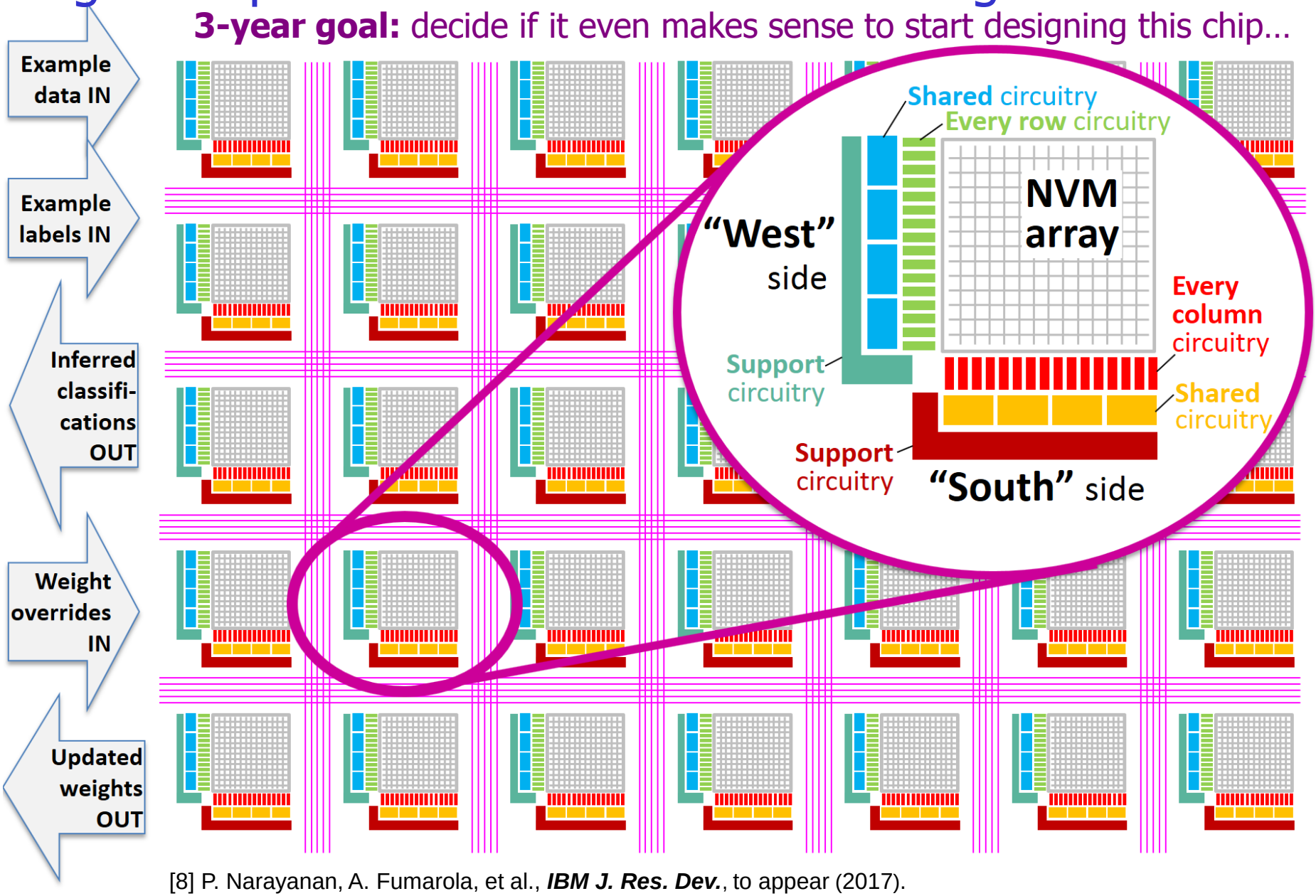
These initial, "back-of-the-envelope" calculations suggest up to **25x** speedup and **120-2850x** lower power Machine Learning than GPUs



[3] G. W. Burr, P. Narayanan, et al. (invited), *IEDM 2015,* T4.4 (2015).

Target: chip for NVM-for-Machine-Learning acceleration

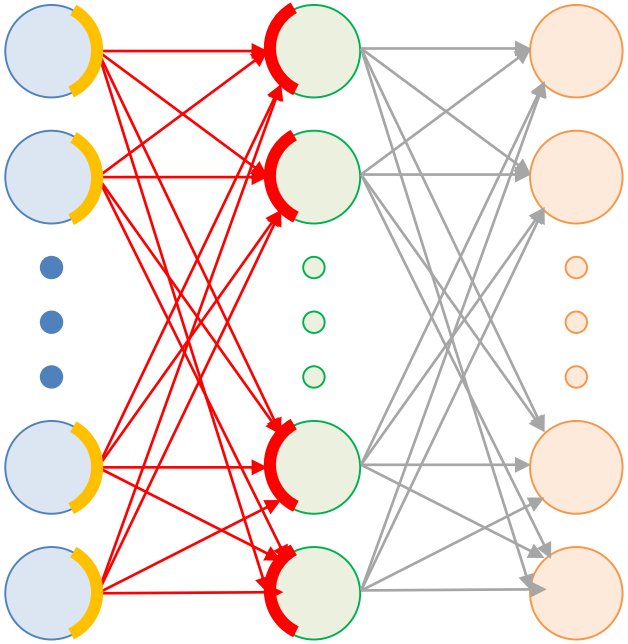
3-year goal: decide if it even makes sense to start designing this chip…



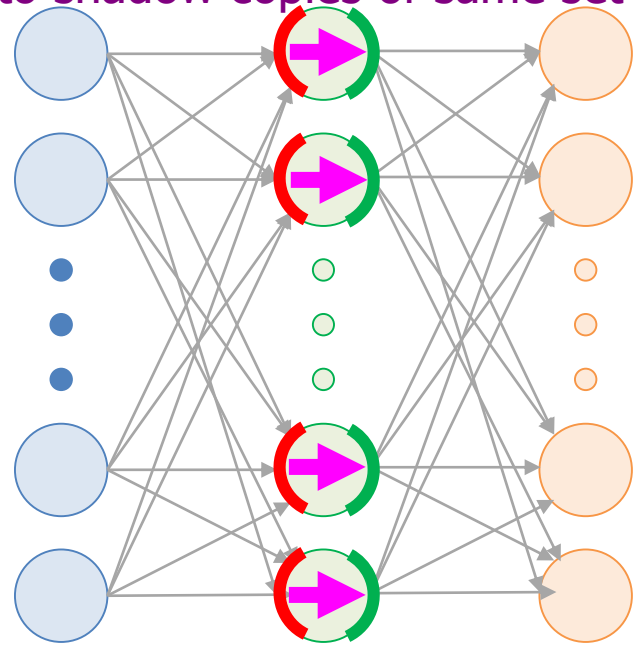
[8] P. Narayanan, A. Fumarola, et al., ***IBM J. Res. Dev.***, to appear (2017).

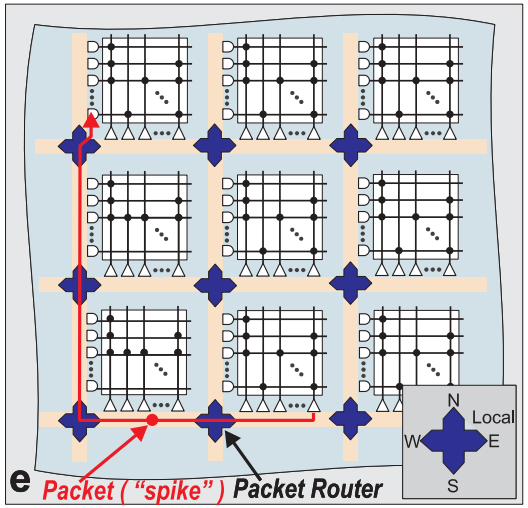
Our routing is NOT point-to-point

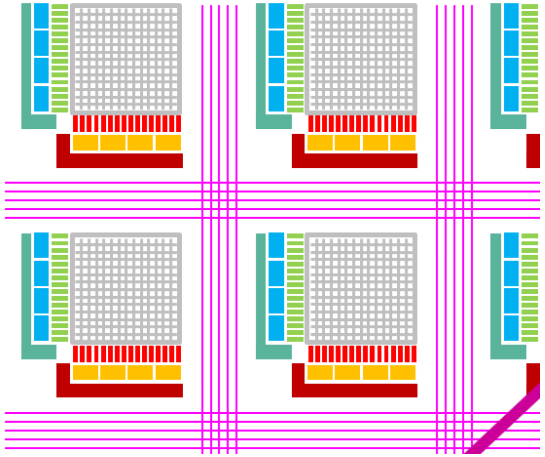
"Route spikes to the right axons"



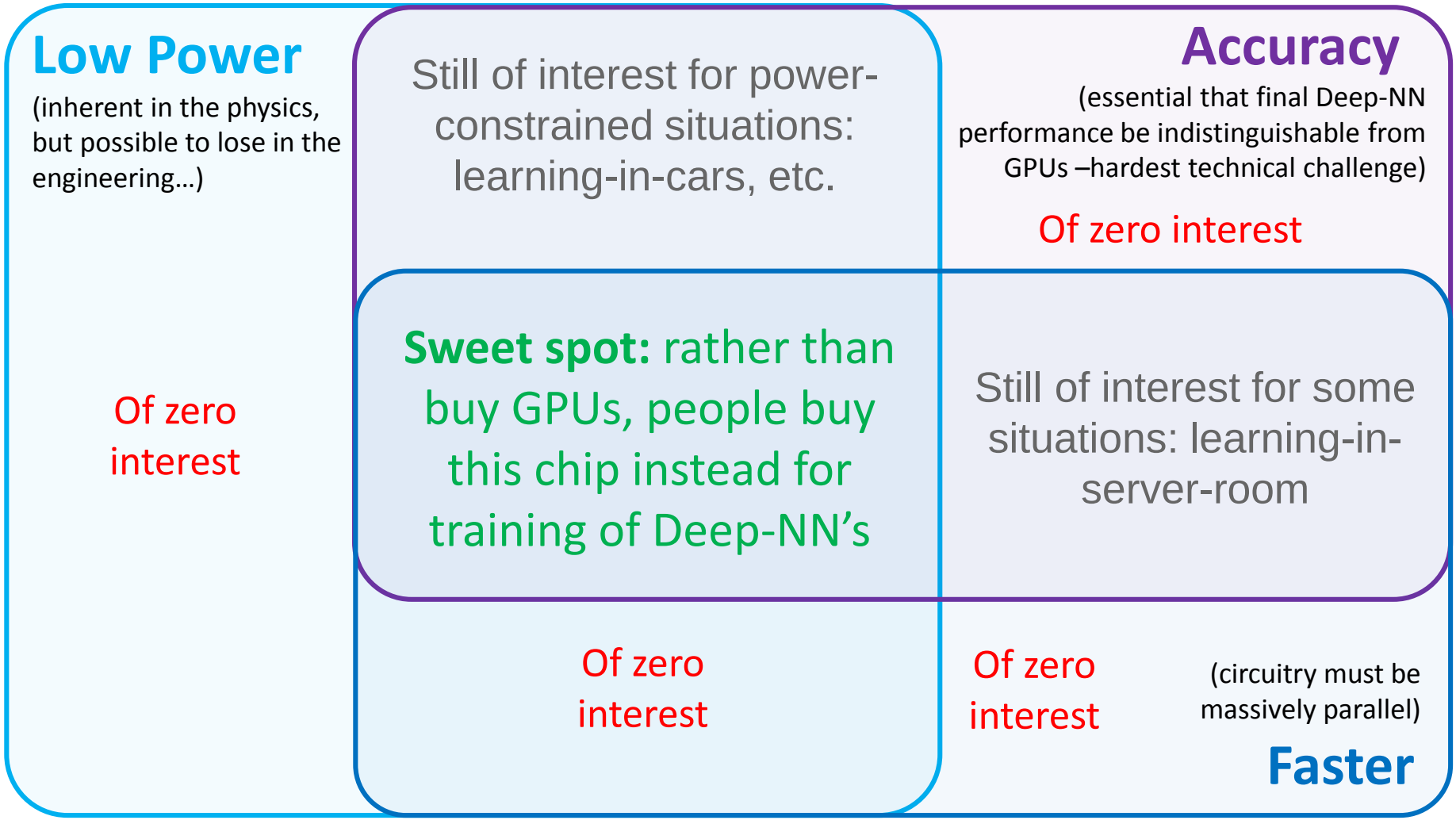
"Copy to shadow copies of same set of somas"







The business case for such a system (vs. a GPU)

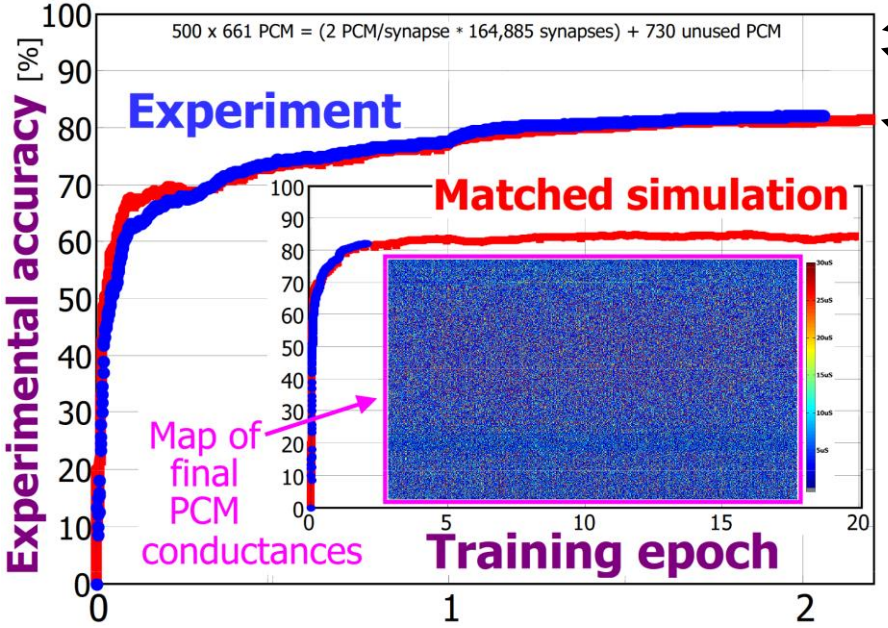


"Out of plane" axis → wide applicability
(networks of varying shape with varying types of layers)

Deep-ML performance with existing NVM devices

Where we were in **June 2014**

- 82% accuracy w/ 5000 examples
- Too slow to try all 60,000 examples in the "MNIST" dataset



[1] G. W. Burr, R. M. Shelby, et al., *IEDM Technical Digest*, 29.5, (2014).

Where we are in **January 2017**

- **SAME wafer** of PCM devices
- **96-97% TEST accuracy** w/ 60,000 examples

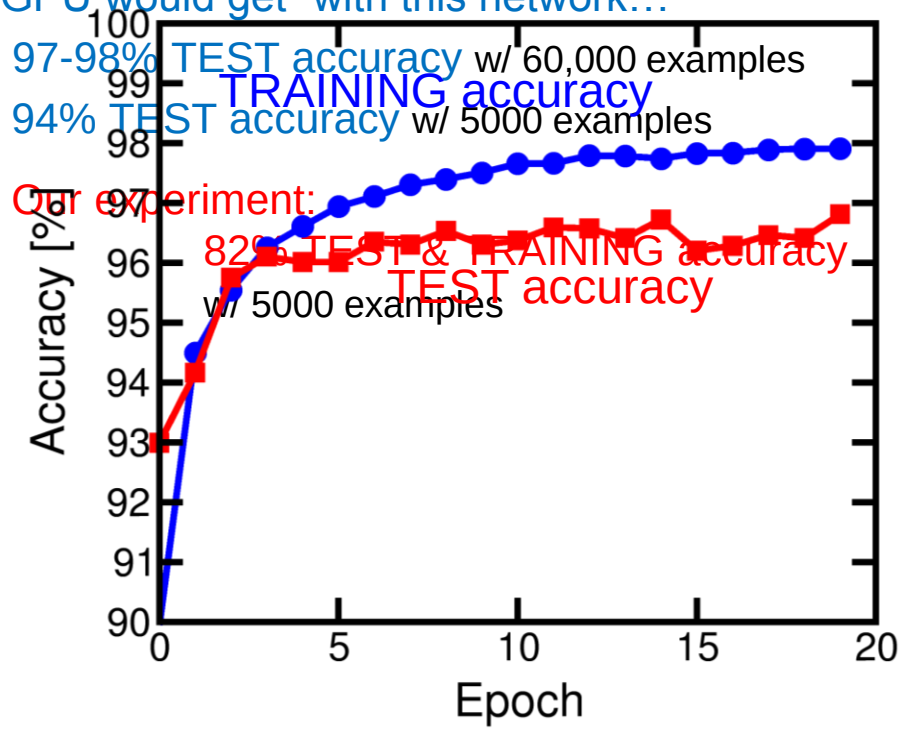
"What a GPU would get" with this network…

97-98% TEST accuracy w/ 60,000 examples

94% TEST accuracy w/ 5000 examples

Our experiment:

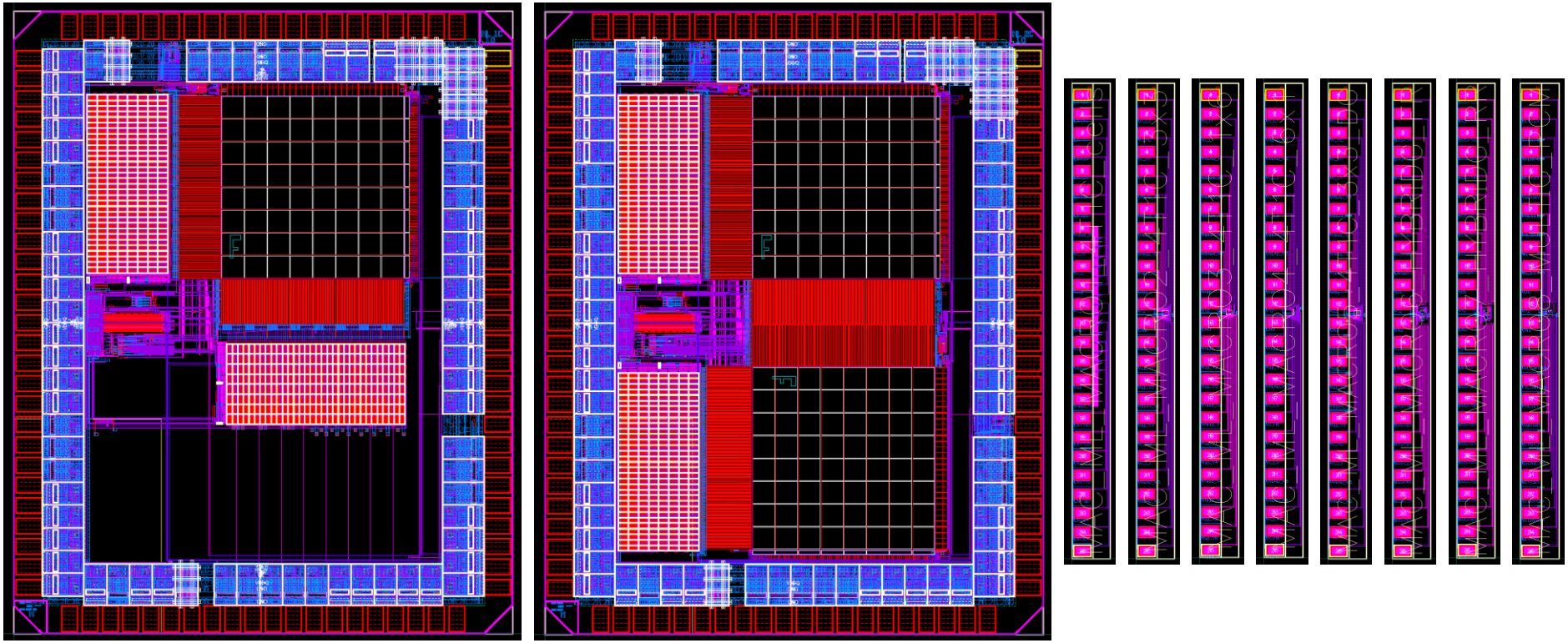
82% TEST & TRAINING accuracy w/ 5000 examples



What changed?

- Multiple inventions, including a **new unit-cell concept** (PCM^{++})

March 2017 tapeout: 2 ADMs and 8 macros



What we estimated in **June 2015**

Estimated a …

- **25x** speedup over GPU
- 100-3000x power advantage

Where we are in **January 2017**

Estimating a …

- **500x** speedup over GPU
- power analysis in progress…

NVM-for-Machine Learning: Recent/upcoming papers

1. S. Sidler, I. Boybat, et al., "Large-scale neural networks implemented with non-volatile memory as the synaptic weight element: impact of conductance response," ***ESSDERC 2016***, Sept. **2016**.
   - Impact of idealized jump-tables –
     e.g., "how much of the conductance response must be linear?"

2. A. Fumarola, P. Narayanan, et al., "Accelerating Machine Learning with Non-Volatile Memory: exploring device and circuit tradeoffs," ***2016 Internat. Conf. on Rebooting Computing***, Oct. **2016**.
   - Impact of real PCMO jump-tables, time-dependent conductances, some circuit choices

3. G. W. Burr, R. M. Shelby et al., "Neuromorphic computing using non-volatile memory," ***Advances in Physics X***, 2(1), 89-124 (**2017**).
   - Review of the NVM-for-neuromorphic field as a whole…

4. P. Narayanan, L. Sanches, et al., "Reducing Circuit Design Complexity for Neuromorphic Machine Learning Systems Based on Non-Volatile Memory Arrays," ***ISCAS* 2017**.
   - Impact of circuit choices (nonlinearity, derivative, implementation of "Occasional RESET")

5. P. Narayanan, A. Fumarola, et al., "Towards on-chip acceleration of the backpropagation algorithm using non-volatile memory," ***IBM Journal of Research and Development***, to appear (**2017**)
   - Summarizes the circuit design challenges

6. I. Boybat, C. di Nolfo, et al., "Improved Deep Neural Network hardware-accelerators based on Non-Volatile-Memory: the Local Gains technique," submitted to ***Intl. Conf. Rebooting Computing*** (**2017**)
   - Explains our local-gains algorithm

Summary: on-chip learning with non-volatile memory

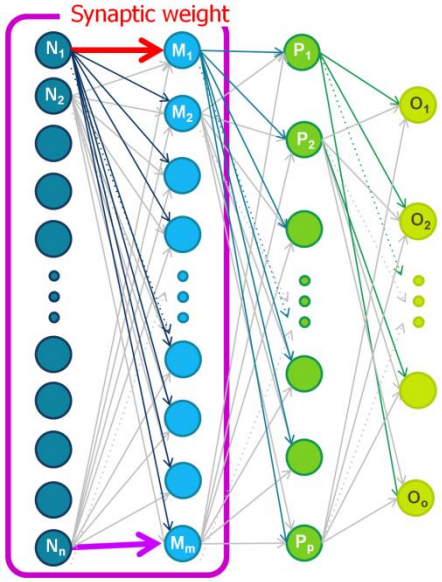
- **NVM-based crossbar arrays CAN accelerate training of Deep Machine Learning** compared to GPU-based training
  - → Multiply-accumulate performed AT the data
  - → We see possibilities for **500x speedup** & **orders-of-magnitude lower power**

- **Need: competitive ML accuracy**
  - ✓ **experimental results**: 96-97% on "minor-league" MNIST using PCM
    - ✓ Nearly ready to move from "minor-league" to "major-league" DNN problems
  - ✓ "ideal" NVM w/ linear G-response of high dynamic range → sufficient!
    - → ARC (RFI): use ***existing*** NVM (PCM, etc.); invent device/circuit/network techniques
    - → YKT: Cognitive Materials/RPU project: develop ***new*** forms of NVM

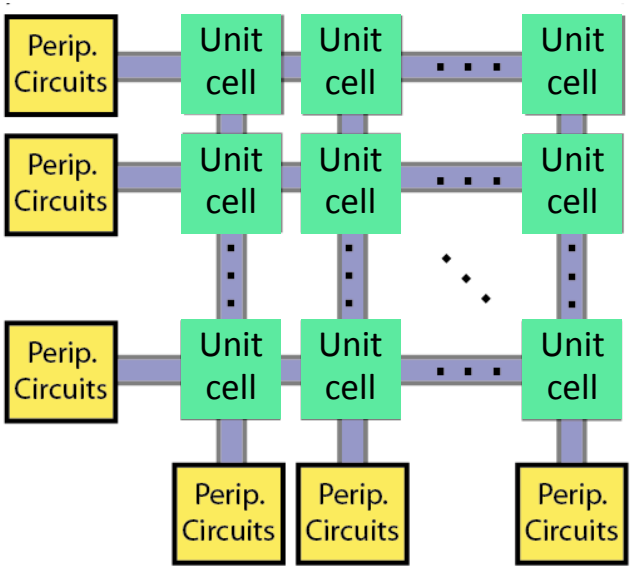
- **Need: area-efficient peripheral circuitry**
  - ✓ power benefits are quite significant
  - ✓ but **design must preserve** speedup benefits
    - → **Aggressive** timing & **minimal circuit sharing** **(low c_s)**

IBM Research – multiple paths to faster ML training

Accelerate backpropagation training (e.g., Deep-NN, Conv-NN, and LSTM)…



…by performing **multiply-accumulates** *on-chip* using ***analog*** resistive memory elements.



Unit cell =

Existing NVM
(e.g., PCM, "PCMO")

- Available now
- Truly non-volatile
- Compact cell
- Nonlinear + asymmetric

Capacitors
(CMOS-RPU)

- Available now
- Leaky → need refresh?
- Larger cell
- Suitably linear

Improved NVM
(Device-RPU)

- Yet to be developed
- Non-volatile
- Compact cell
- Linearity is key

(asymmetry can be dealt with)

How memory device requirements vary between applications

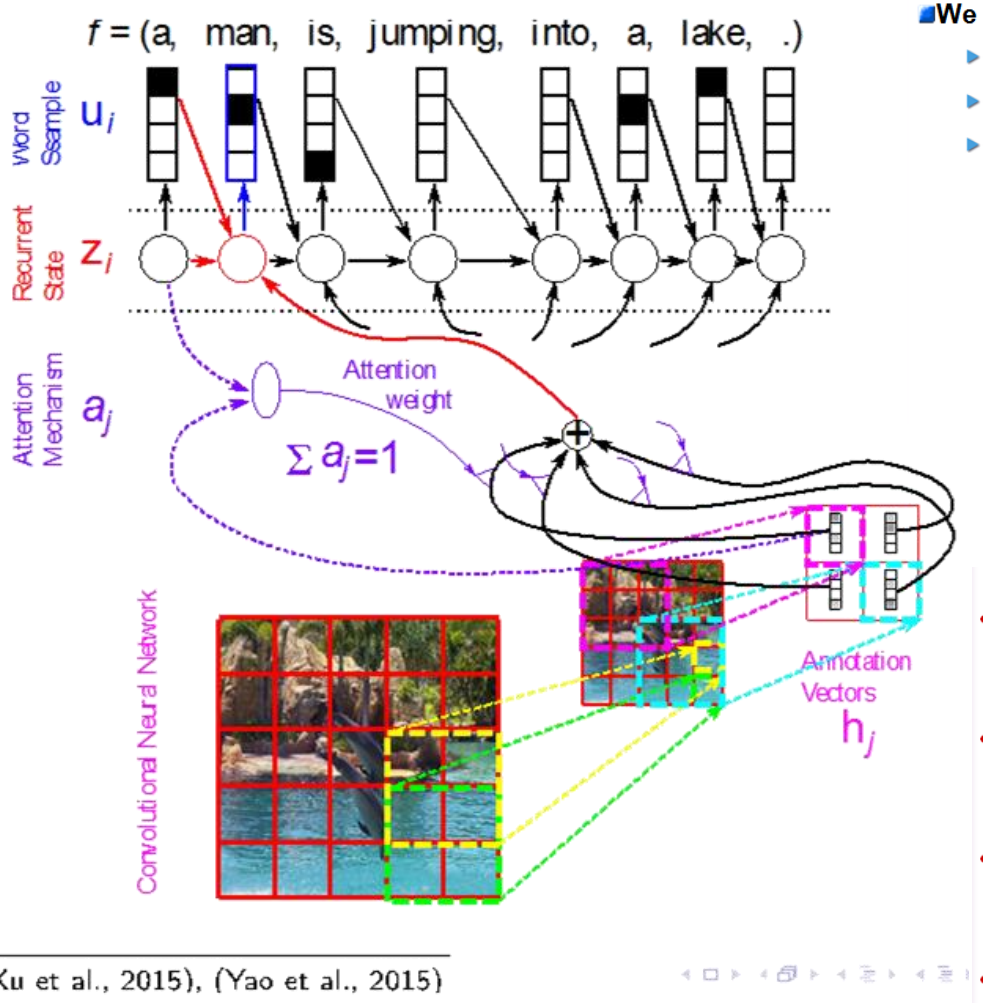
	for **Storage Class Memory**	for **Neuromorphic**
Resistance states:	Need **2-8 distinct** states	need **continuous range** of resistance states
"device history" is…	a distraction.	absolutely **essential.**
LRS cannot be …	too high → need fast read	too low → read aggregates 100's of devices
Failing-as-SHORT	is just as bad as Failing-as-OPEN	much **worse** than Failing-as-OPEN
Any 2-terminal access device had better be…	nearly perfect + **cannot fail** as a SHORT	nearly perfect + **cannot fail** as a SHORT

What are the CS folks doing/ignoring?

- What are the computer scientists up to?
  - Design-based Big Data Analytics – not "neural"
  - Machine Learning
    - Deep Neural Networks
      - cross-entropy loss
        (→ "permission" to ignore f' in the output layer)
      - dropout
        (better regularization → better generalization)
      - ADAGRAD
        (adaptively decrease learning rate → less hyperparameter tuning)
      - ReLU, batch normalization
        (suppress "internal covariant shift" during learning)
      - convNets with NO fully-connected layers
        (response to limited GPU memory & memory-bandwidth)
    - Transcending DNN – LSTM, Recurrent NN, Reinforcement Learning

For CS researchers, what is next?

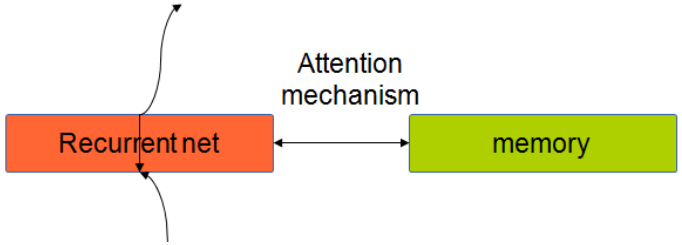
Image-to-Text: Caption Generation with Attention



(Xu et al., 2015), (Yao et al., 2015)

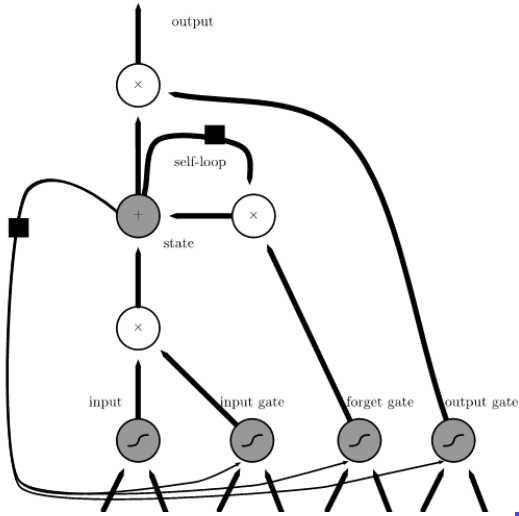
Attention on "Memory Elements"

- **Recurrent networks cannot remember things for very long**
  - The cortex only remember things for 20 seconds
- **We need a "hippocampus" (a separate memory module)**
  - LSTM [Hochreiter 1997], registers
  - **Memory networks** [Weston et 2014] (FAIR), associative memory
  - NTM [Graves et al. 2014], "tape".



Gated Recurrent Units & Long Short-Term Memory

- **Create a path where gradients can flow for longer with self-loop**
- Corresponds to an eigenvalue of Jacobian slightly less than 1
- LSTM is **heavily used** *(Hochreiter & Schmidhuber 1997)*
- GRU light-weight version *(Cho et al 2014)*

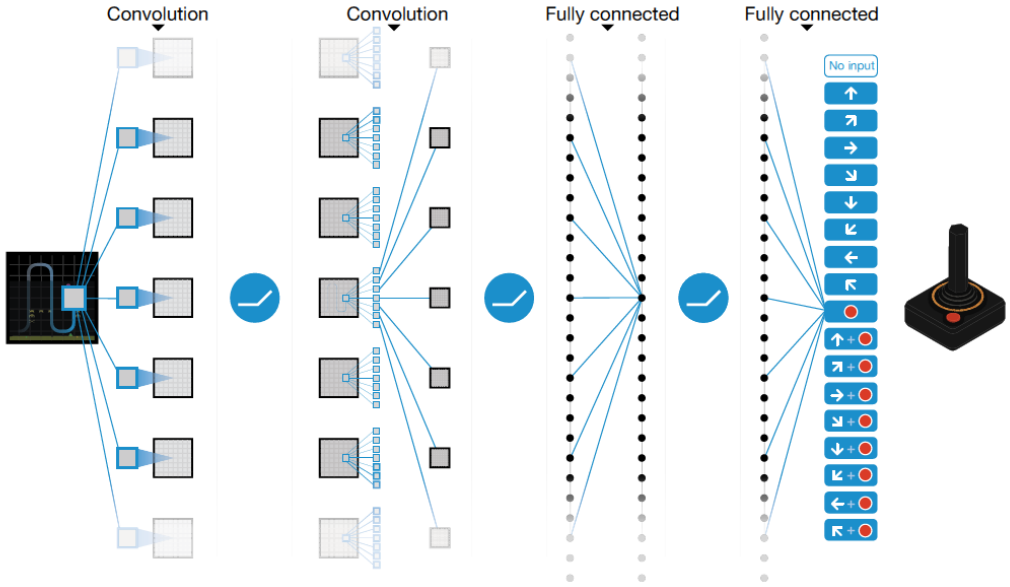


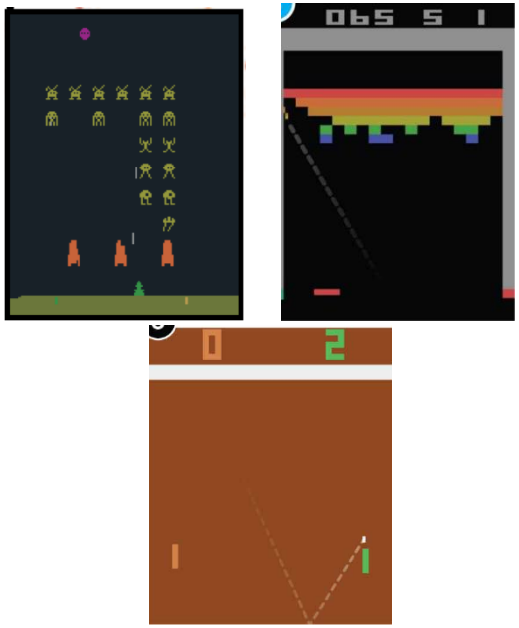
Slides from Y. Bengio, NIPS 2015 talk.

For CS researchers, what is next?

Deep Neural Networks + Reinforcement Learning

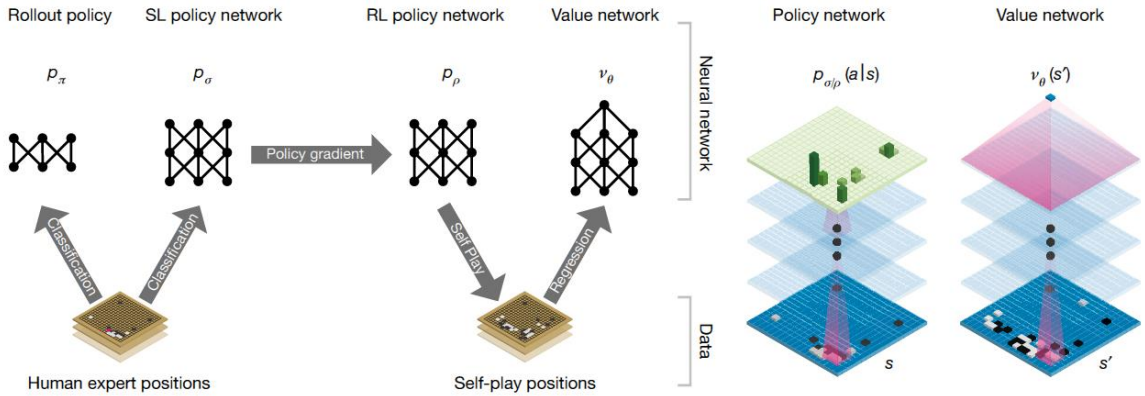
…for video games

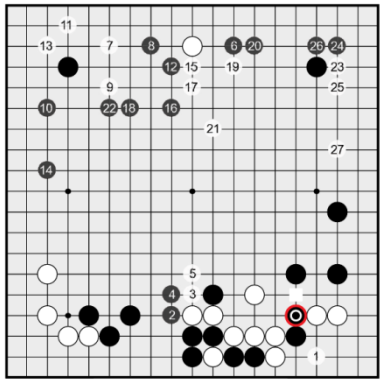




V. Mnih et al., *Nature* **518**, 529 (2015).

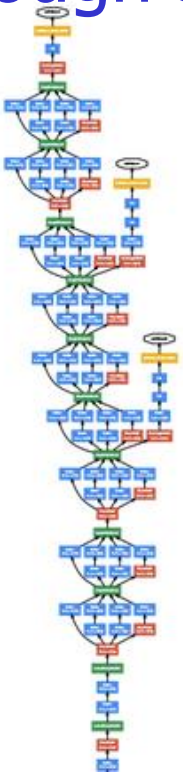
…for Go



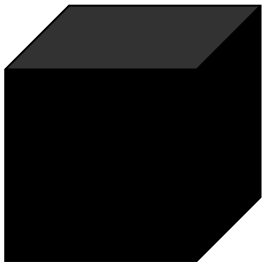


D. Silver et al., *Nature* **529**, 484 (2016).

Too much art, not enough science?



1971



Deep-NN might be a "black box"…

…yet we can (inefficiently) **engineer for performance** because we get a **quantitative metric** (accuracy) from the "black box."

"Rhine River Crossing"

By William Vandivert

1945-04

Prediction: 1945

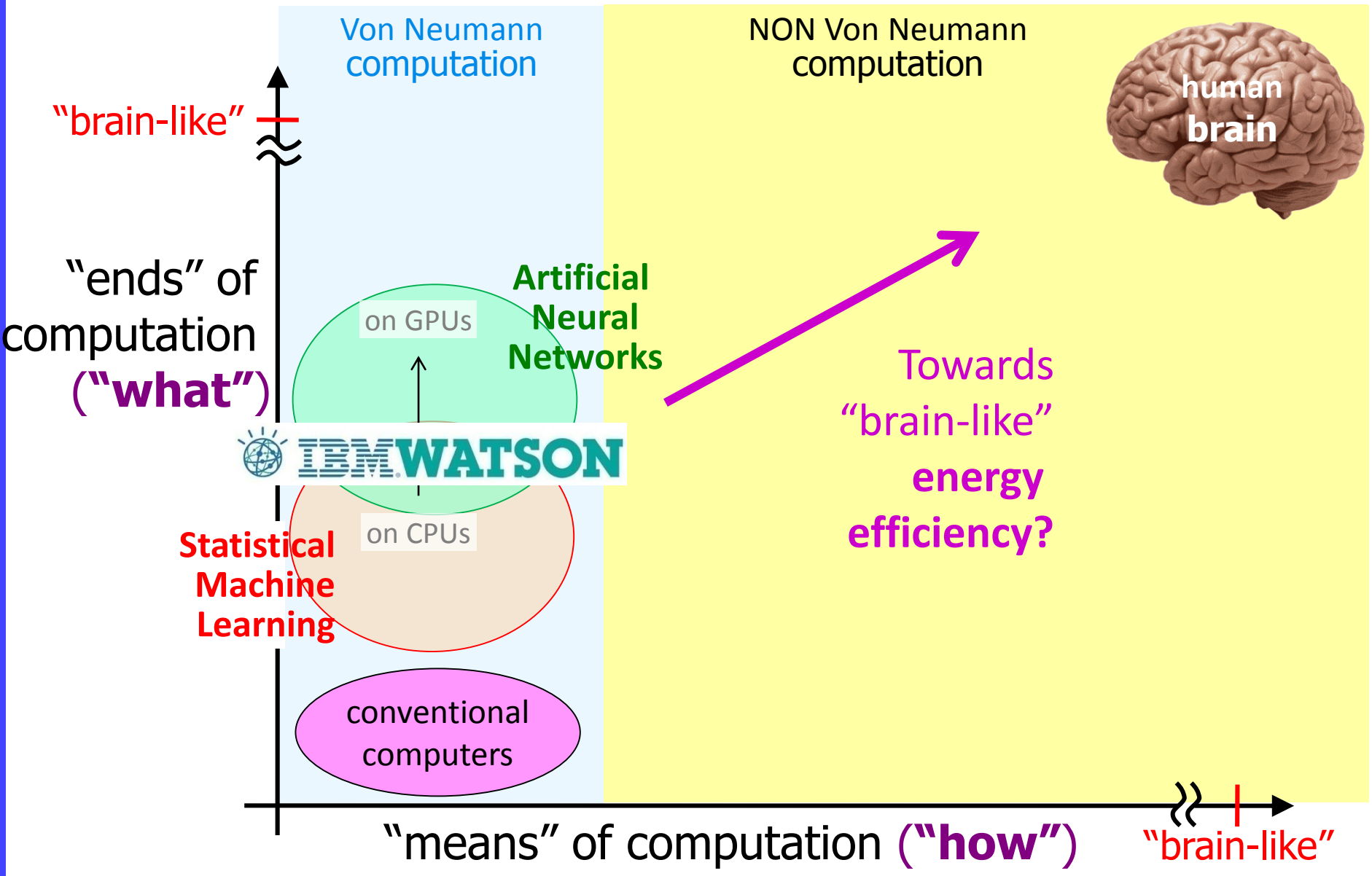
from "Deep learning, big data, and personal devices," Blaise Agüera y Arcas, Google, May 2016 @ SRC/NSF workshop on Intelligent Cognitive Assistants

What are the CS folks doing/ignoring?

- What are the computer scientists up to?
  - Design-based Big Data Analytics – not "neural"
  - Machine Learning
    - Deep Neural Networks
      - pros & cons here
    - Transcending DNN – LSTM, Recurrent NN, Reinforcement Learning
  - Machine Intelligence

- What are the computer scientists missing out on?
  - Too addicted to backpropagation & classification → "gravity well"
  - Robustness in the presence of imperfections/noise
  - Energy efficiency ("Is 0.5% higher accuracy really worth 30x more time & energy?")
  - Spike-based learning techniques
    - **STDP**
  - How to implement "strong" AI → "**machine intelligence**"??

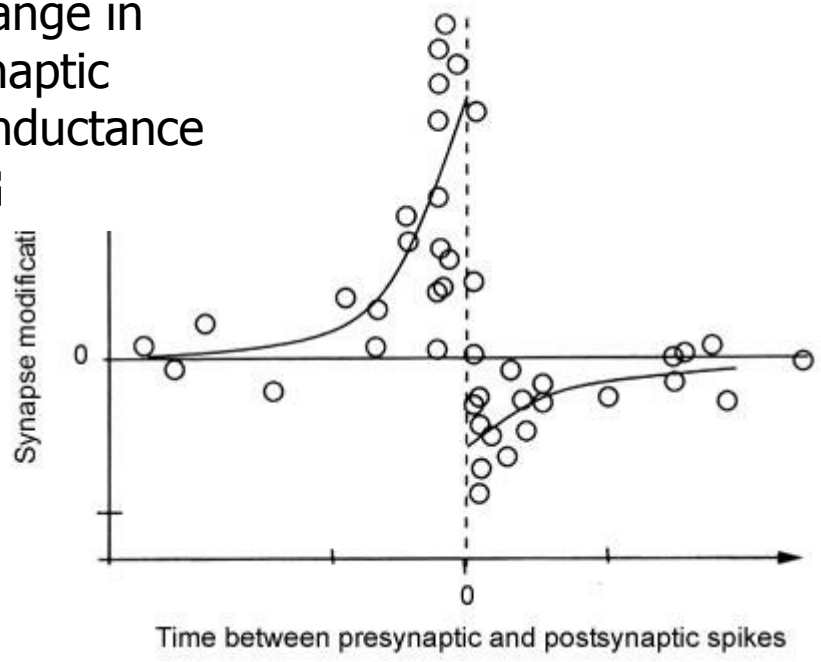
How to get to brain-like energy efficiency?



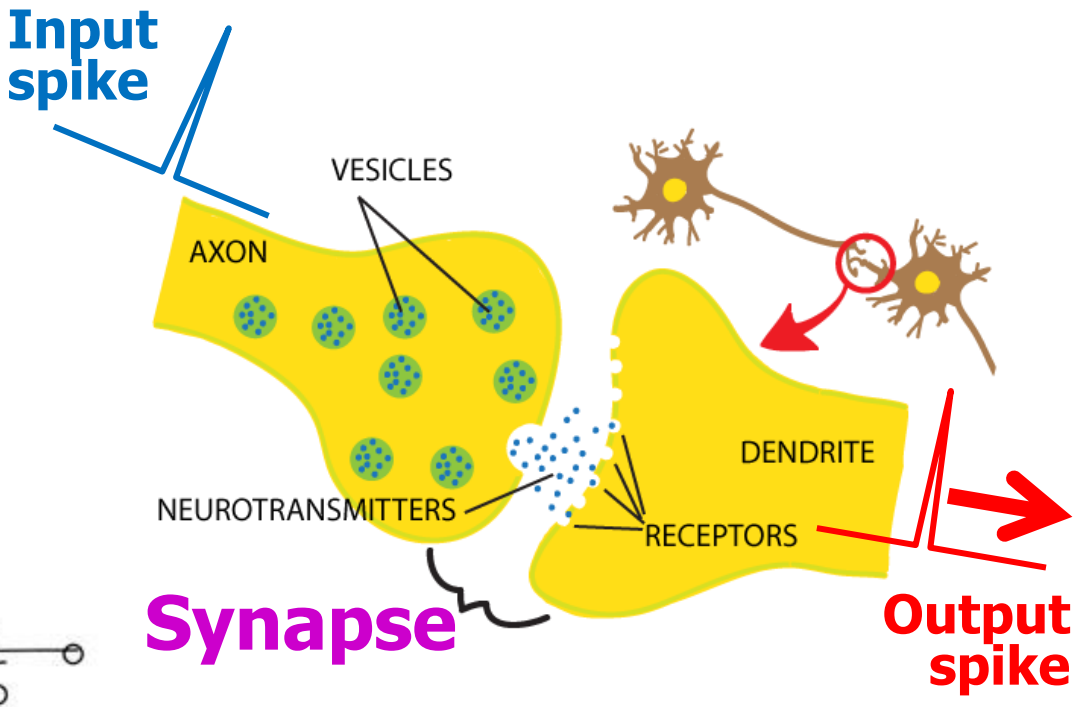
Spike-Timing-Dependent Plasticity

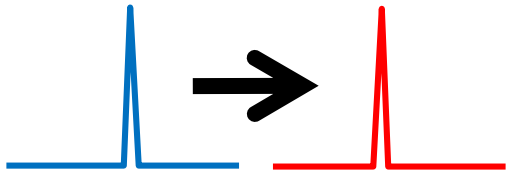
Spike timing dependent plasticity
(Bi and Poo)

Change in Synaptic Conductance
ΔG



G. Bi and M. Poo, *J. Neuroscience*, 18(24), 10464 (1998)





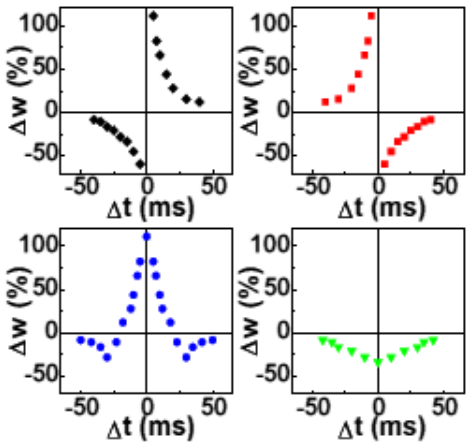
Pre THEN **Post** → STRENGTHEN synapse, G↑

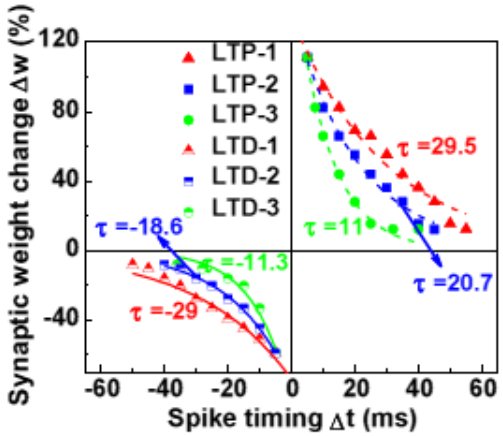
Pre AFTER **Post** → WEAKEN synapse, G↓

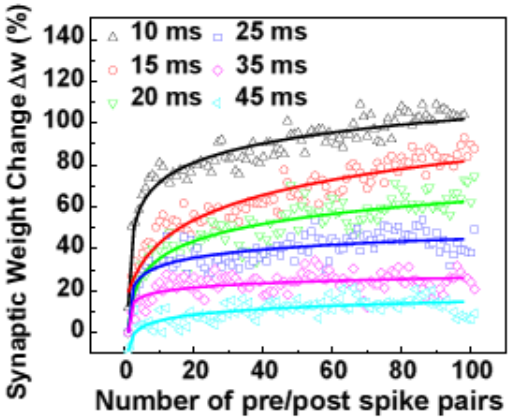
STDP in NVM devices

Nanoscale Memory Can Emulate Biological Synaptic Behavior

STDP (spike-timing-dependent plasticity)







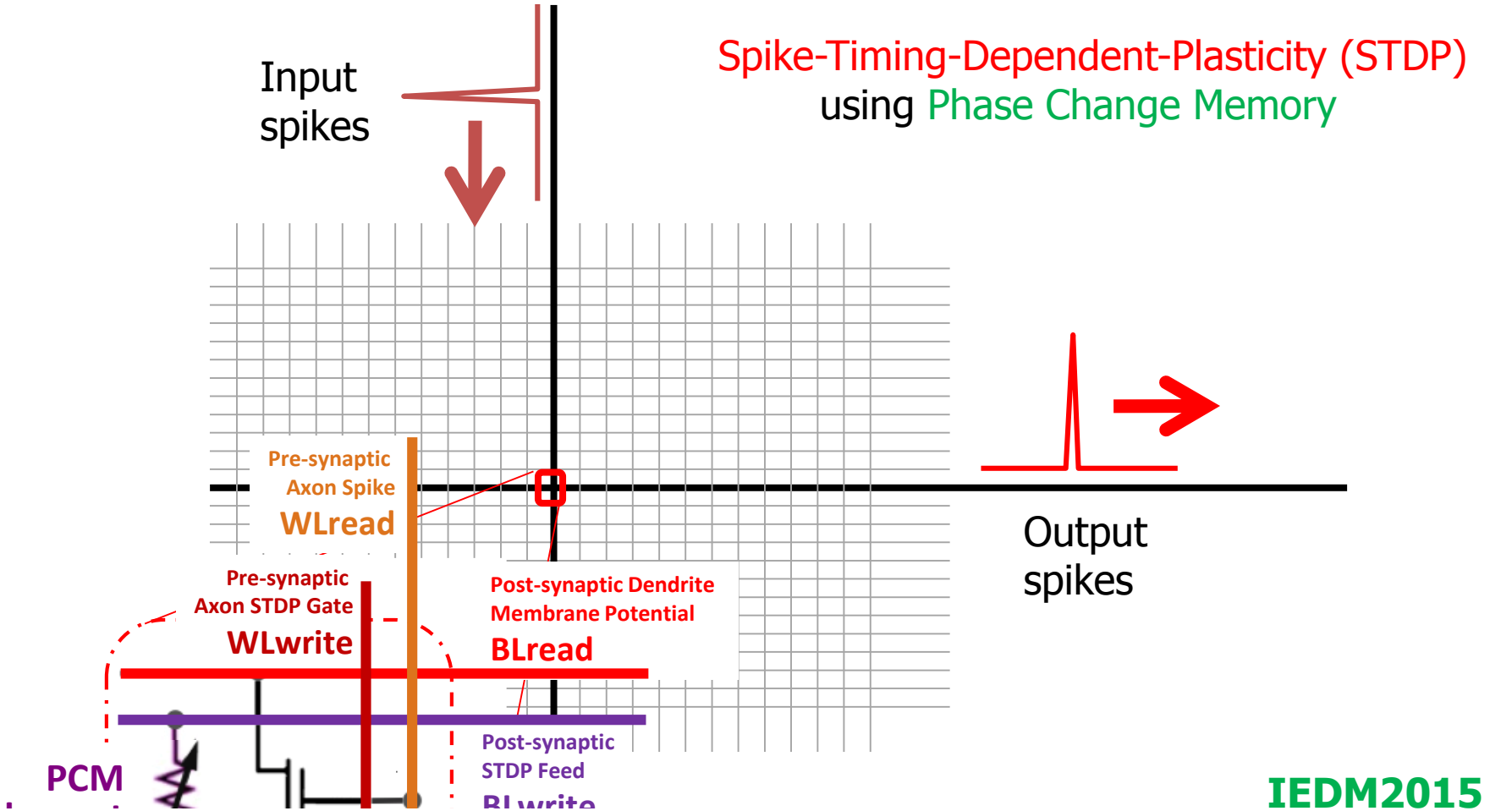
Various STDP kernels

Various time constants

Weight update saturation

Slide from: S. B. Eryilmaz et al., IEDM 2015, T4.1 (2015).
showing work from: D. Kuzum *et al., Nano Lett., p. 2179 (2012)*

2T1R PCM design for **S**pike-**T**iming-**D**ependent-**P**lasticity



IEDM2015

NVM Neuromorphic Core with 64k-cell (256-by-256) Phase Change Memory Synaptic Array with On-Chip Neuron Circuits for Continuous In-Situ Learning

S. Kim, M. Ishii[†], S. Lewis, T. Perri, M. BrightSky, W. Kim, R. Jordan, G. W. Burr[#], N. Sosa, A. Ray, J.-P. Han, C. Miller, K. Hosokawa[†], and C. Lam
IBM T. J. Watson Research Center, 1101 Kitchawan Rd., Yorktown Heights, NY, 10598, USA
[†]IBM Tokyo Research Lab, Tokyo, Japan, [#]IBM Research–Almaden, San Jose, CA, USA.
Tel: +1(914)945-2530, Fax: +1(914)945-4256, email: SangBum.Kim@us.ibm.com

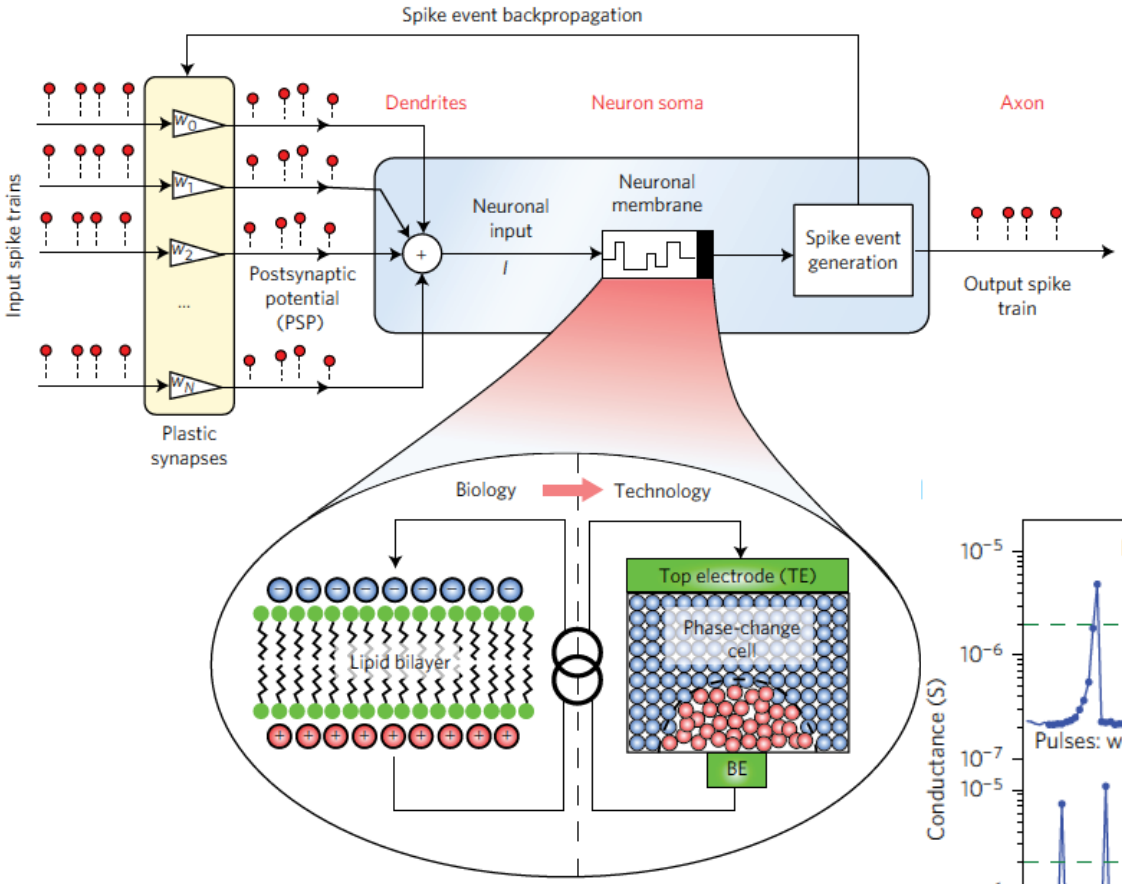
nature nanotechnology

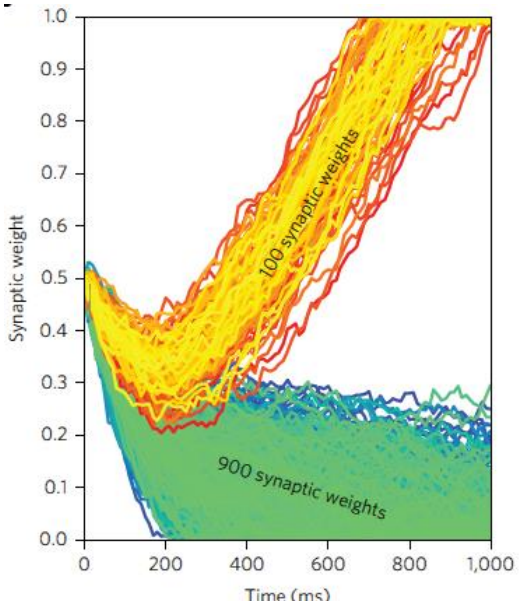
ARTICLES

PUBLISHED ONLINE: 16 MAY 2016 | DOI: 10.1038/NNANO.2016.70

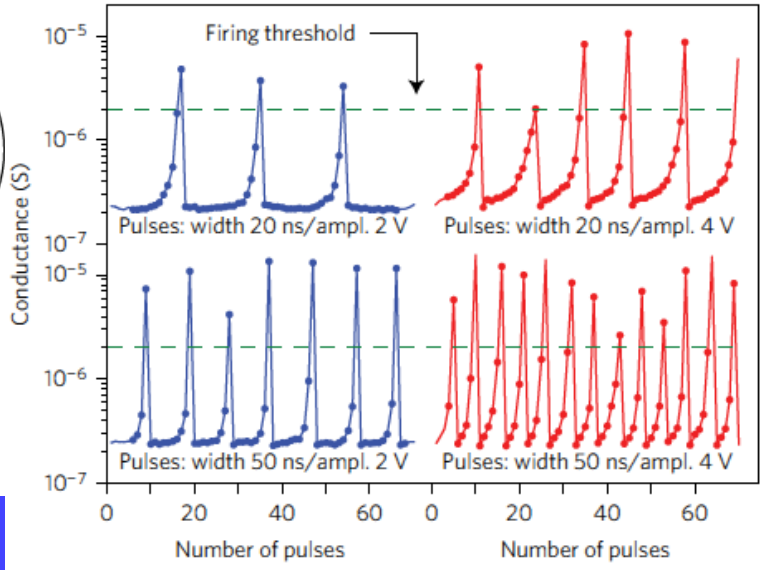
Stochastic phase-change neurons

Tomas Tuma[1]*, Angeliki Pantazi[1], Manuel Le Gallo[1,2], Abu Sebastian[1] and Evangelos Eleftheriou[1]*

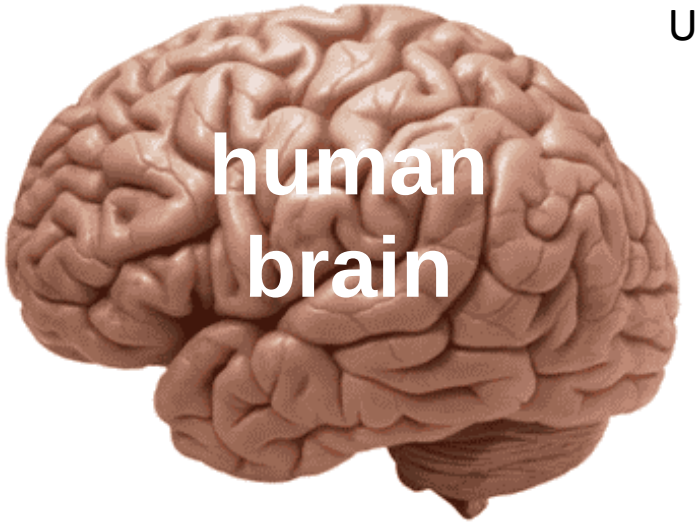




At the steady state, the synapses corresponding to the correlated input streams are potentiated, whereas the synapses corresponding to the uncorrelated input streams are depressed.



Spike-Timing Dependent Plasticity



Uses PART of what the brain exhibits …

~~Electrochemical signal transmission~~

Hierarchy?? (multiple layers)

Local inhibition

~~Stochastic behavior~~

Recurrent connectivity??

Local ~~AND remote~~ connectivity

Lots of neurons

Lots of synapses PER neuron

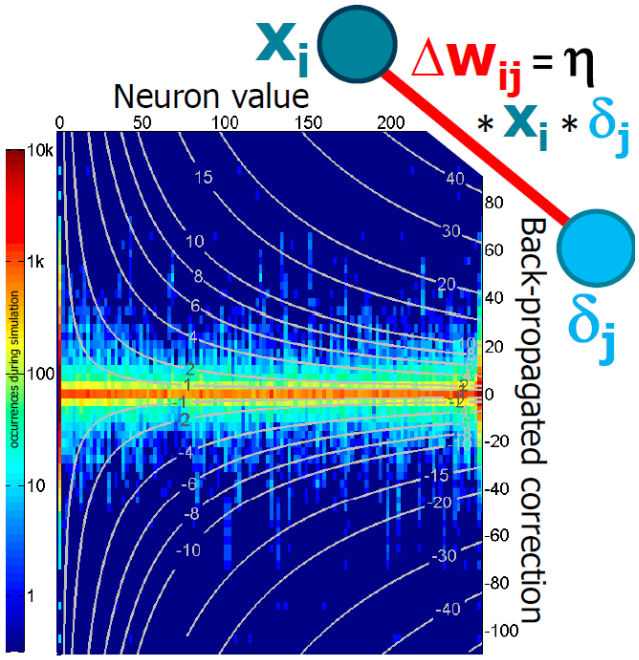
~~Sparse distributed representations~~

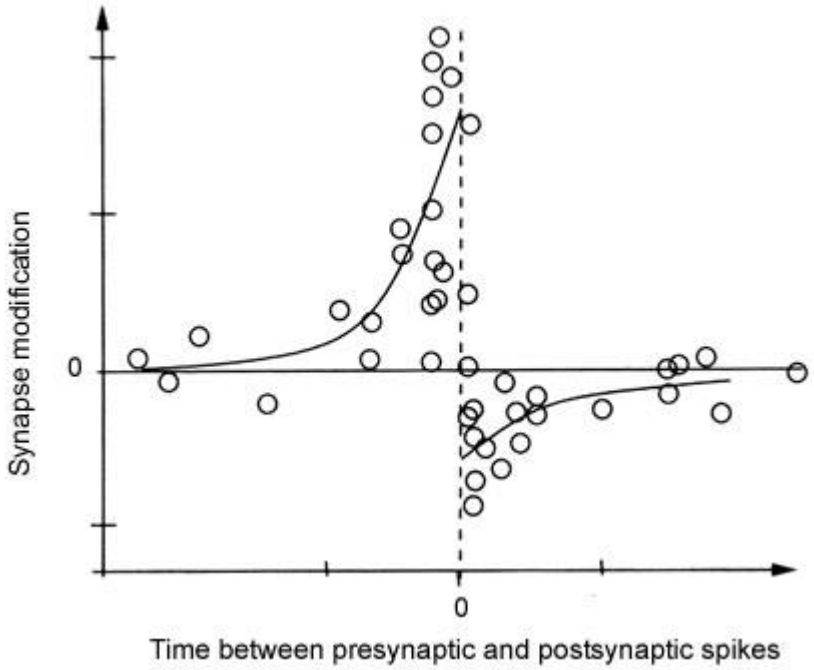
Hebbian synaptic plasticity (cells that fire together, wire together) ~~Backprop~~

Integrate-and-fire in space (many spikes arriving in a vicinity)

Integrate-and-fire in time (many spikes over time)

Backprop vs. Spike-Timing Dependent Plasticity





For BOTH: Instantaneous weight update of each synapse
depends only on information available to
TWO **local** neurons: immediate upstream & downstream neurons

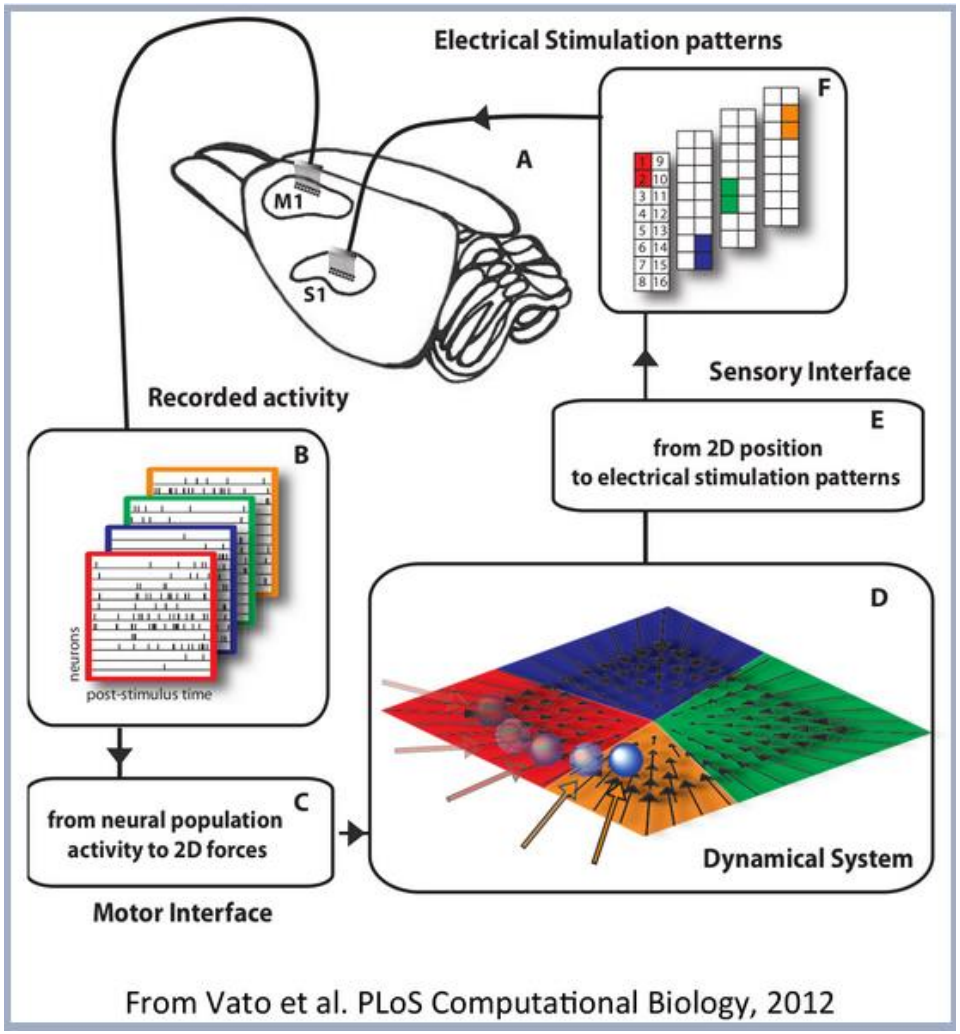
But only for backprop are all those weight updates working together coherently
across the entire network towards a common goal
→ **lower error on the data-examples in question**

Backprop is **scalable**: it's easy to make network bigger & it tends to get better…

Need to find an architecture/global-algorithm that can harness
an STDP-like local learning rule for **robust & scalable learning**

Brain-machine interfaces

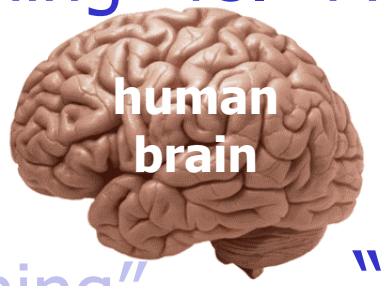
Brain-machine interfaces & circuits @ biological time-scales (G. Indiveri, E. Chicca, etc.)



http://www.ini.uzh.ch/research/36620

"Deep Machine Learning" vs. "Machine Intelligence"

"Brain-inspired" computing
(1940's understanding of the brain)



"Brain-inspired" computing
(modern understanding of the brain)

"Deep Machine Learning"

solving a specific task on **labeled** data by defining & optimizing an objective function

PRO:

- can follow gradient descent thru backpropagation
  → convergence to "good" solutions
- mapping to matrix manipulation → GPUs!!
- great progress in ML thanks to competitions
  - Many datasets created
  - Focus on **quantifying** performance
- algorithm is **scalable:**
  more resources → better performance

CON:

- we're sure the brain doesn't do backpropagation
- can only handle **static, labelled** data
- insistence on quantifying performance may now be stifling innovation

"Machine Intelligence"

flexible systems that continuously learn from **unlabeled** data, and that perform (motor) actions, predict consequences of those actions, and then plan ahead to reach goals

PRO:

- we're sure this is what the brain does
- MI should be able to handle **unlabelled** & **temporal** data
- MI should enable continuous learning

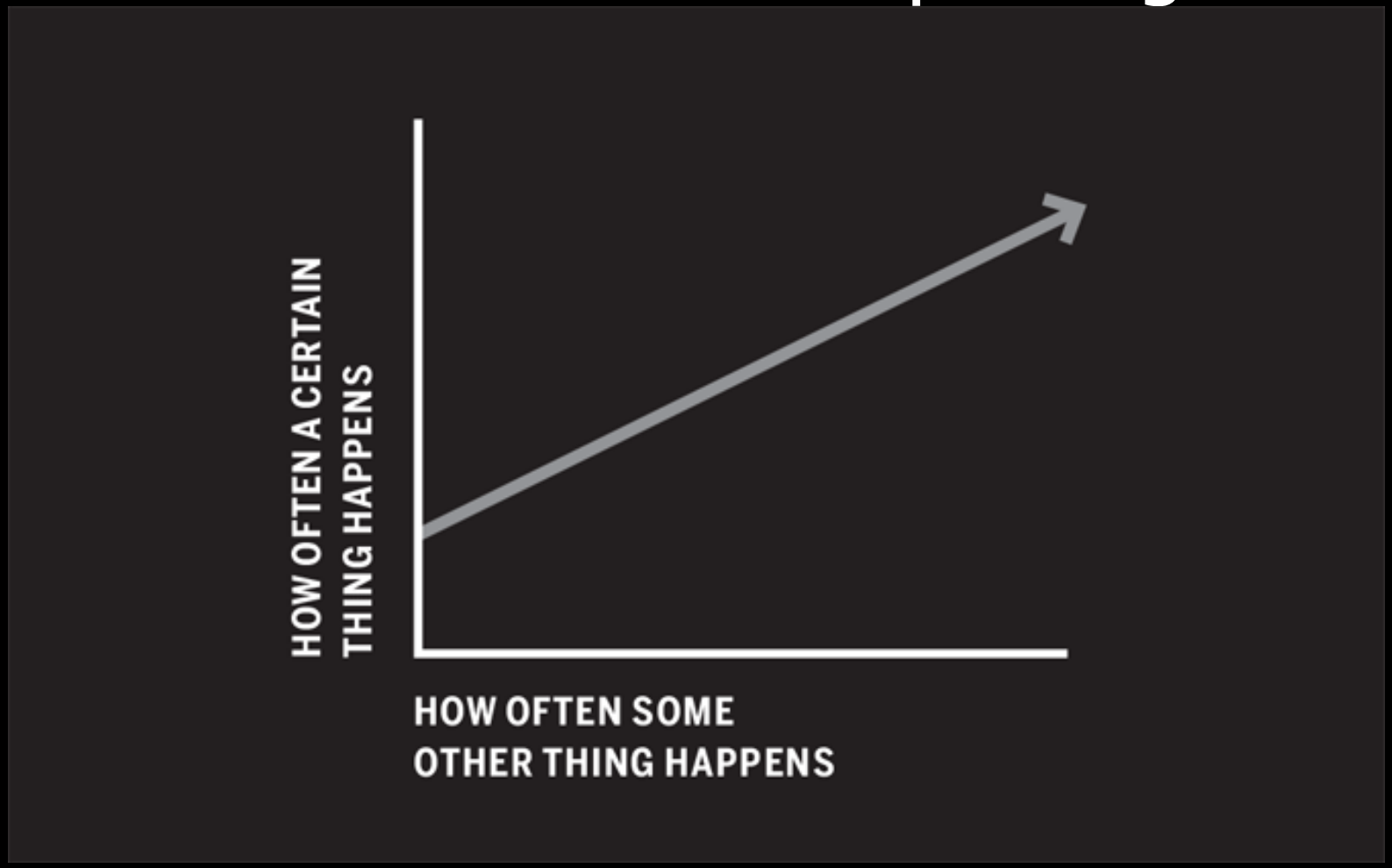
CON:

- we don't know (yet) how the brain guarantees robust, stable convergence in learning
- we have to figure out how to appropriately **quantify** "performance"

Easy Question: what is this man carrying?
Harder Question: What makes this scene unusual?

Gary Marcus, NYU (May2016 Workshop on Intelligent Cognitive Assistants)

Kids vs dominant AI paradigm



In deep learning, it's all correlation, and no causation

Gary Marcus, NYU (May2016 Workshop on Intelligent Cognitive Assistants)

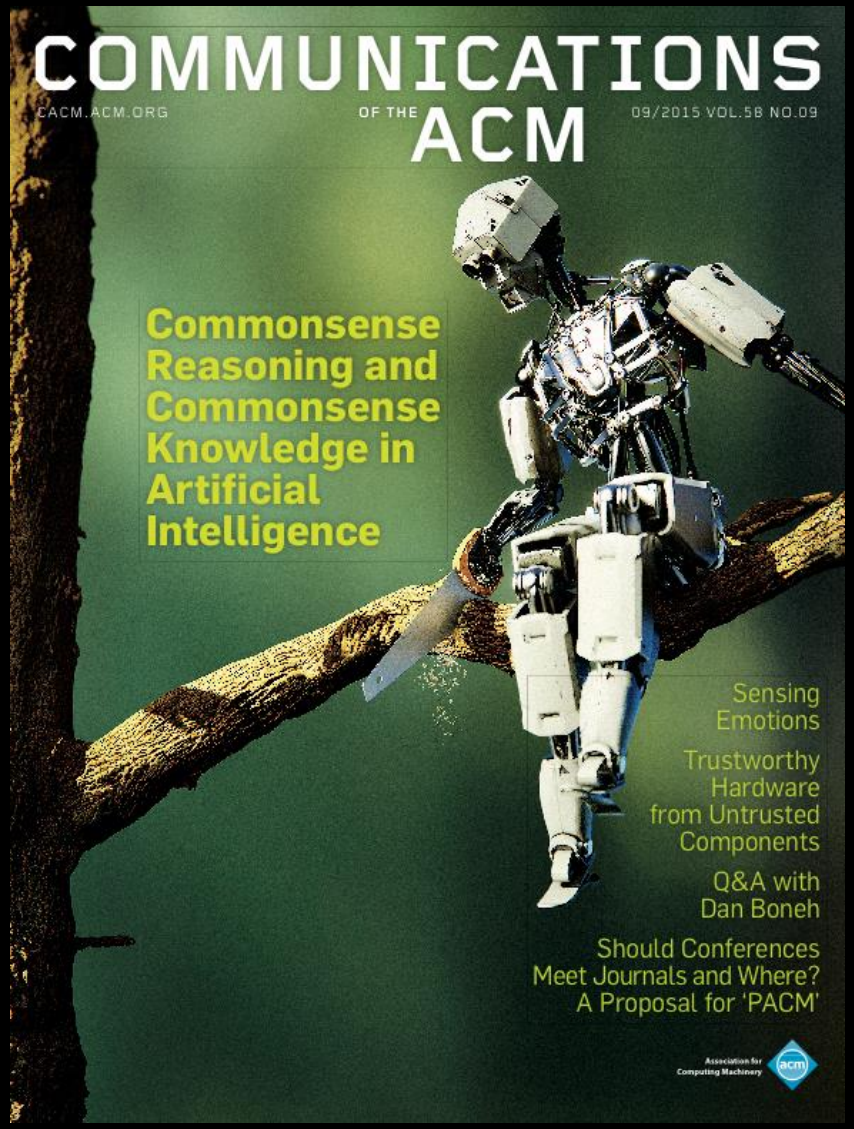
Children don't just care about correlations

- They want to know WHY

- Why is the sky blue?
- How do birds fly?
- Where do babies come from?

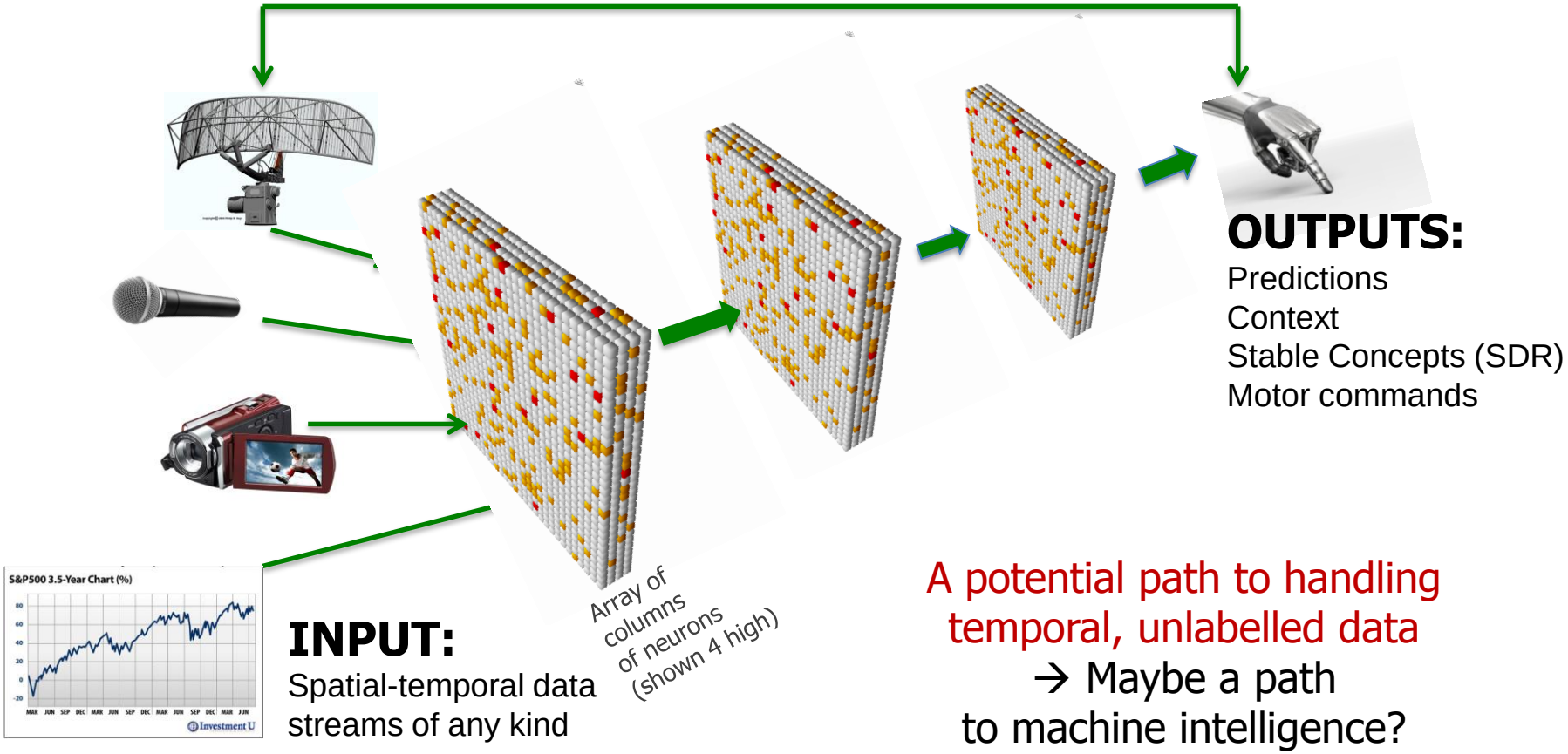
Gary Marcus, NYU (May2016 Workshop on Intelligent Cognitive Assistants)

this quickly leads them to a rich common-sense understanding of the world



Gary Marcus, NYU (May2016 Workshop on Intelligent Cognitive Assistants)

Machine Intelligence based on sequences of Sparse Distributed Representations



INPUT:
Spatial-temporal data streams of any kind

OUTPUTS:
Predictions
Context
Stable Concepts (SDR)
Motor commands

A potential path to handling temporal, unlabelled data
→ Maybe a path to machine intelligence?

"**C**ontext-**A**ware **L**earning"

winfriedwilcke@us.ibm.com

Requires HUGE fanout:
many POTENTIAL synapses
(internally analog, externally binary)

Hierarchical Temporal Memory

Map **temporal data** into sequences of **S**parse **D**istributed **R**epresentations

4) Temporal Pooler

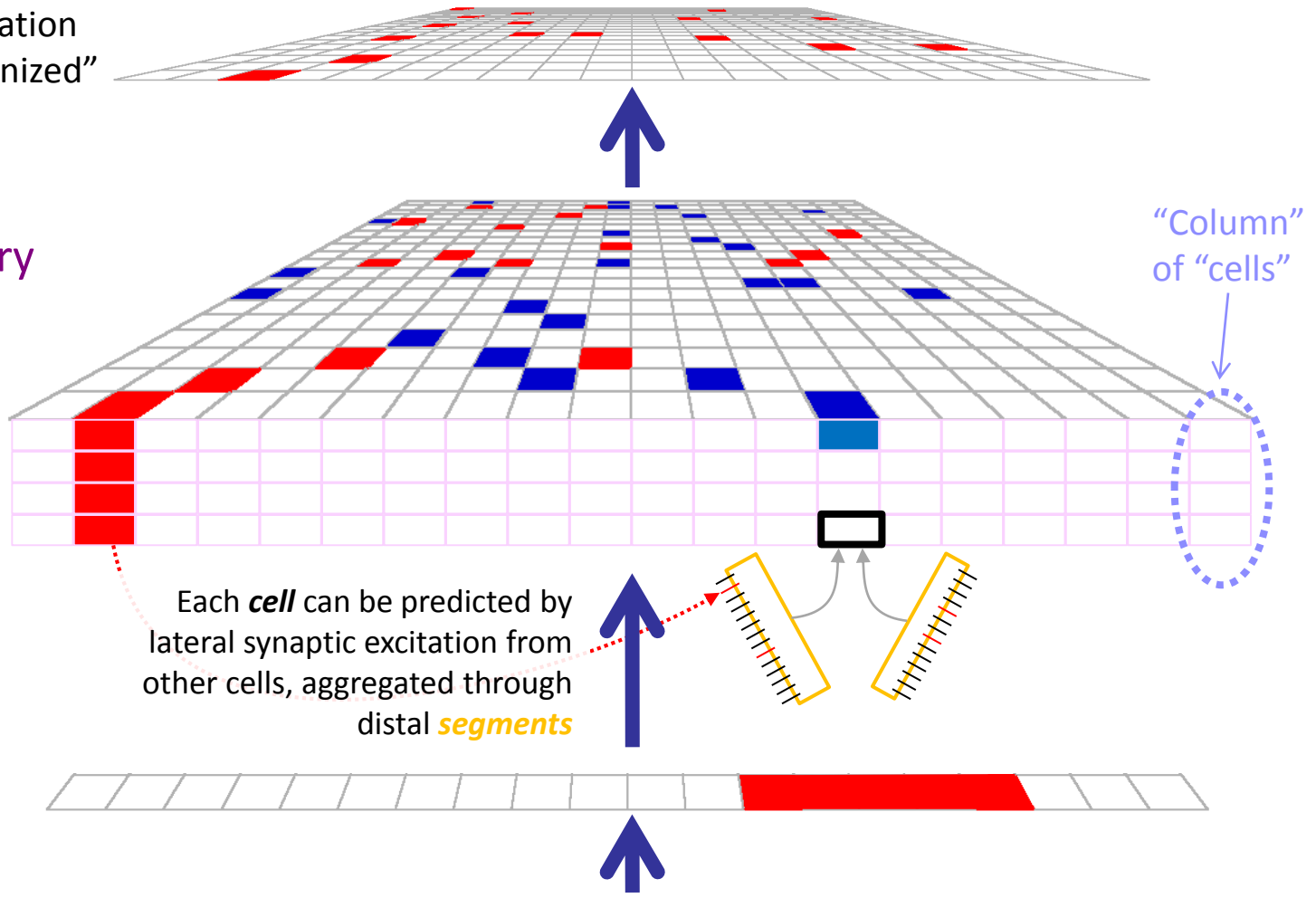
- invariant representation (SDR) of each "recognized" sequence

3) Temporal Memory

- predict the **"next" SDR** in the sequence given **"this" SDR**

2) Spatial Pooler

- map each input excitation to an appropriate SDR of constant sparsity

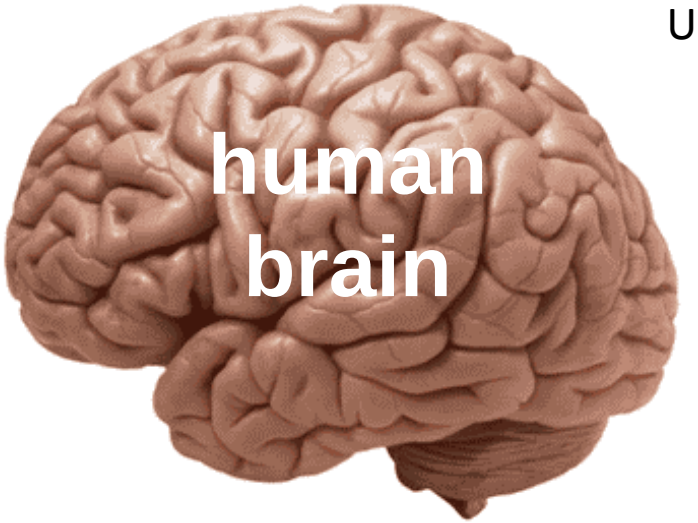


1) Input Data

- encode input data into input neurons

J. Hawkins, S. Blakeslee. ***On intelligence.*** Macmillan, 2007.

Hierarchical Temporal Memory



Uses PART of what the brain exhibits …

~~Electrochemical signal transmission~~

Hierarchy (multiple layers)

Local inhibition

~~Stochastic behavior~~

Recurrent connectivity

Local AND remote **connectivity**

Lots of neurons

Lots of synapses PER neuron

Sparse-distributed representations

~~**Hebbian**~~ **synaptic plasticity** ~~(cells that fire together, wire together)~~

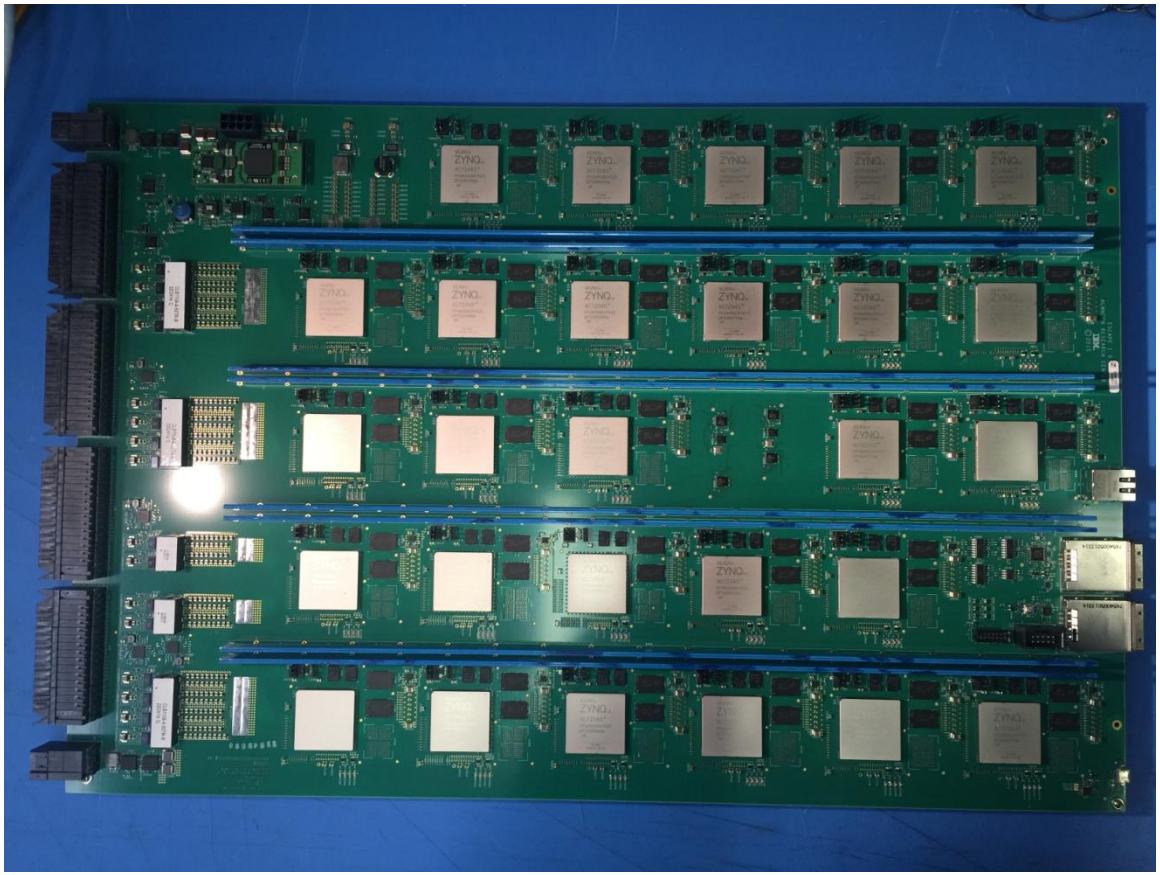
"permanence"

Integrate-and-fire in space (many spikes arriving in a vicinity)

"dendritic segments"

~~**Integrate-and-fire in time** (many spikes over time)~~

ESCAPE: Accelerator for Machine Intelligence



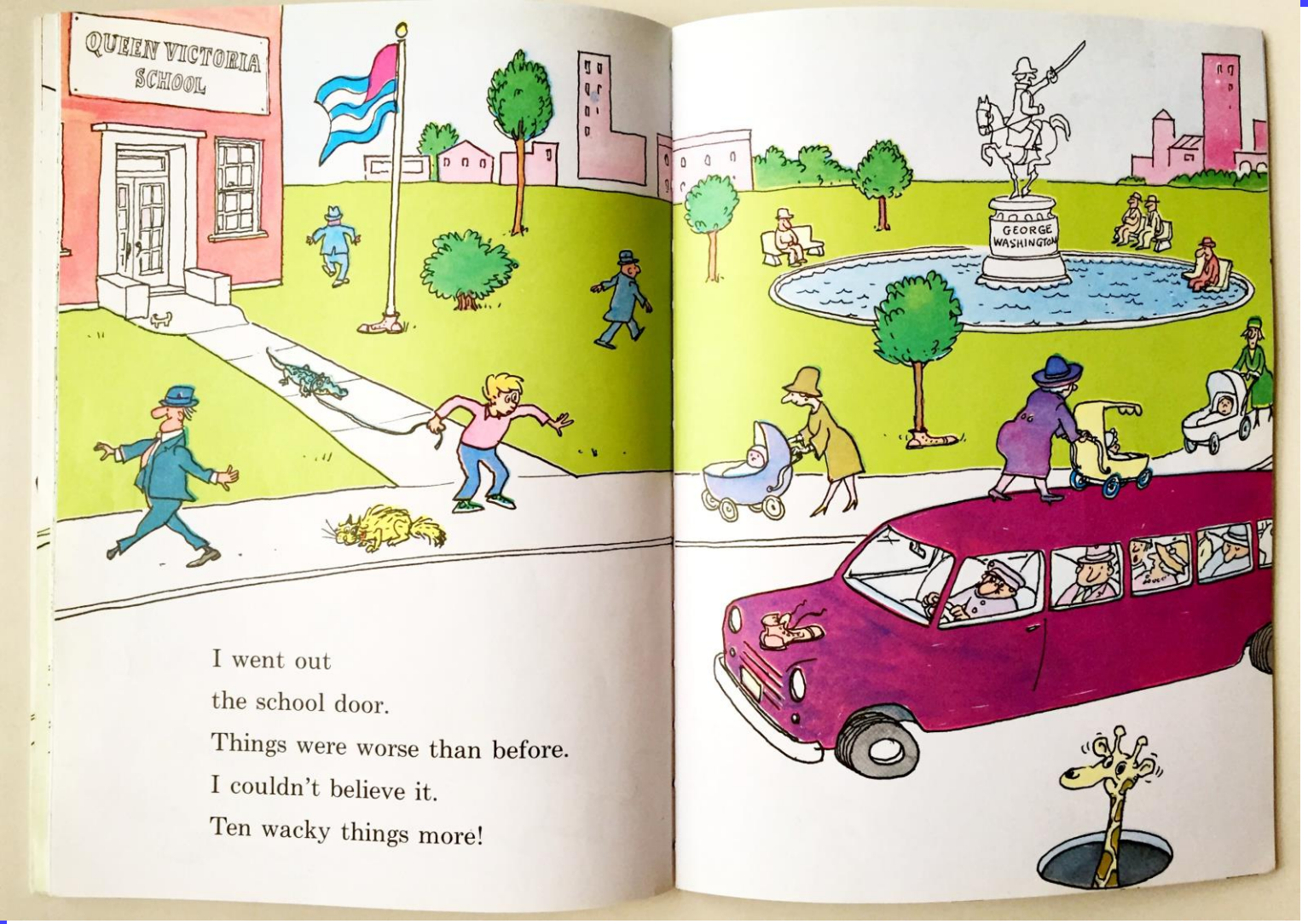
1000 node parallel system
- Xilinx Zynq dual A9 core + FPGA,
- 1 GB RAM,
- 6x2 bi-di high-speed links
  - system topology: 3D mesh
  - very high bandwidth

Dual purpose
- scale up HTM simulations
  - to > 10^8 realistic neurons
- platform for design
  - of waferscale system

Winfried Wilcke,
IBM Almaden
winfriedwilcke@us.ibm.com

- ESCAPE will consist of 37 of these cards
- 27 FPGA/ARM nodes/card
- Very large & complex card, 55 cm x 46 cm, 46 layers (!)
- Very fast network (hundreds of GB/sec)

Common Sense



"Wacky Wednesday"

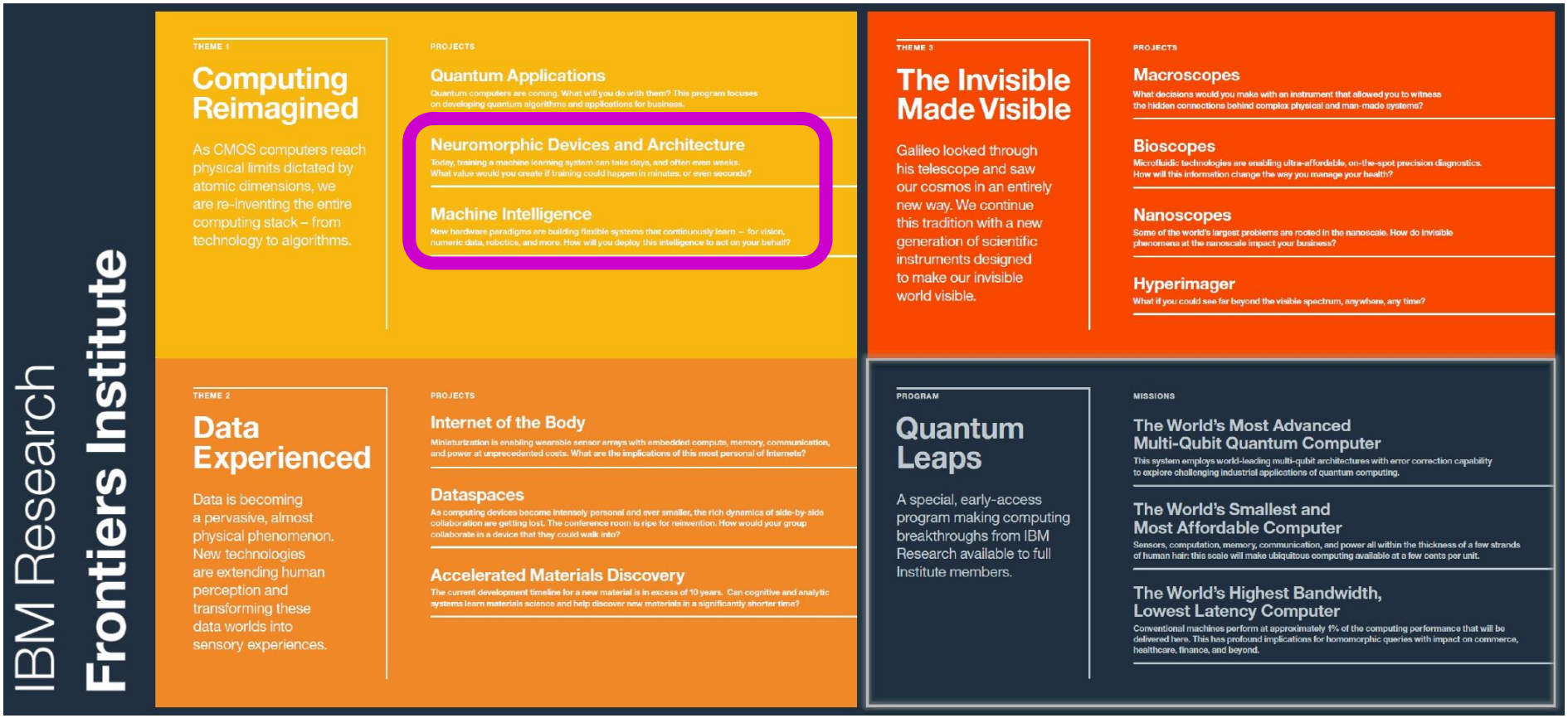
Children's book by Dr. Seuss

Random House Publishing, 1974

Need a **robust analog metric** to quantize performance on such tasks – "we'll know it when we see it" will not suffice

Without such a metric, it will be difficult to combine the efforts of many researchers & perfect these systems through many tiny incremental improvements…

IBM Research Frontiers Institute



For more information: Sudhir Gowda,
Associate Director, IBM Research Frontiers Institute
gowda@us.ibm.com
www.research.ibm.com/frontiers

How can hardware play a role?

- Where can hardware (devices, circuits, systems) play a role, and what's been done so far?
  - Approximate computing
  - Stochastic computing
  - Hardware for communicating between VN/Non-VN cores
    - Address Event Representation
    - TrueNorth, SpiNNaker, etc.
  - Crossbar Memory for
    - accelerating backpropagation
    - implementing STDP
      - role for stochasticity
  - Hardware for large-scale realistic brain simulations
    - help understand epilepsy, Parkinson's , Alzheimer's
  - Brain-machine interfaces
  - Machine Intelligence
    - ESCAPE system for Machine Intelligence

About the brain

- What **DON'T we like** about how the brain does computing?
  - Not good at providing exact/precise answers
    - always a non-zero chance of being "wrong"
  - Cognitive biases due to numerous "shortcuts"

→→→

Cognitive biases

- ***Ambiguity effect*** – talking without knowing (exactly) what you're talking about
- ***Focalism (Availability heuristic)*** – weighing the first piece of incoming (recalled) information too much
- ***Illusory truth-effect –*** repeating something doesn't make it true
- ***Myside bias*** – interpreting info in a way which confirms our existing beliefs
- ***Selective perception –*** ignoring info that contradicts our existing beliefs
- ***Base rate fallacy –*** brain ignores general information, focuses a specific example
- ***Belief bias*** – we connect conclusions to premises based on their credibility, even if that conclusion is not validly supported by **those** premises.
- ***Choice-supportive bias –*** psychological validation of our previous decisions
- ***The familiarity principle*** – we like the things we have been repeatedly exposed to
- ***Social desirability bias –*** we calibrate response to receive a positive evaluation

Social biases:

- ***Actor-observer asymmetry –*** our negative behavior → reflects unique situation; others' negative behavior → general characteristic of their personality.
- ***Dunning-Kruger effect –*** less competence → more confidence; minimum confidence occurs at medium competence – you finally know enough to know you don't know everything
- ***False consensus effect*** – individuals consider themselves "normal," thus assume others must think like they do
- ***The illusion of asymmetric insight*** –people tend to believe that their knowledge of others is many times more meaningful and broad than other people's knowledge of them

www.zmescience.com/science/cognitive-biases-list/

About the brain

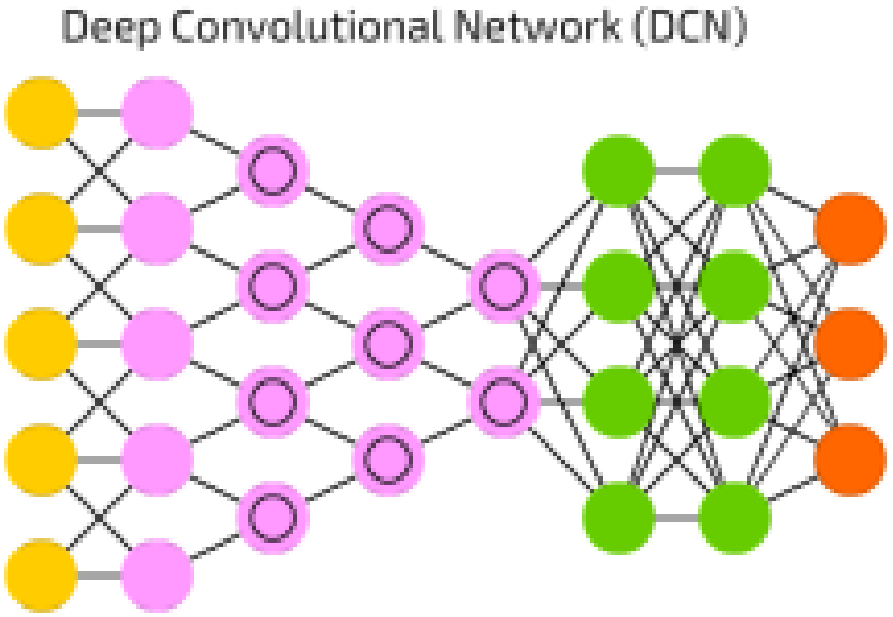
- What **DON'T we like** about how the brain does computing?
  - Not good at providing exact/precise answers
    - always a non-zero chance of being "wrong"
  - Cognitive biases due to numerous "shortcuts"
  - Learning procedure is inefficient (can't transfer learned weights)
  - System is too holistic, so "debugging" is a nightmare
    - we're forced to understand the WHOLE system, because the system is not sufficiently modular

- **What are we going to miss** about the Von-Neumann architecture?
  - General purpose → one piece of hardware, many customers/users
    - Programmable – often can address problems not considered by designers
  - Provides precise, reliable, repeatable answers
  - Design is inherently modular
    - It's OK to have many domain experts who don't/can't comprehend whole system
    - Input/output/requirements of each module can be specified readily

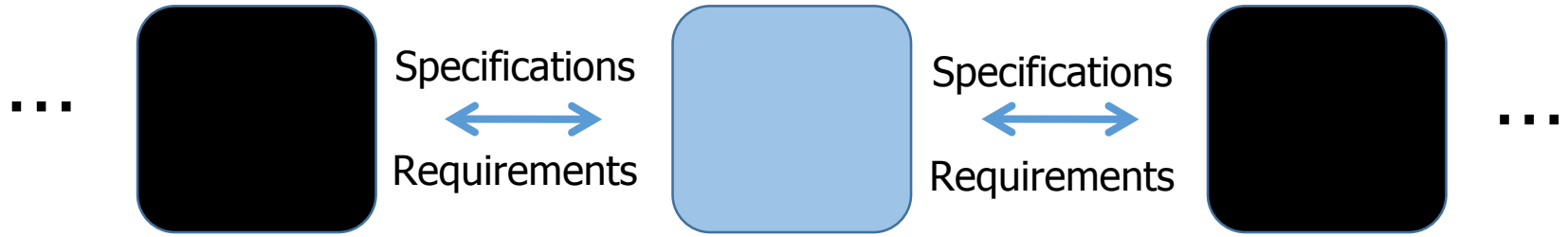
"Some" imperfections are OK – great!

But basic engineering – like **identifying** how many would be "too many" imperfections – will **not be easy.**

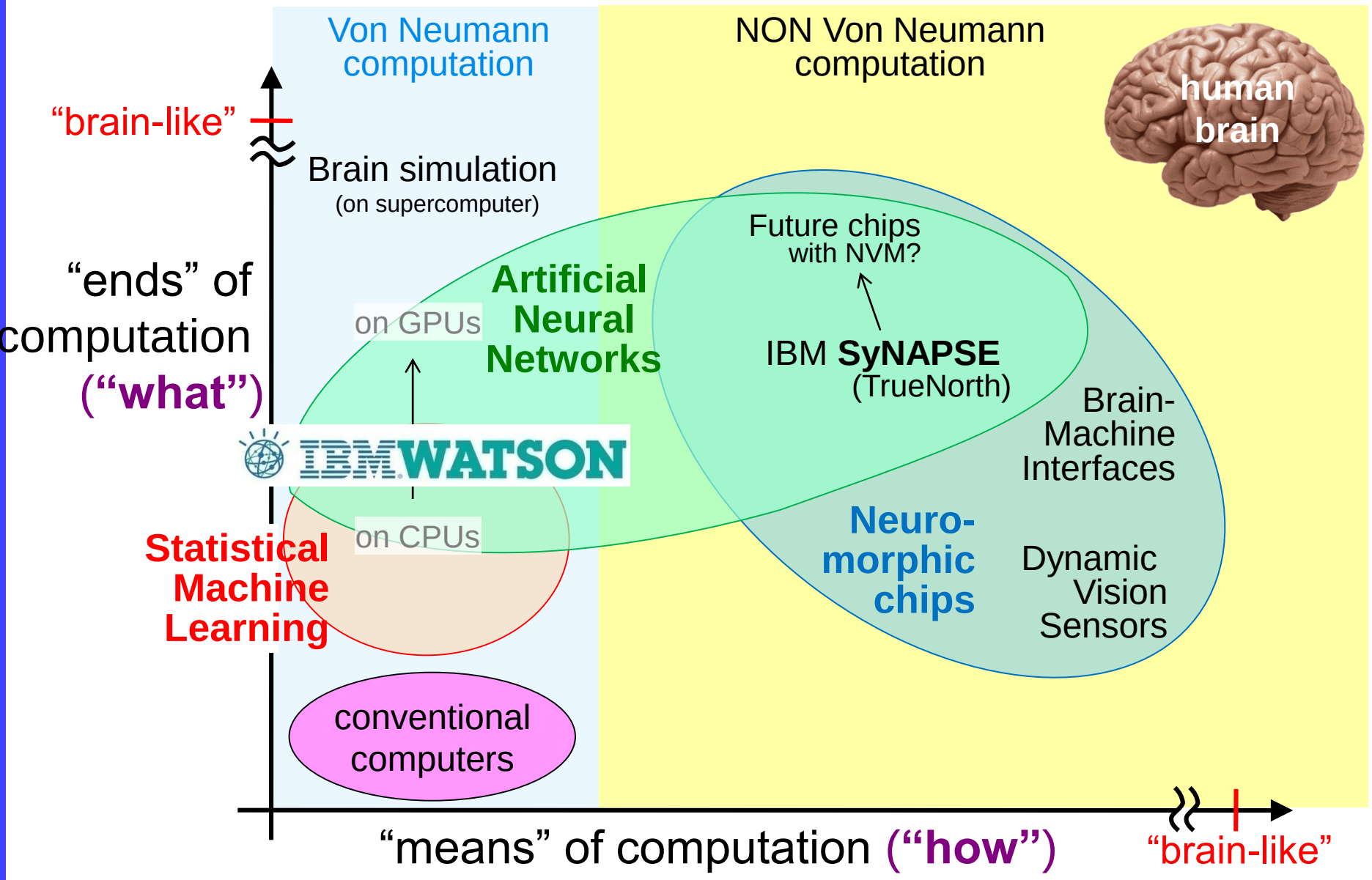
Big data sets →



www.datasciencecentral.com/profiles/blogs/concise-visual-summary-of-deep-learning-architectures



A roadmap of brain-inspired computing



Applications

Anything that requires

- Comprehension: **natural language,** vision of **complex scenes**
- understanding of **context**
- **prediction**/consequences of actions

- "Big Data" analytics
- multi-modal sensors ("electronic nose")
- early-alert sensor networks (health-care, policing, tsunamis, etc.)
- personal assistants (Siri, but predictive/proactive)
- "guide glasses" for the blind
- self-driving cars
- autonomous robots
  - emergency
  - search & rescue
  - exploration
  - military

Challenges

- nature used evolution – we use engineering & design
  - need "transparent" programming model
  - how can we "modularize" design, enabling highly complex systems?

- ethical issues
  - "**network effect**": systems built with lots of data will perform better
    → a few large companies could dominate
    - how to **maintain trust** between consumers & suppliers of cognitive computing?

  - a large number of jobs (47% [1]) could be affected by "computerisation"
    - what will be the consequences of this shift?
      in politics, economics, sociology

  - "**super-intelligence**": Artificial Intelligence systems that are "smarter" than we are
    - until we really understand consciousness,
      can we be sure it's not just a function of network size?
    - if we use the today's AI to design tomorrow's, what prevents "runaway" AI?

[1] C. B. Frey and M. A. Osborne, "The Future of Employment: How susceptible are jobs to computerisation?" Sept. 2013

Non-Von Neumann opportunities

- **Forward inference engines** → potentially huge volumes
  - On smartphones (TrueNorth → NorthPole)
  - Digital first → later analog opportunity too or no…?

- **On-chip learning** → lower volumes but potentially an essential enabling technology
  - **Digital** w/ reduced precision

  - **Analog** IF…
    1. … peripheral circuitry supports **massive parallelism → speed-up** over GPU
    2. … NVM devices support **linear conductance change → same accuracy** as GPU

- **STDP-based NN:** (e.g., spikes for learning not just communication)
  - Killer app that requires learning-from-timing
  - Architecture/global-algorithm that harnesses STDP-like local learning rule for **robust learning** to support/enable above killer-app

- **Machine Intelligence:**
  - Significant algorithm development needed → **too early for crossbar device arrays!**

Device researchers who want to have an impact will likely also need to **learn/know/advance** the **circuits/systems/algorithms** module(s)

Conclusions

- Brain-inspired computing …
  - is already here – "brain-like" computation on conventional computers
    - will attain better energy-efficiency through more "brain-like" neuromorphic chips
    - is unlikely to reach the full energy efficiency of the human brain anytime soon
      → any AI with complexity similar to the human-brain would not be portable
- Moore's law will end soon → future improvements in computers will come MOSTLY from **improvements in architecture**, NOT from better, or from more densely-packed, devices
- Opportunities from…
  - neuromorphic circuitry to …
    - … understand network dynamics
    - … interface with the brain (prosthetics, etc.)
  - **accelerating** Deep Learning…
  - **transcending** Deep Learning…

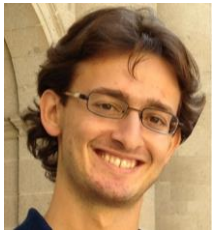
Thank you for your attention!

Acknowledgements


Geoffrey Burr


Pritish Narayanan


Bob Shelby


Stefano Ambrogio


Hsinyu (Sidney) Tsai


Kohji Hosokawa


Scott Lewis

Other research groups:

Sandia (M. Marinella, S. Agarwal, A. Talin)
ASU (Prof. S. Yu)
UCSB (Prof. D. Strukov)
Stanford (Prof. H.S.P. Wong)

IBM (T. Gokmen & Y. Vlasov)

IMEC, LETI, others

Collaborators:

Prof. Hyunsang Hwang, POSTECH

Prof. Yusuf Leblebici, EPFL

Students:

Junwoo Jang (now at Samsung)

Kibong Moon

Carmelo di Nolfo (now at IBM Almaden)
Irem Boybat (student @ IBM Zurich)
Severin Sidler (student @ IBM Zurich)
Alessandro Fumarola
Martina Bodini
Massimo Giordano

Lucas Sanches (from USP, Brazil)
Nathan Farinha (from USP, Brazil)
Yassine Jaoudi (Benedict College)
Christina Cheng (U. Chicago)
Benjamin Killeen (U. Chicago)

NVM-for-SCM & MIEC : For more information & *acknowledgements*

• K. Virwani, G. W. Burr, **Rohit S. Shenoy**, C. T. Rettner, A. Padilla, T. Topuria, P. M. Rice, G. Ho, R. S. King, K. Nguyen, A. N. Bowers, M. Jurich, M. BrightSky, E. A. Joseph, A. J. Kellock, N. Arellano, B. N. Kurdi and **Kailash Gopalakrishnan**, "Sub-30nm scaling and high-speed operation of fully-confined Access-Devices for 3-D crosspoint memory based on Mixed-Ionic-Electronic-Conduction (MIEC) Materials," ***IEDM Technical Digest***, 2.7, (2012).

• **Geoffrey W. Burr**, **Kumar Virwani**, R. S. Shenoy, **Alvaro Padilla**, M. BrightSky, E. A. Joseph, M. Lofaro, A. J. Kellock, R. S. King, K. Nguyen, A. N. Bowers, M. Jurich, C. T. Rettner, B. Jackson, D. S. Bethune, R. M. Shelby, T. Topuria, N. Arellano, P. M. Rice, **Bulent N. Kurdi**, and K. Gopalakrishnan, "Large-scale (512kbit) integration of Multilayer-ready Access-Devices based on Mixed-Ionic-Electronic-Conduction (MIEC) at 100% yield," ***Symposium on VLSI Technology***, T5.4, (2012).

• R. S. Shenoy, K. Gopalakrishnan, **Bryan Jackson**, K. Virwani, G. W. Burr, C. T. Rettner, A. Padilla, **Don S. Bethune**, R. M. Shelby, A. J. Kellock, M. Breitwisch, E. A. Joseph, R. Dasaka, R. S. King, K. Nguyen, A. N. Bowers, M. Jurich, A. M. Friz, T. Topuria, P. M. Rice, and B. N. Kurdi, "Endurance and Scaling Trends of Novel Access-Devices for Multi-Layer Crosspoint Memory based on Mixed Ionic Electronic Conduction (MIEC) Materials," ***Symposium on VLSI Technology***, T5B-1, (2011).

• K. Gopalakrishnan, R. S. Shenoy, C. T. Rettner, K. Virwani, Don S. Bethune, R. M. Shelby, G. W. Burr, A. J. Kellock, R. S. King, K. Nguyen, A. N. Bowers, M. Jurich, B. Jackson, A. M. Friz, T. Topuria, P. M. Rice, and B. N. Kurdi, "Highly-Scalable Novel Access Device based on Mixed Ionic Electronic Conduction (MIEC) Materials for High Density Phase Change Memory (PCM) Arrays," ***Symposium on VLSI Technology***, 19.4, (2010).

• G. W. Burr, **Matt J. Breitwisch**, **Michele Franceschini**, **Davide Garetto**, K. Gopalakrishnan, B. Jackson, B. Kurdi, C. Lam, **Luis A. Lastras**, A. Padilla, **Bipin Rajendran**, S. Raoux, and R. Shenoy, "Phase change memory technology," ***Journal of Vacuum Science & Technology B***, **28**(2), 223-262, (2010).

• G. W. Burr, B. N. Kurdi, J. C. Scott, C. H. Lam, K. Gopalakrishnan, and R. S. Shenoy, "An overview of candidate device technologies for Storage-Class Memory," ***IBM Journal of Research and Development***, **52**(4/5), 449 (2008).

• S. Raoux, G. W. Burr, M. J. Breitwisch, C. T. Rettner, Y. Chen, R. M. Shelby, M. Salinga, D. Krebs, S. Chen, H. L. Lung, and C. H. Lam, "Phase-change random access memory — a scalable technology," ***IBM Journal of Research and Development***, **52**(4/5), 465,, (2008).

• **Rich Freitas** and **Winfried Wilcke**, "Storage Class Memory, the next storage system technology," ***IBM Journal of Research and Development***, **52**(4/5), 439, (2008).

• **Yi-Chou Chen**, **Charlie T. Rettner**, **Simone Raoux**, G. W. Burr, S. H. Chen, **R. M. (Bob) Shelby**, M. Salinga, W. P. Risk, T. D. Happ, G. M. McClelland, M. Breitwisch, A. Schrott, J. B. Philipp, M. H. Lee, R. Cheek, T. Nirschl, M. Lamorey, C. F. Chen, E. Joseph, S. Zaidi, B. Yee, H. L. Lung, R. Bergmann, and **Chung Lam**, "Ultra-Thin Phase-Change Bridge Memory Device Using GeSb," ***IEDM Technical Digest***, paper S30P3, (2006).

http://researcher.ibm.com, search for "Burr" or "Storage Class Memory"

Brain-like Computing References

- www.hpcwire.com/2013/12/11/hpc-progress-free-lunch/#/
- G. Baccarani, M. R. Wordeman, and R. Dennard, IEEE Trans. Electr. Dev., **ED-31**(4), 452 (1984)
- en.wikipedia.org/wiki/Place_cell
- www.ibm.com/smarterplanet/us/en/ibmwatson/index.html
- M. Minsky and S. Papert. *Perceptrons* (1968); D. Rumelhart, Parallel Distributed Processing, MIT Press (1986). Krizhevsky, Sutskever, and Hinton, NIPS 2012
- Y. Bengio, NIPS 2015
- V. Mnih et al., *Nature* **518**, 529 (2015); D. Silver et al., *Nature* **529**, 484 (2016).
- S. Gupta et al. "Deep learning with limited numerical precision." arXiv:1502.02551 (2015).
- www.ini.uzh.ch/people/tobi; www.inilabs.com/products/dynamic-vision-sensors/overview
- Lazzaro et al, 1993; Mahowald, 1994; Deiss 1994; Boahen 2000; S. B. Erylimaz et al., IEDM 2015, T4.1 (2015).
- Merolla et al., *Science*, **345**(6197), 668 (2014).
- G. W. Burr et al., *IEDM Technical Digest*, 29.5, (2014); G. W. Burr et al., *IEEE Trans. Electr. Dev.,* 62(11), 3498-3507 (2015); G. W. Burr et al., *IEDM 2015,* T4.4 (2015).
- G. Bi and M. Poo, *J. Neuroscience*, 18(24), 10464 (1998)
- D. Kuzum *et al., Nano Lett., p. 2179 (2012)*
- S. Kim et al*., IEDM 2015*, 17.1, (2015).
- http://www.ini.uzh.ch/research/36620
- J. Hawkins, S. Blakeslee. ***On intelligence.*** Macmillan, 2007.
- www.zmescience.com/science/cognitive-biases-list/
- C. B. Frey and M. A. Osborne, "The Future of Employment: How susceptible are jobs to computerisation?" Sept. 2013

NVM-for-Machine Learning: References

gwburr@us.ibm.com

1. G. W. Burr, R. M. Shelby, et al., "Experimental demonstration and tolerancing of a large-scale neural network (165,000 synapses), using phase-change memory as the synaptic weight element," ***IEDM Technical Digest***, T29.5, December **2014**.
2. J.-W. Jang, S. Park, et al., "Optimization of Conductance Change in $Pr_{1-x}Ca_xMnO_3$-based} Synaptic Devices for Neuromorphic Systems," ***IEEE Electron Device Letters***, 36(5), 457-459 (**2015**).
3. G. W. Burr, R. M. Shelby, et al., "Experimental demonstration and tolerancing of a large-scale neural network (165,000 synapses), using phase-change memory as the synaptic weight element," ***IEEE Transactions on Electron Devices***, 62(11), 3498-3507 (**2015**).
4. G. W. Burr, P. Narayanan, et al., “Invited: Large-scale neural networks implemented with nonvolatile memory as the synaptic weight element: comparative performance analysis (accuracy, speed, and power),“ ***IEDM Technical Digest***, T4.4, December **2015**.
5. S. Sidler, I. Boybat, et al., “Large-scale neural networks implemented with non-volatile memory as the synaptic weight element: impact of conductance response," ***ESSDERC 2016***, Sept. **2016**.
6. A. Fumarola, P. Narayanan, et al., “Accelerating Machine Learning with Non-Volatile Memory: exploring device and circuit tradeoffs," ***2016 Internat. Conf. on Rebooting Computing***, Oct. **2016**.
7. G. W. Burr, R. M. Shelby et al., “Neuromorphic computing using non-volatile memory," ***Advances in Physics X***, **2**(1), 89-124 (**2017**).
8. P. Narayanan, A. Fumarola, et al., “Towards on-chip acceleration of the backpropagation algorithm using non-volatile memory,” ***IBM Journal of Research and Development***, to appear (**2017**).

Thank you for your attention