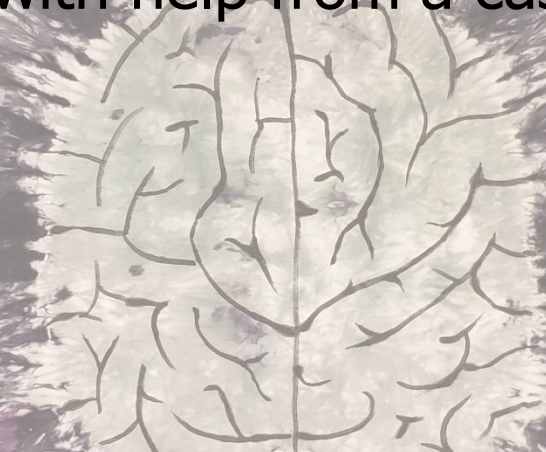
Using magnetic tunnel junctions to compute like the brain

Mark Stiles with help from a cast of dozens



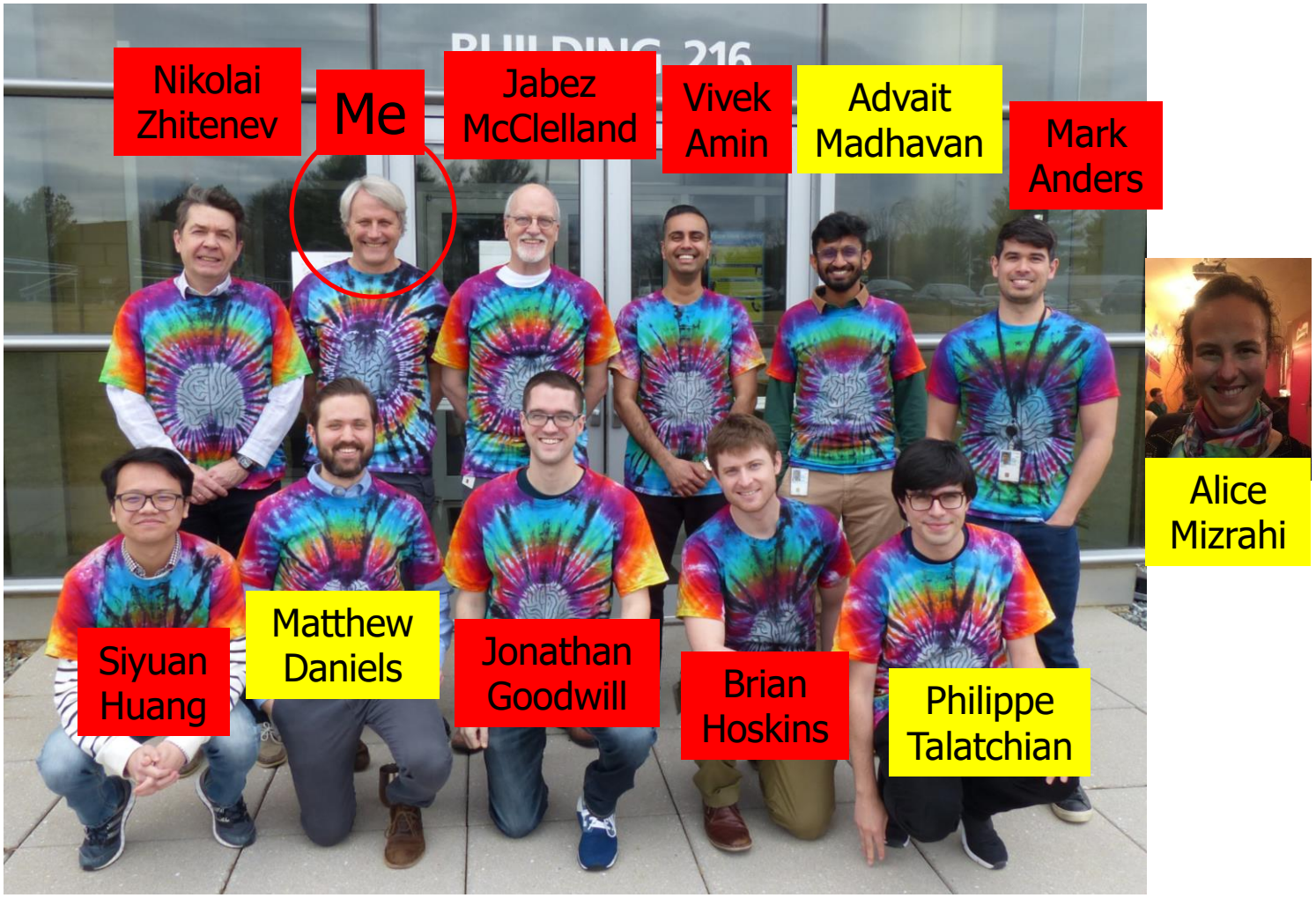
Why?

To understand how the brain works

To develop machine consciousness

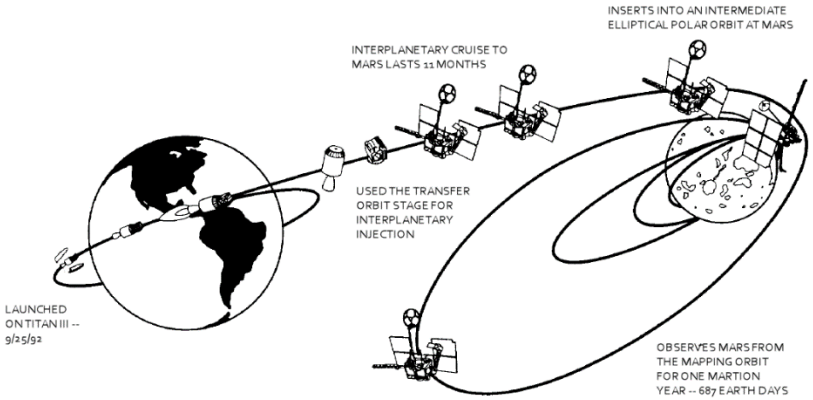
To do cognitive computing more efficiently

Hardware for Artificial Intelligence – NIST Gaithersburg



Computers are designed to solve numerical problems, the brain excels at categorical problems

Image: wikimedia.org



Satellite trajectories

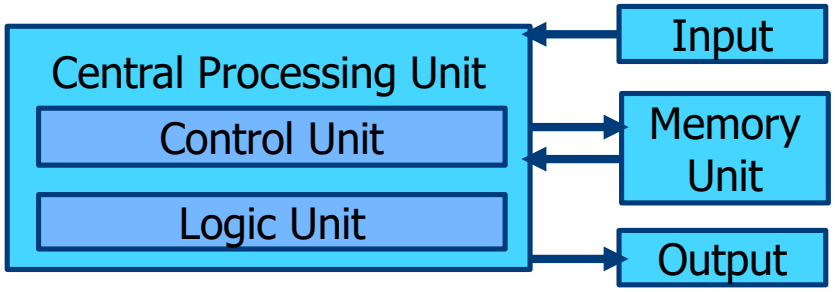
Image: cmu.edu

Natural language processing

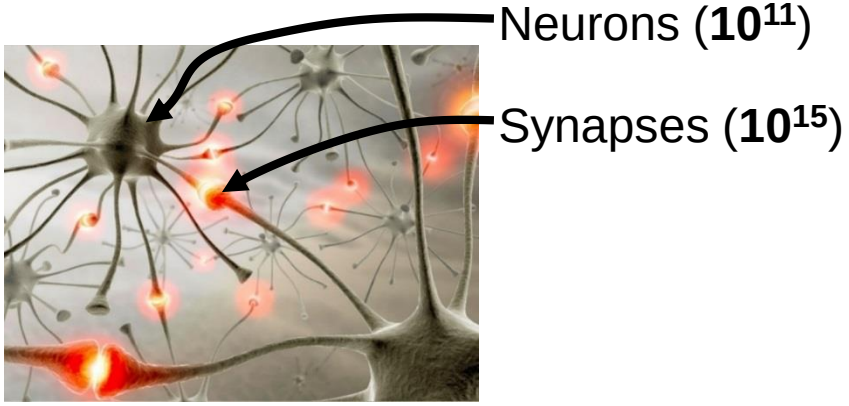
Digital Computer	**Neuromorphic System**
High precision numerical	Categorical

Computers separate long term memory from processing, while in the brain, they are intimately connected

von Neumann architecture:



Small scale structure of the brain:



Digital Computer	**Neuromorphic System**
Memory-Processor Bottleneck	Collocation of processor and memory
High precision numerical	Categorical

Computers use a rigid encoding scheme, the brain prioritizes resiliency with more flexible schemes

Binary Representation

64	32	16	8	4	2	1
1	0	0	1	1	0	1

Clocked processing

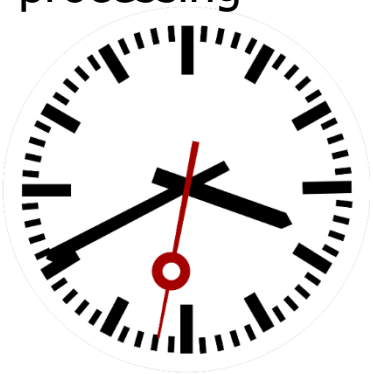


Image: Wikipedia

*Rossant et al., Frontiers in Neuroscience **5**, 9 (2011)*

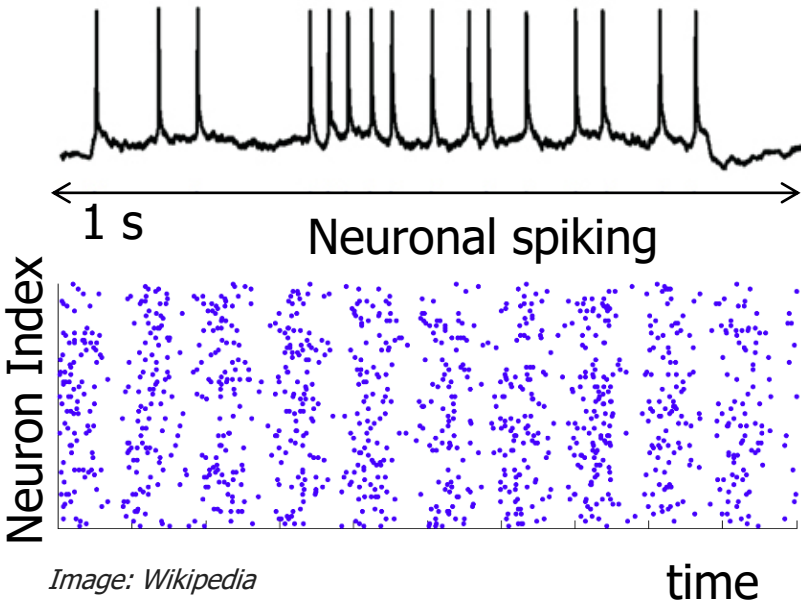
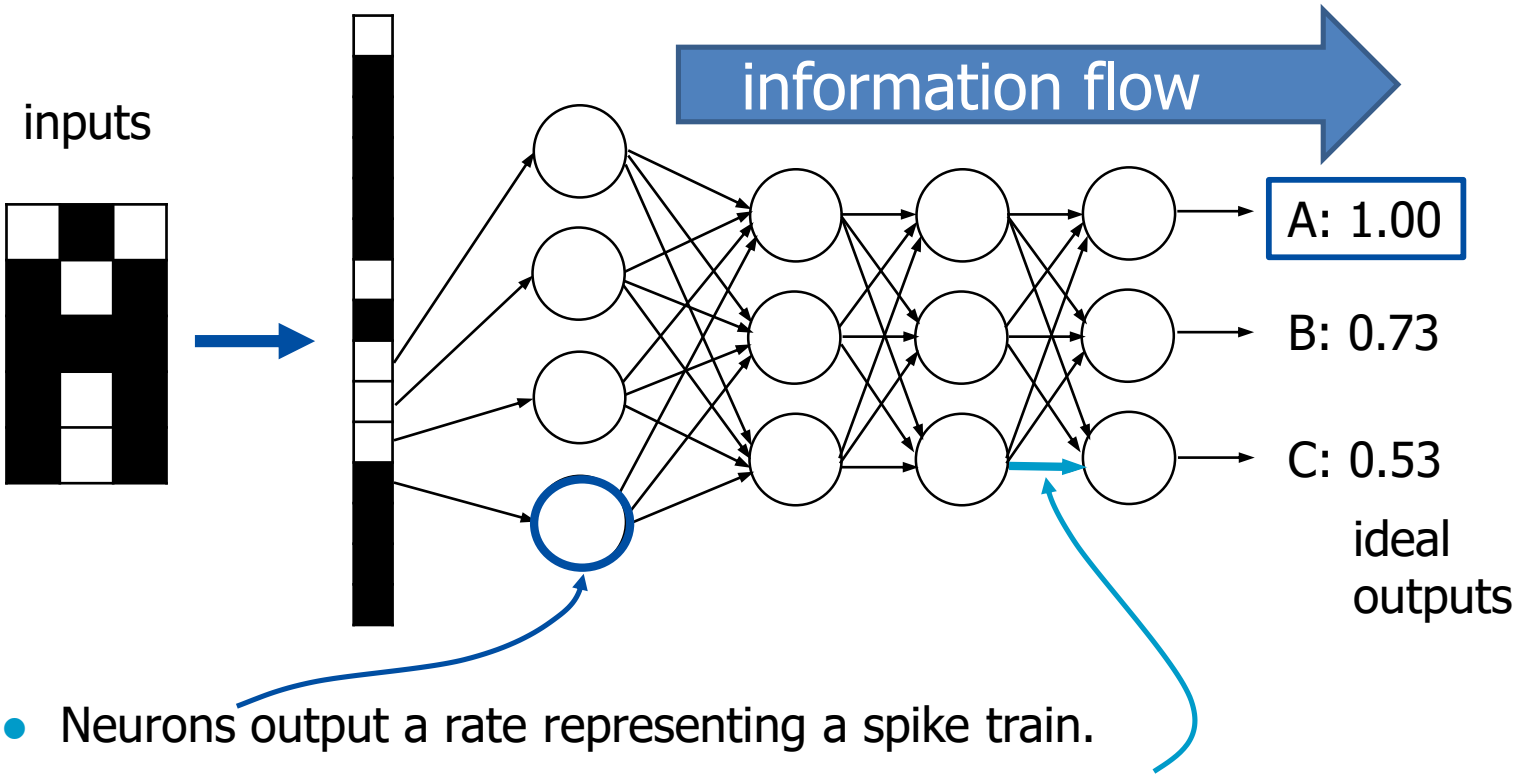


Image: Wikipedia

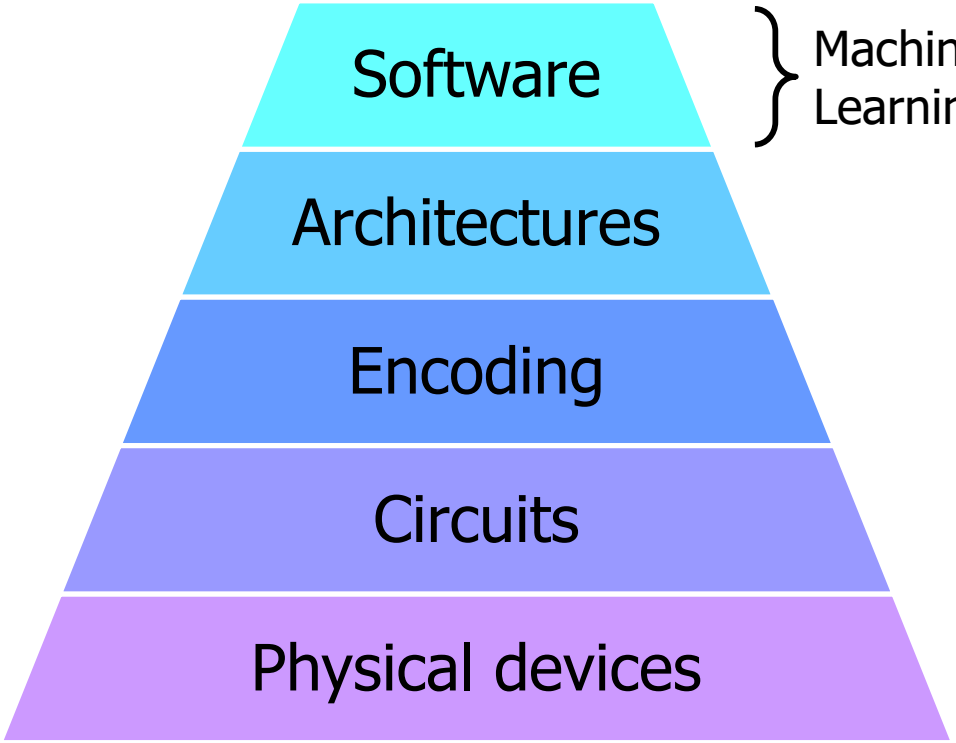
Digital Computer	**Neuromorphic System**
Discrete, Deterministic, & Synchronous	Analog, Stochastic, & Asynchronous
Memory-Processor Bottleneck	Collocation of processor and memory
High precision numerical	Categorical

Neural networks provide the simplest implementation of brain-like cognitive computing (rate coding)



- Neurons output a rate representing a spike train.
- The rate from each neuron is multiplied by synaptic weights for each neuron pair.
- Neurons sum the incoming weights and output a non-linear function of the sum (activation function).
- Network non-linearly transforms space to bring similar inputs together

A whole range of approaches to more efficient computing



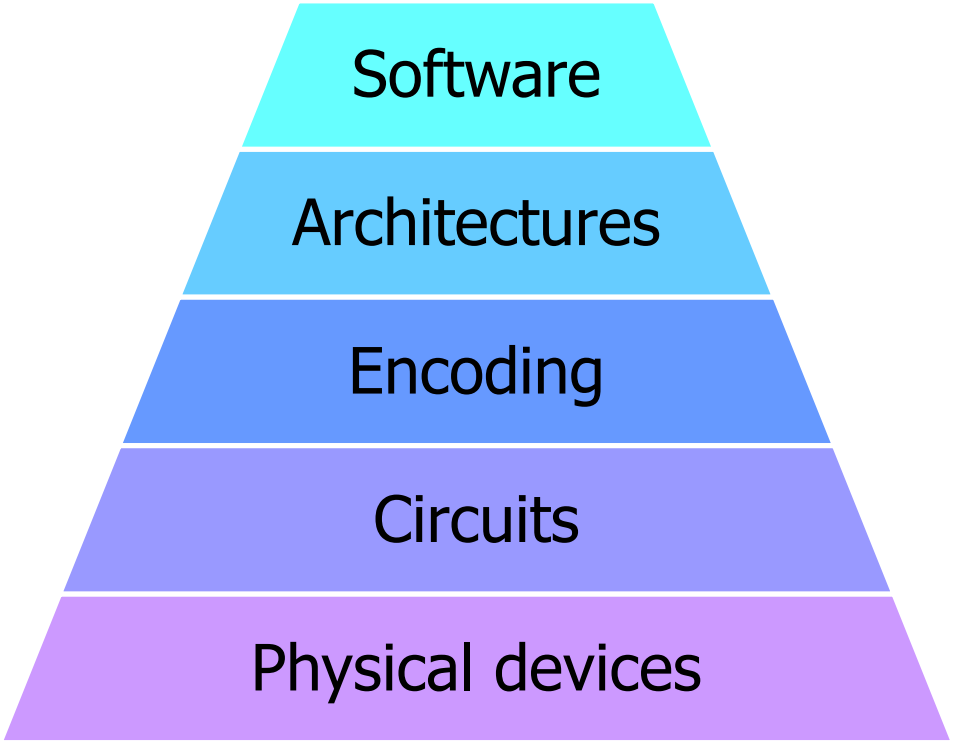
Machine Learning

Specialized Hardware, GPU, TPU

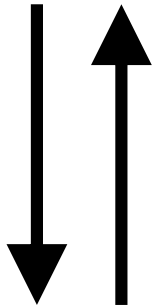
Specialty Chips, TrueNorth, Loihi, BrainScales

Hardware for Artificial Intelligence

CMOS has developed well-defined abstraction layers, but with novel devices, designing across the stack is necessary



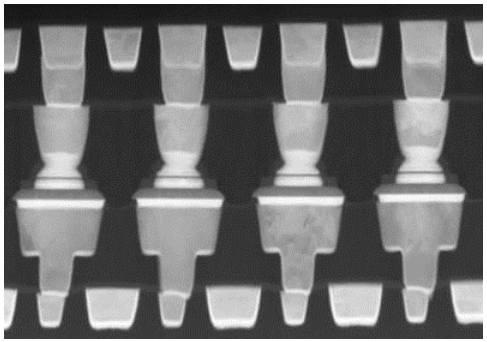
Details depend on what the device can do.



What is important depends on what the architecture needs

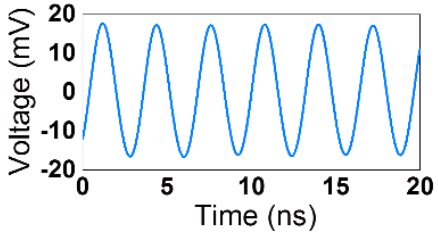
Magnetic tunnel junctions are multifunctional devices

- Non-volatile embedded memory

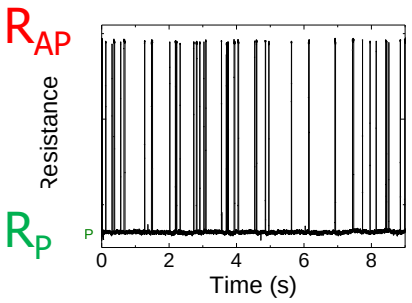


Intel: MRAM integrated into 22nm FinFET CMOS

- Spin-torque nano-oscillators



- Superparamagnetic tunnel junctions



Neuronal spiking

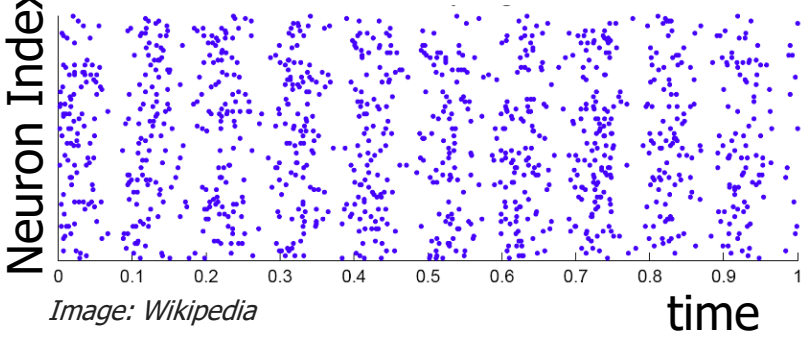
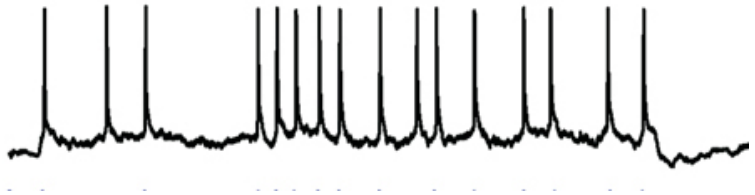
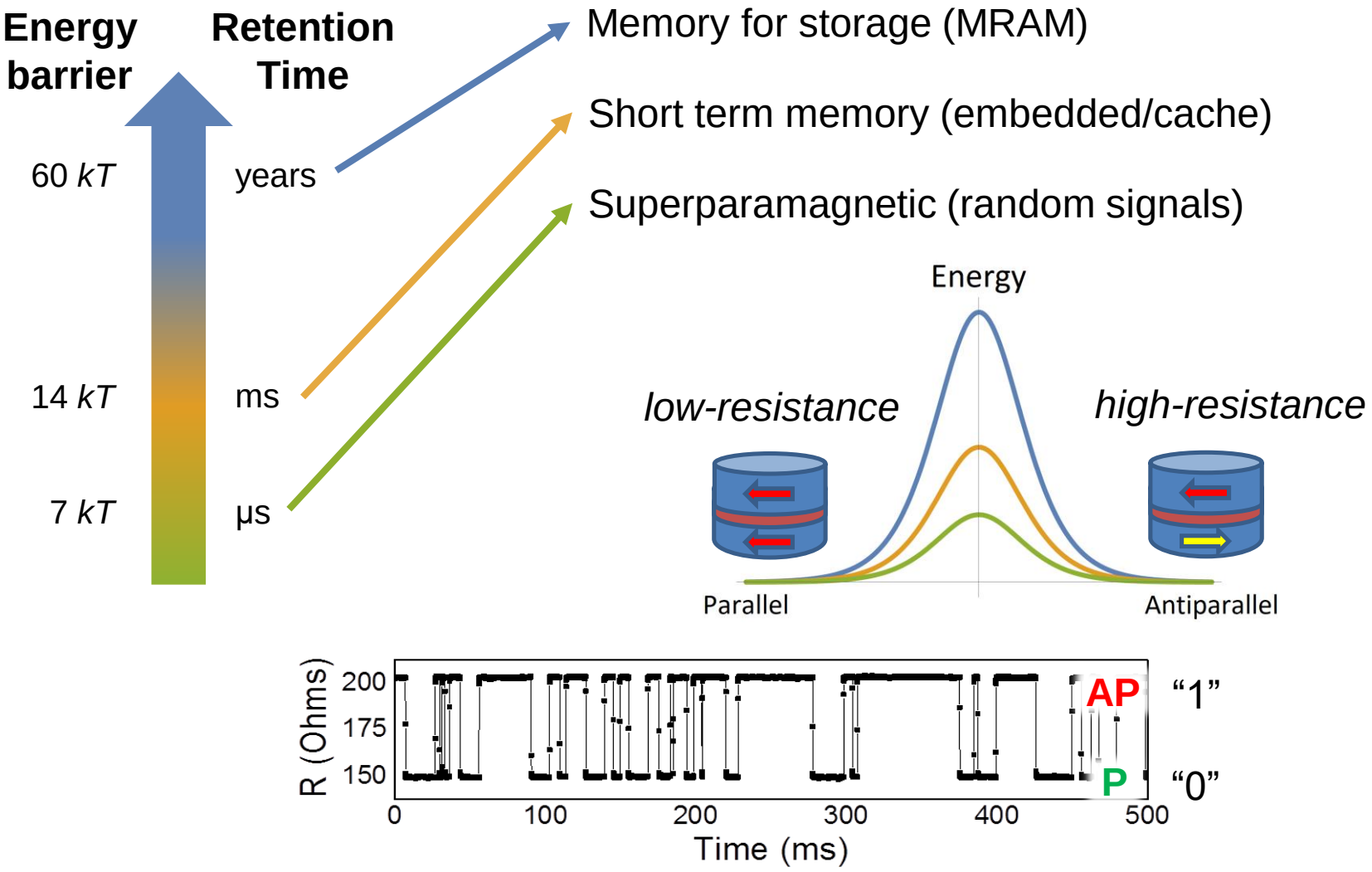


Image: Wikipedia



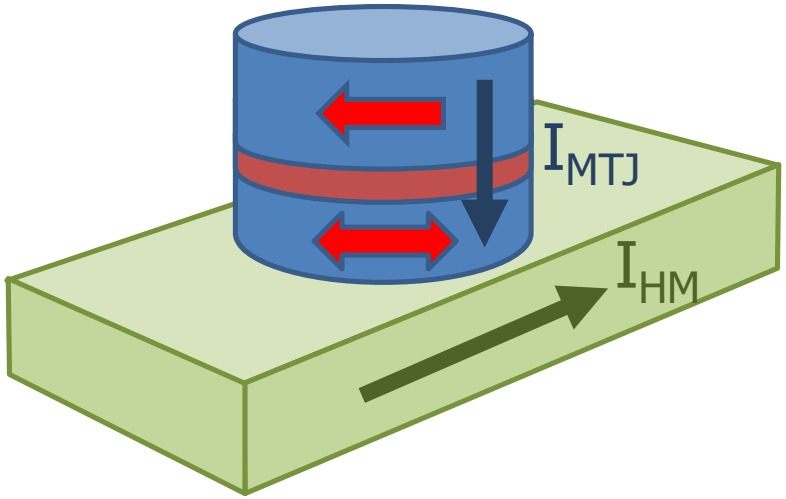
*Rossant et al., Frontiers in Neuroscience **5**, 9 (2011)*

Bigger energy barriers mean longer data retention, but more energy needed to write.



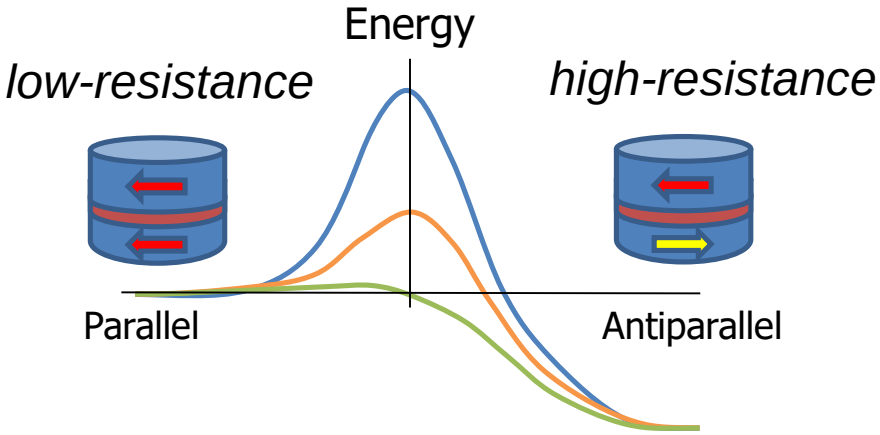
Magnetic tunnel junctions can be controlled by current.

Electron flow through tunnel junction
→ Spin-transfer torque

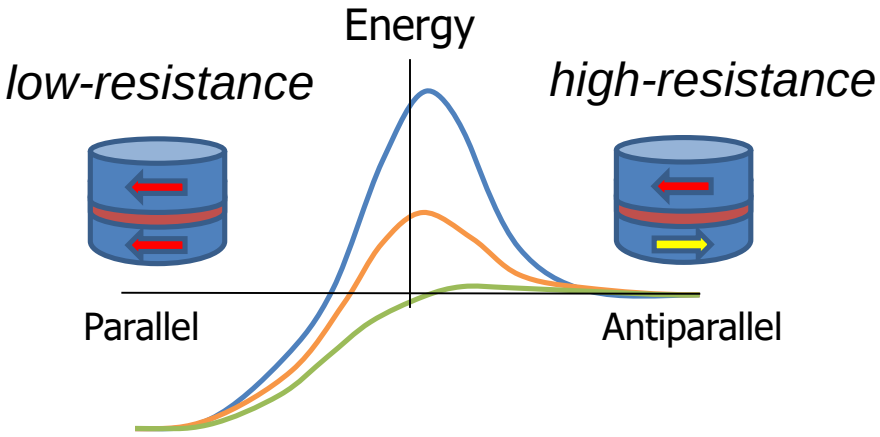


Electron flow through heavy metal underlayer
→ Spin-orbit torque

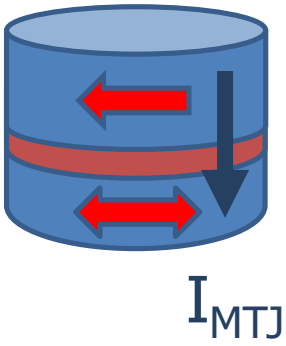
Positive current

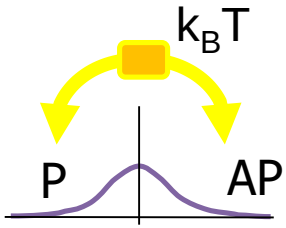


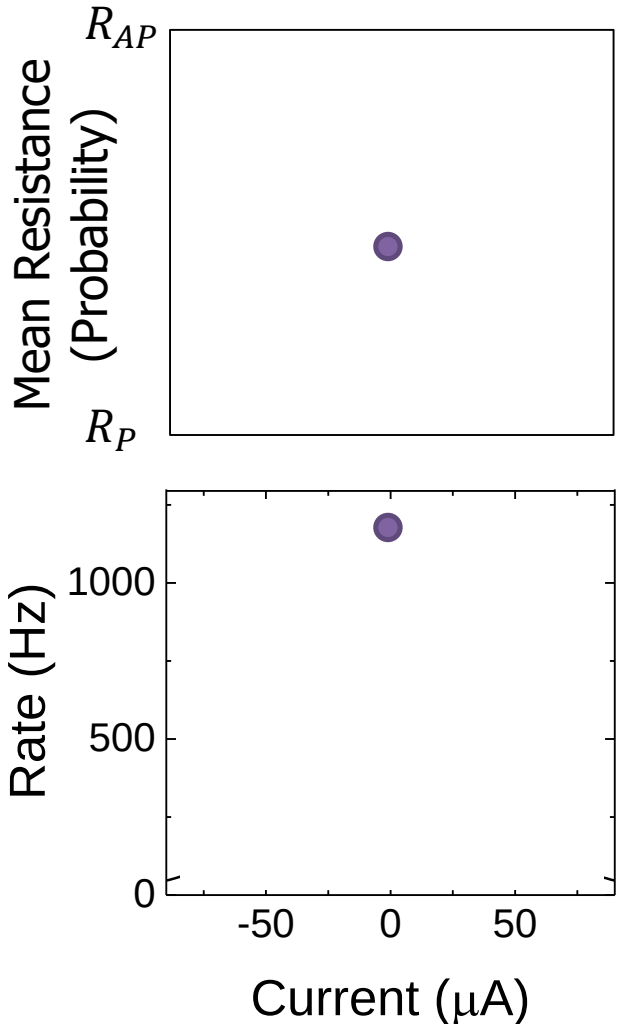
Negative current

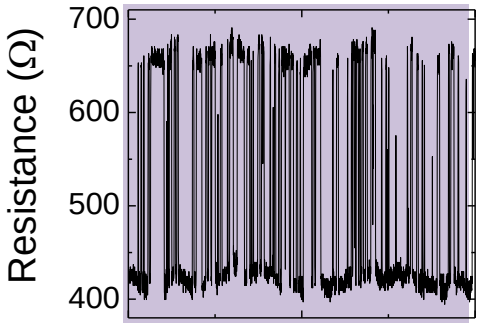


Superparamagnetic magnetic tunnel junctions: Control rate with spin-transfer torque (or spin-orbit torque)



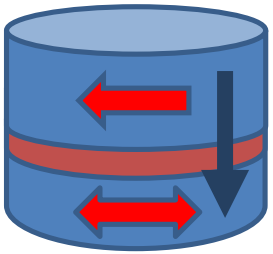






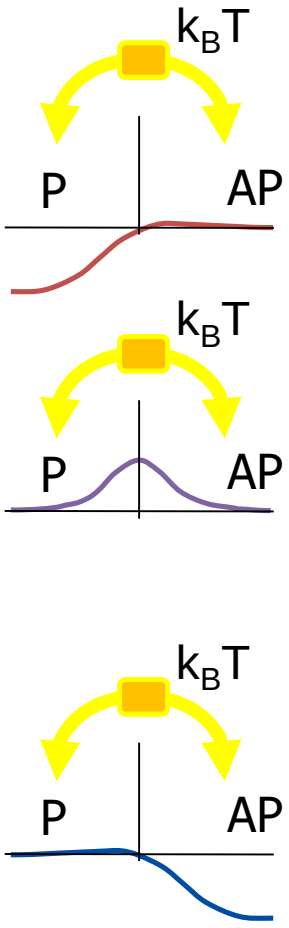
Superparamagnetic magnetic tunnel junctions: Control rate with spin-transfer torque (or spin-orbit torque)

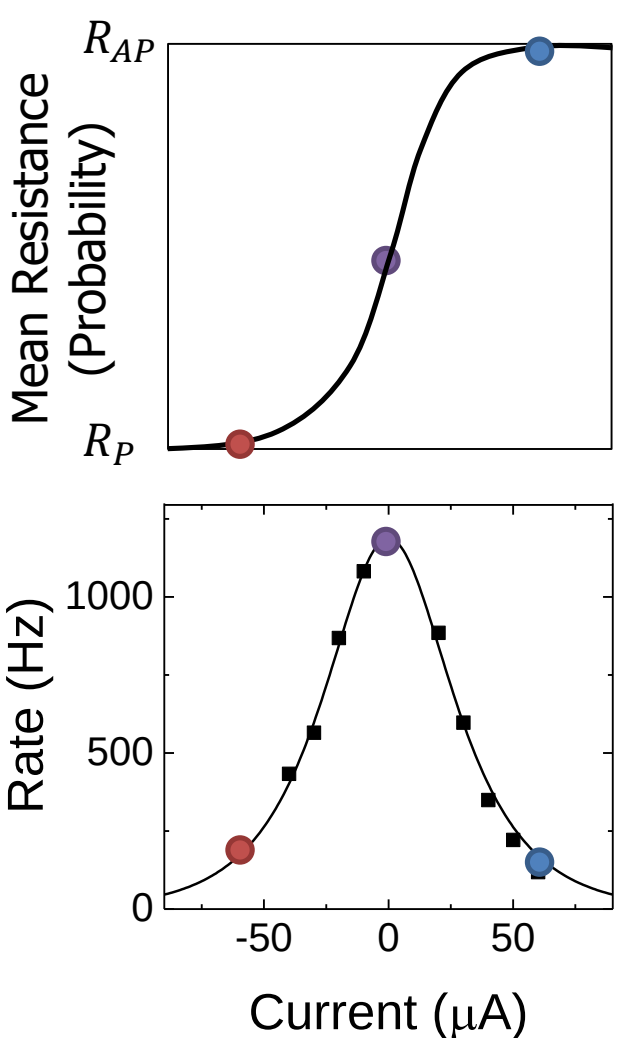
I < 0
Stabilizes parallel state

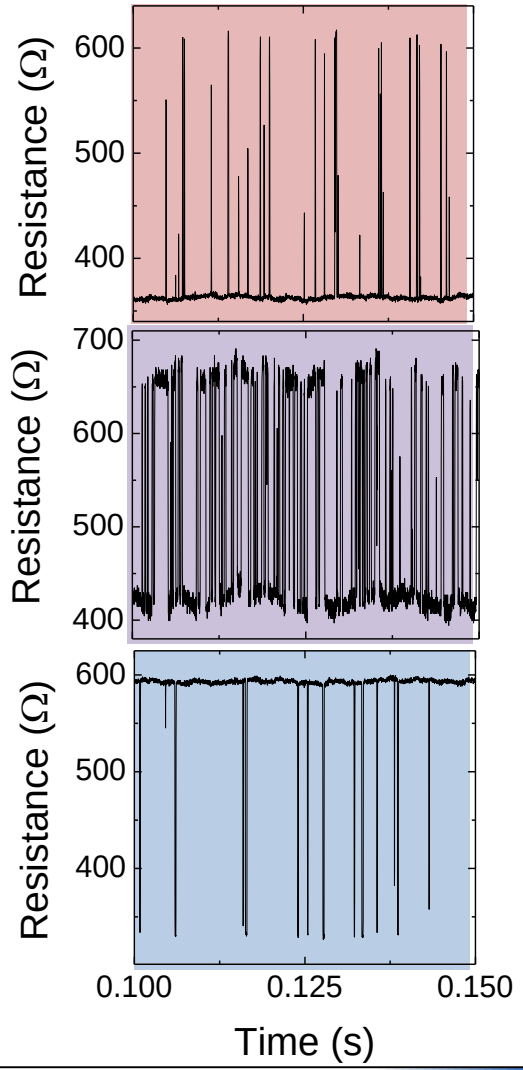


I_{MTJ}

I > 0
Stabilizes antiparallel state



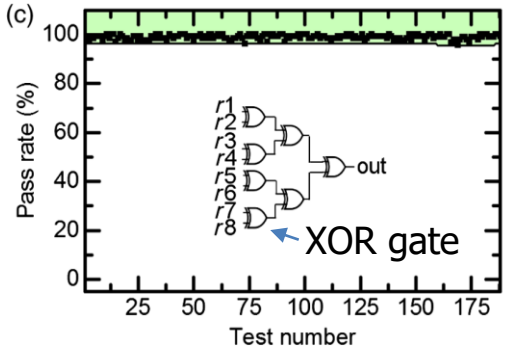




Computing with superparamagnetic tunnel junctions

- Random number generation

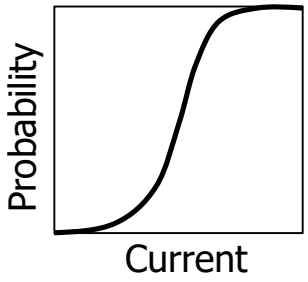
$$p = 0.5$$



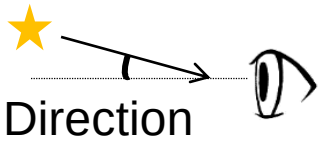
Won Ho Choi et al., 2014 IEEE IEDM pp. 12.5.1-12.5.4,

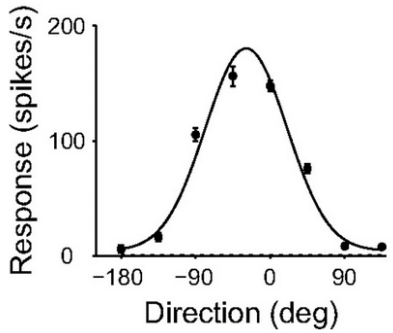
D. Vodenicarevic et al., Phys. Rev. Applied 8, 054045 (2017)

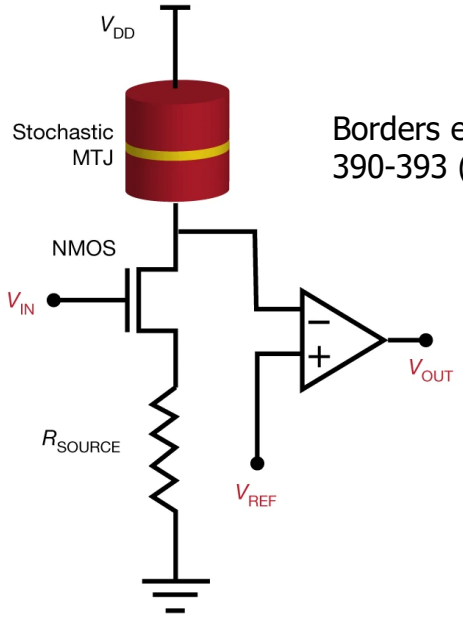
- Probabilistic computing



p -bit

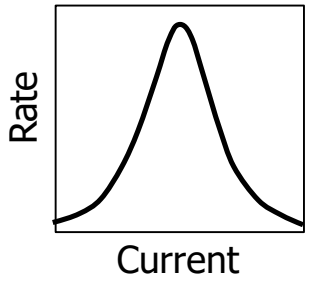






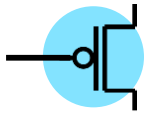
Borders et al., Nature 573, 390-393 (2019)

- Rate coding

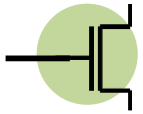


Many reasons to incorporate Complementary Metal Oxide Semiconductor (CMOS)

pmos

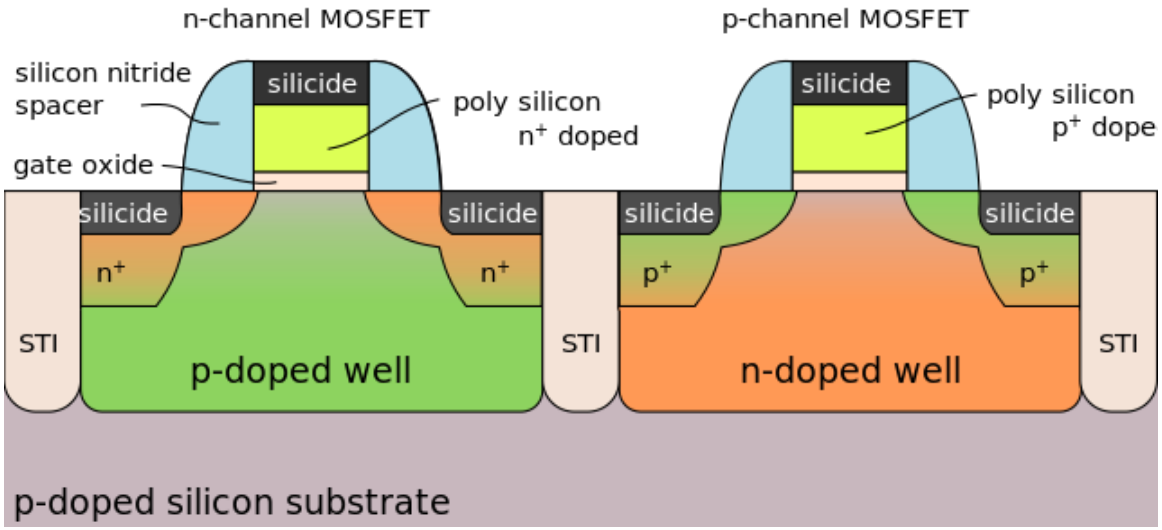
V_g 

On for $V_g = 0$
Off for $V_g = V_{dd}$

V_g 

Off for $V_g = 0$
On for $V_g = V_{dd}$

nmos



Author: Cephidan, https://commons.wikimedia.org/wiki/File:LDD-MOS_transistor_-_CMOS_with_STI.svg

Many reasons to incorporate Complementary Metal Oxide Semiconductor (CMOS)

pmos

V_g

On for $V_g = 0$
Off for $V_g = V_{dd}$

V_g

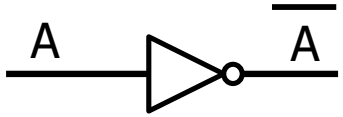
Off for $V_g = 0$
On for $V_g = V_{dd}$

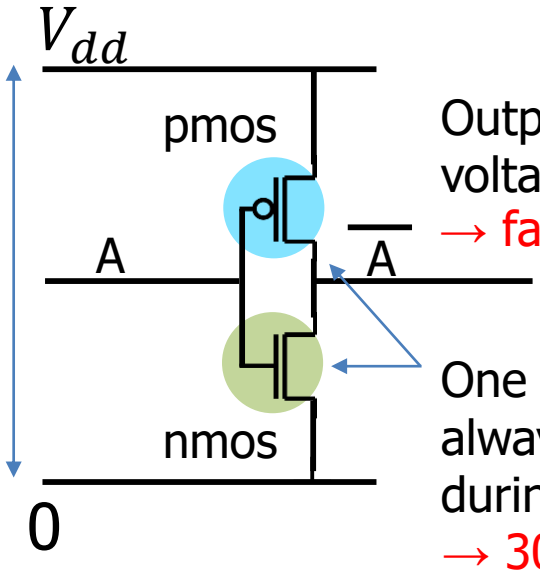
nmos

Voltage pins
→ local energy source

What can other devices do better?
- Non-volatility (local memory)
- Plasticity (local learning)
- Stochasticity
- Oscillators

Inverter





In 2014, 250×10^{18} transistors manufactured
→ cheap substrate

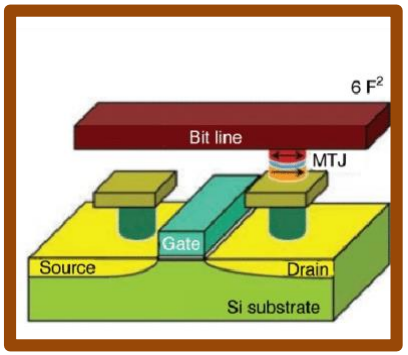
Output voltage from voltage pins, not input
→ fanout

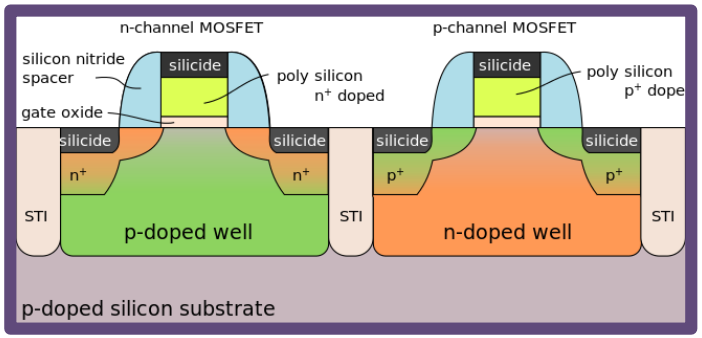
One transistor always off except during switching,
→ 30 aJ and up

Energies vary over many orders of magnitude

1 nJ
1 GeV
Neural spike

1 pJ
1 MeV
MRAM switching
Synaptic event





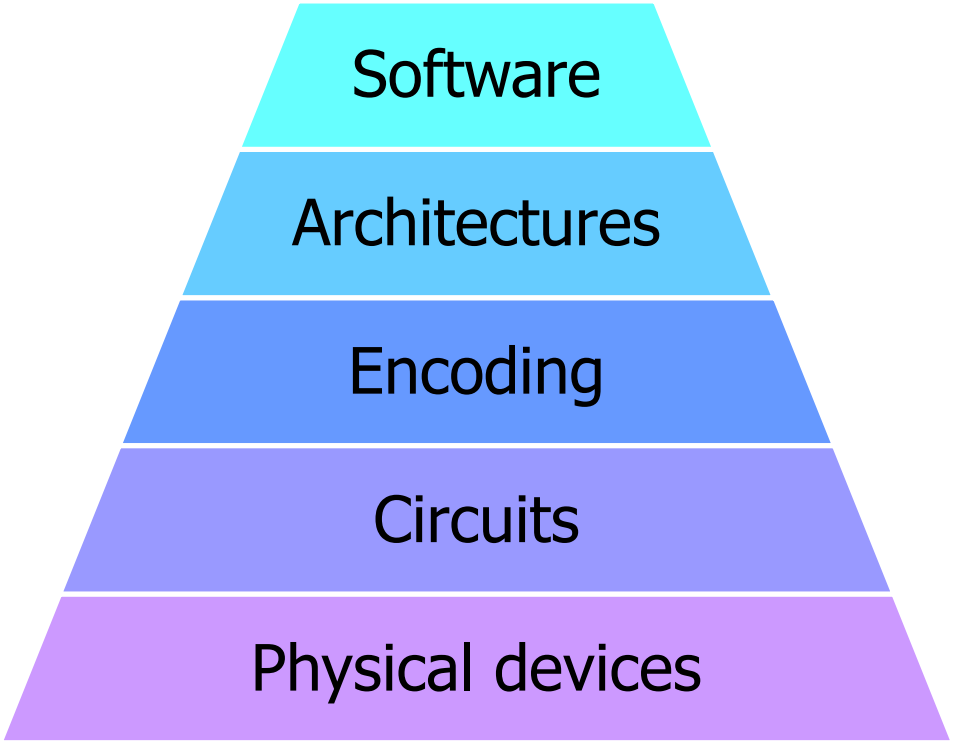
1 fJ
1 keV
Gate transition

1 aJ
1 eV
MRAM barrier

Ion channel
Room temperature

1 zJ
1 meV
Liquid Helium temperature

Population coding with superparamagnetic tunnel junctions

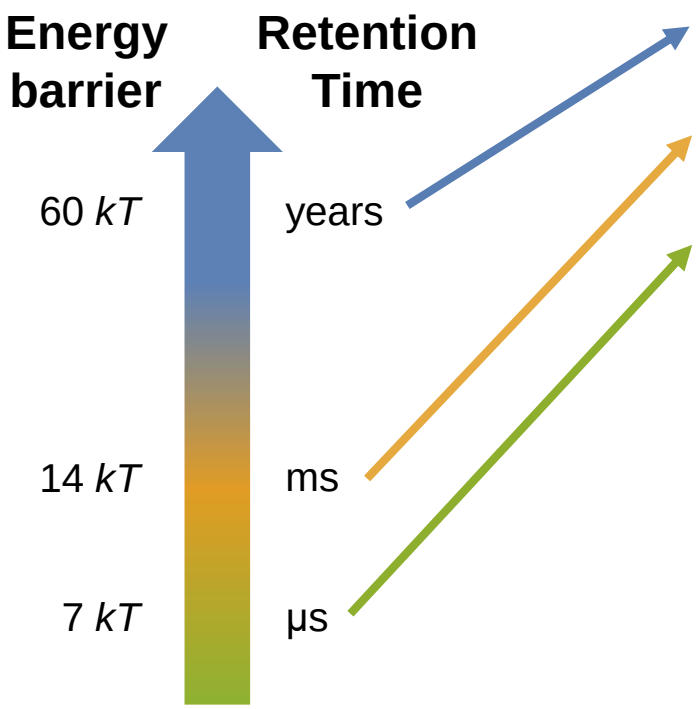


- Rate/Population Coding
- Transition rate, binary
- Pre-charge sense amplifier
- Superparamagnetic tunnel junctions, Short term memory

A. Mizrahi, T. Hirtzlin, A. Fukushima, H. Kubota, S. Yuasa, J. Grollier, and D. Querlioz, Nature Comm. 9, 1533 (2018);

A. Mizrahi, J. Grollier, D. Querlioz, and M. D. Stiles, J. Appl. Phys. 124, 152111 (2018).

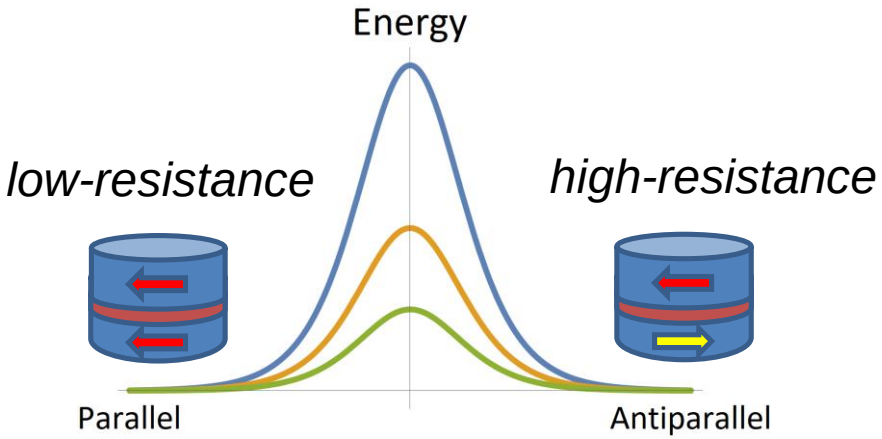
Bigger energy barriers mean longer data retention, but more energy needed to write.

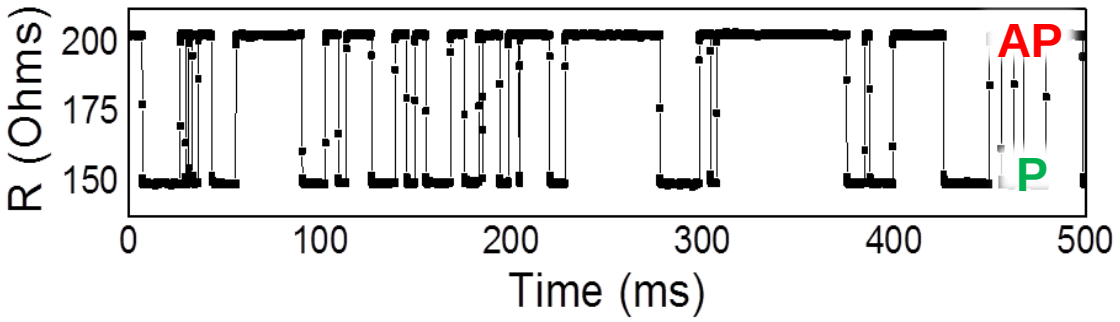


Memory for storage (MRAM)

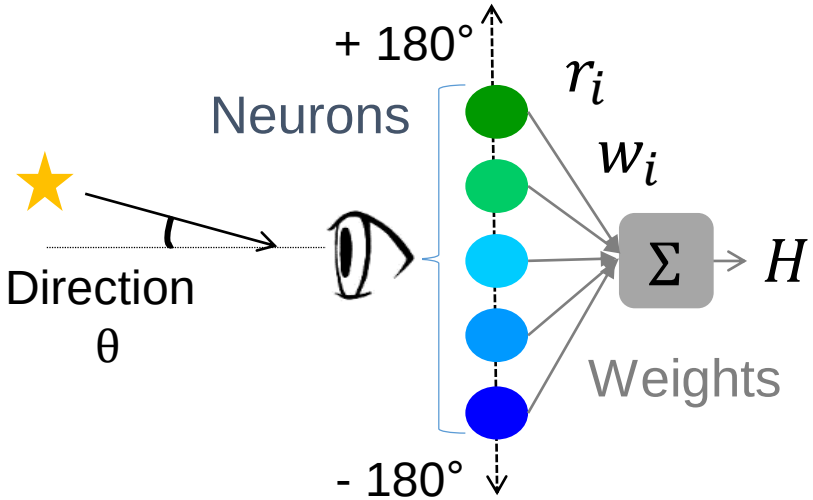
Short term memory (synaptic weights)

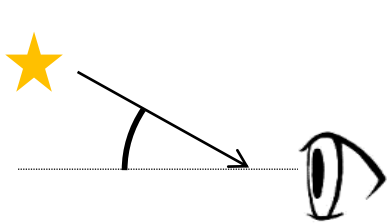
Superparamagnetic (spiking neurons)

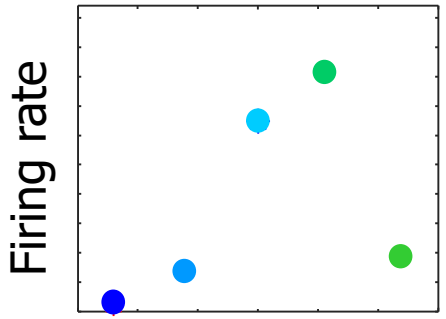


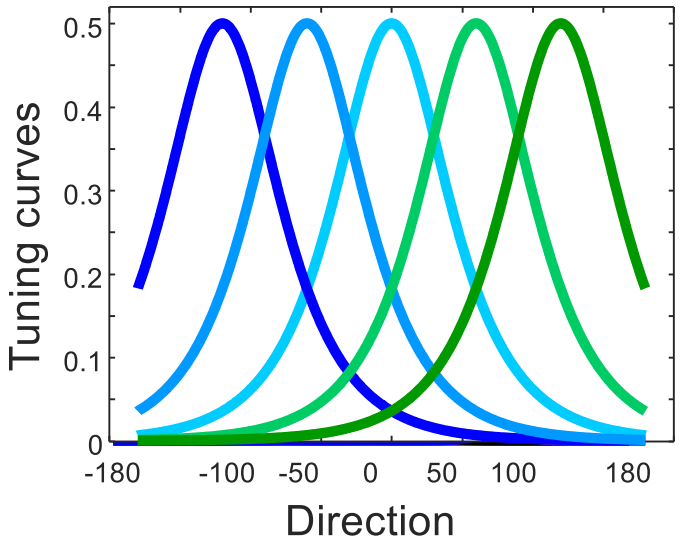


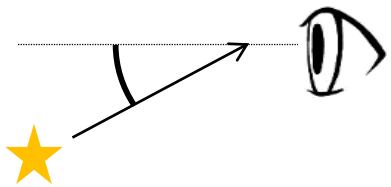
Population coding – represent continuous degrees of freedom with multiple spike rates

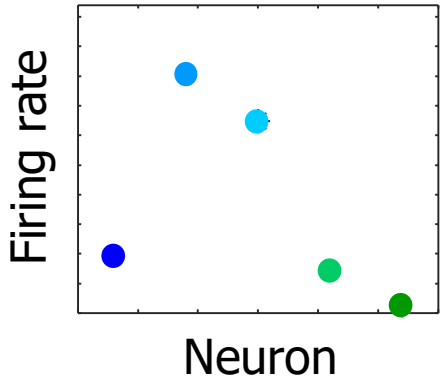






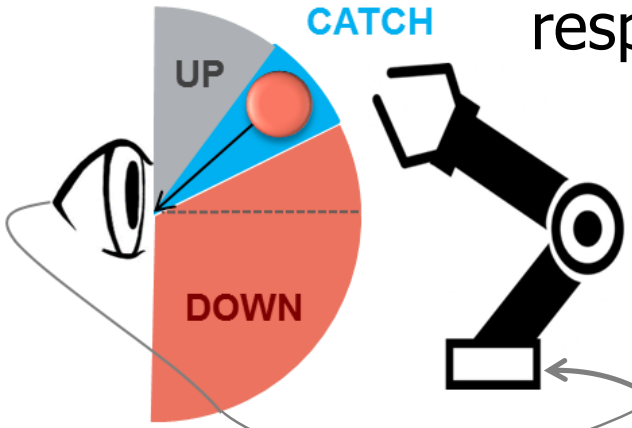




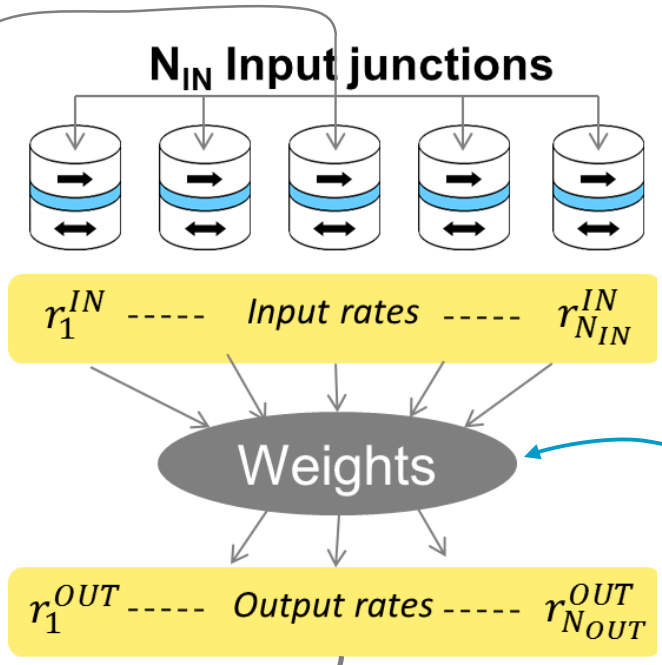


Simple sense, respond, measure, and learn test

Sensory input



Motor response



T. Hirtzlin at C2N:

	SMTJ-based	CMOS only
Area	12,000 µm²	20,000 µm²
Energy	7.8 nJ	20 nJ

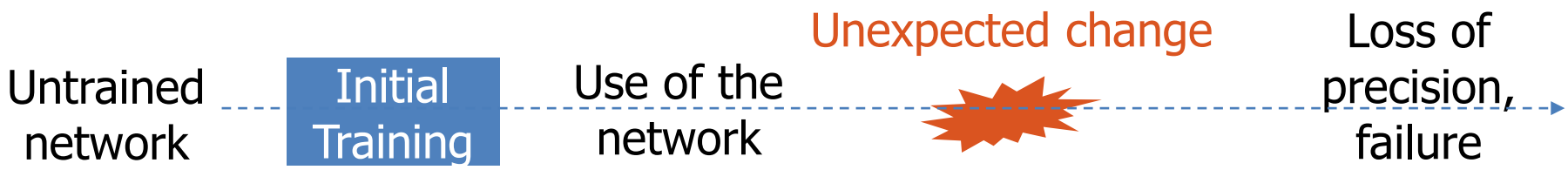
Weights stored in stable MTJs.
Can we save energy there?

Key advantage = stochastic analog to digital conversion

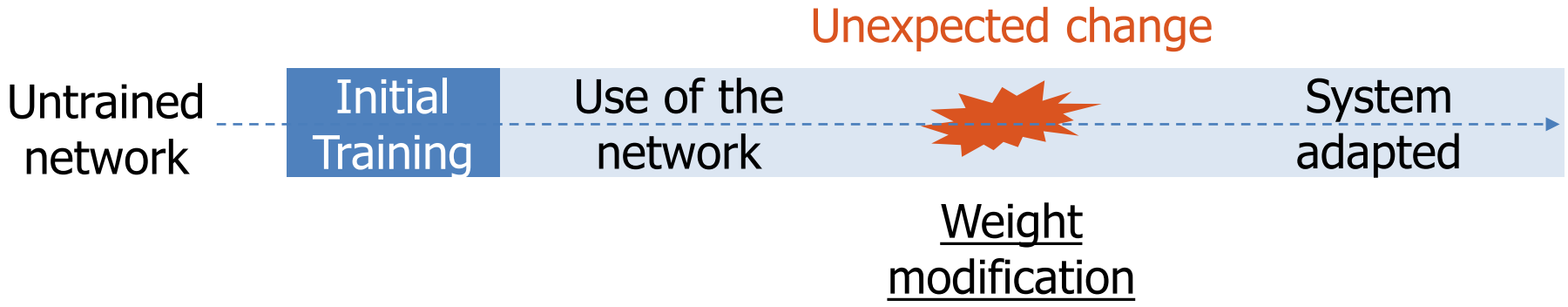
Continuous learning: the key to a robust system that can adapt to changes

Usually system is only trained once because it requires a lot of energy

BUT unable to adapt to changes!

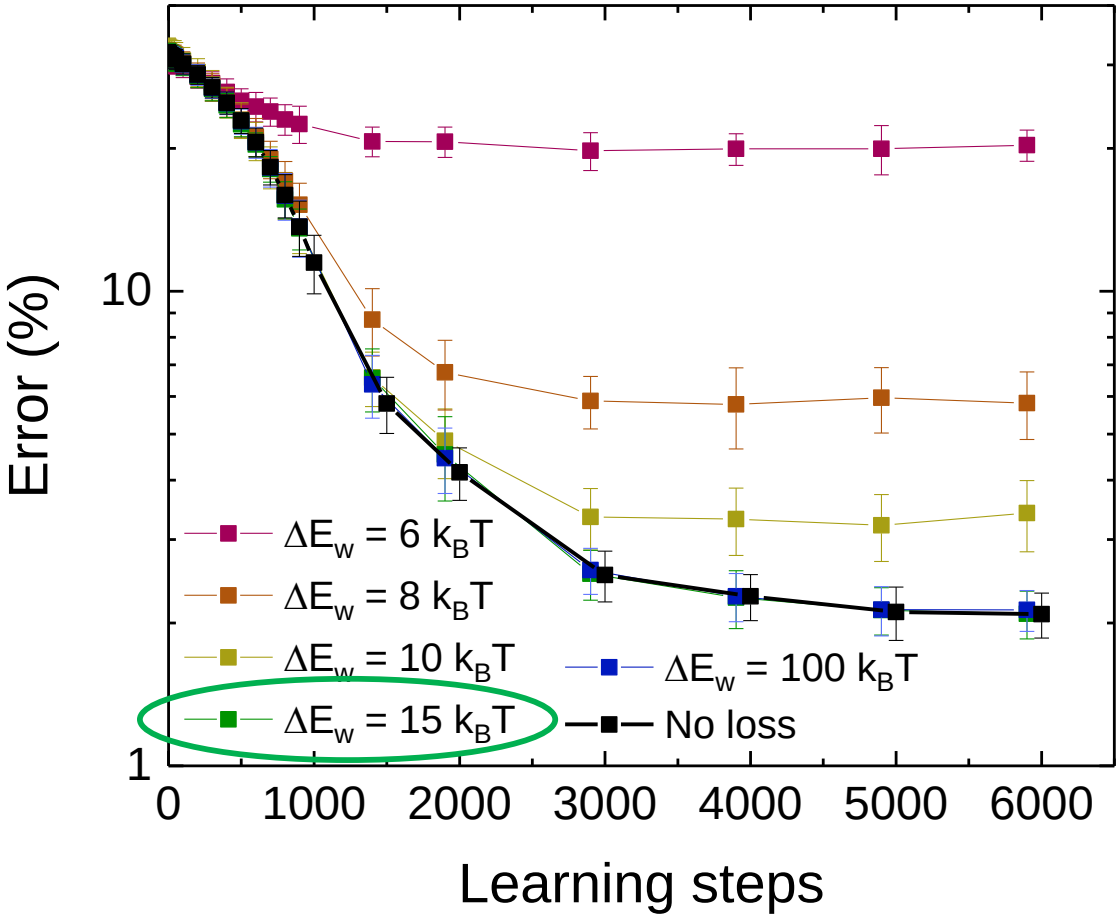


System equipped with continuous learning



Continuous learning overcomes unreliable synapses

Here, $\Delta E = 15\ k_B T$ magnetic tunnel junctions can be used for memory with no precision loss



Using unreliable synapses lowers the energy consumption

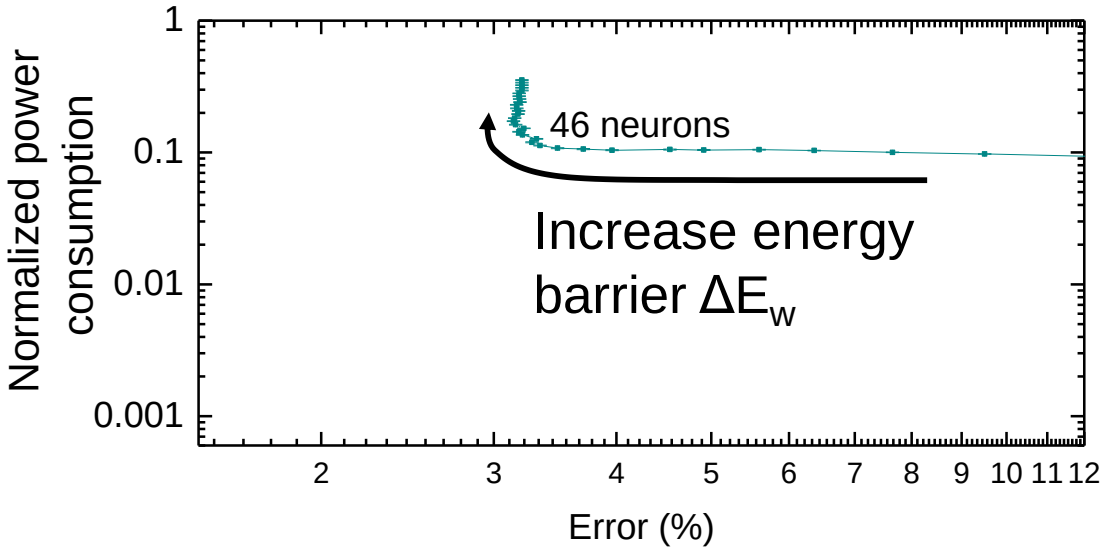
Write current

$$I \propto \Delta E_w$$

(Sato et al., APL 2014)

Write energy

$$Energy \propto \Delta E_w{}^2$$



Using unreliable synapses lowers the energy consumption

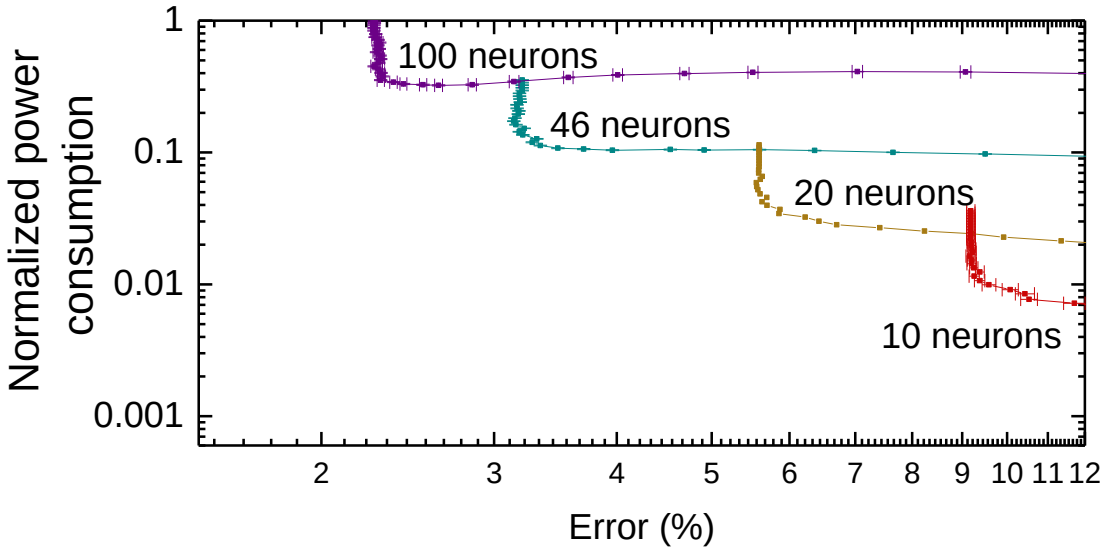
Write current

$$I \propto \Delta E_w$$

(Sato et al., APL 2014)

Write energy

$$Energy \propto \Delta E_w{}^2$$



Using unreliable synapses lowers the energy consumption

Write current

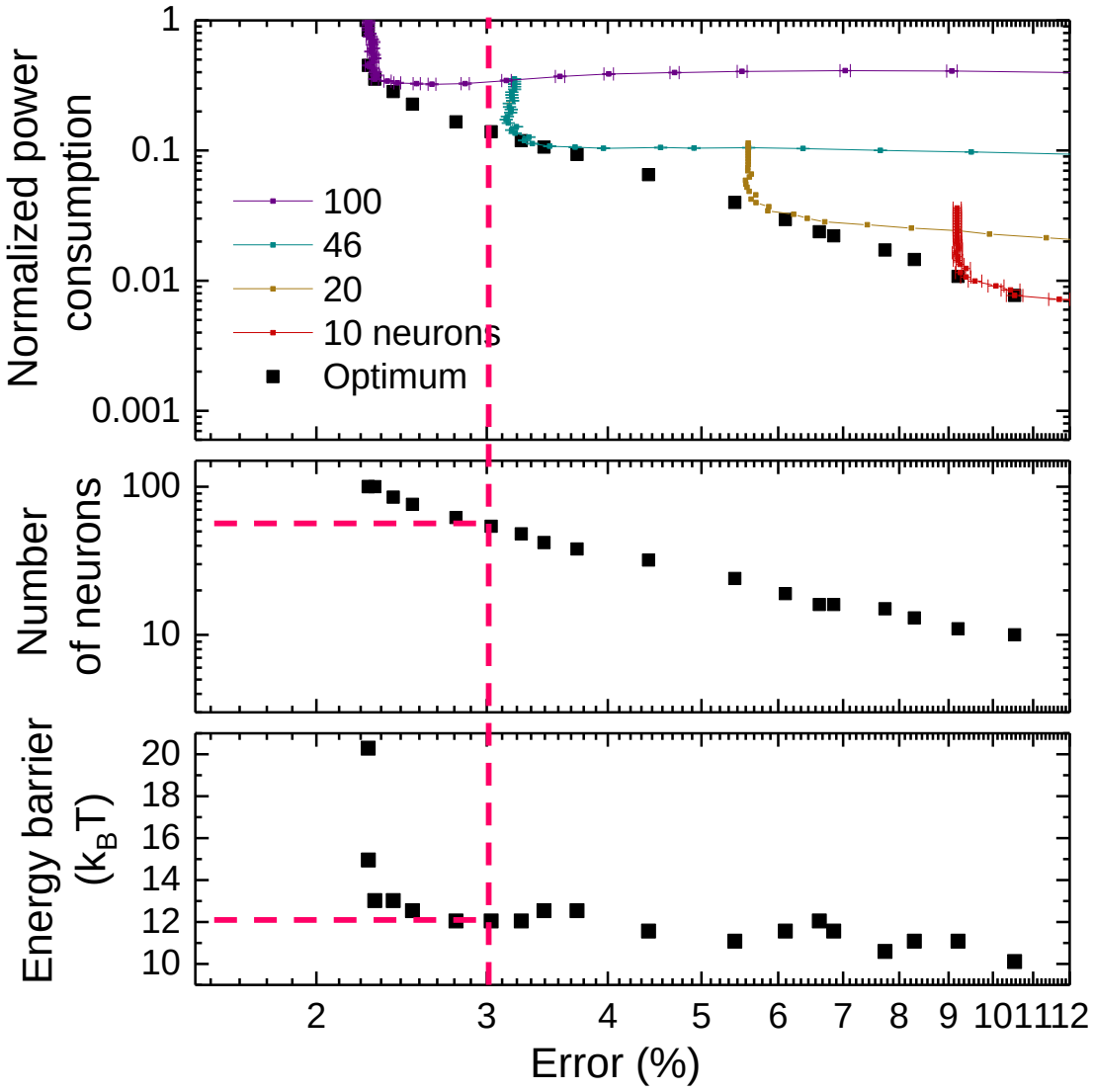
$$I \propto \Delta E_w$$

(Sato et al., APL 2014)

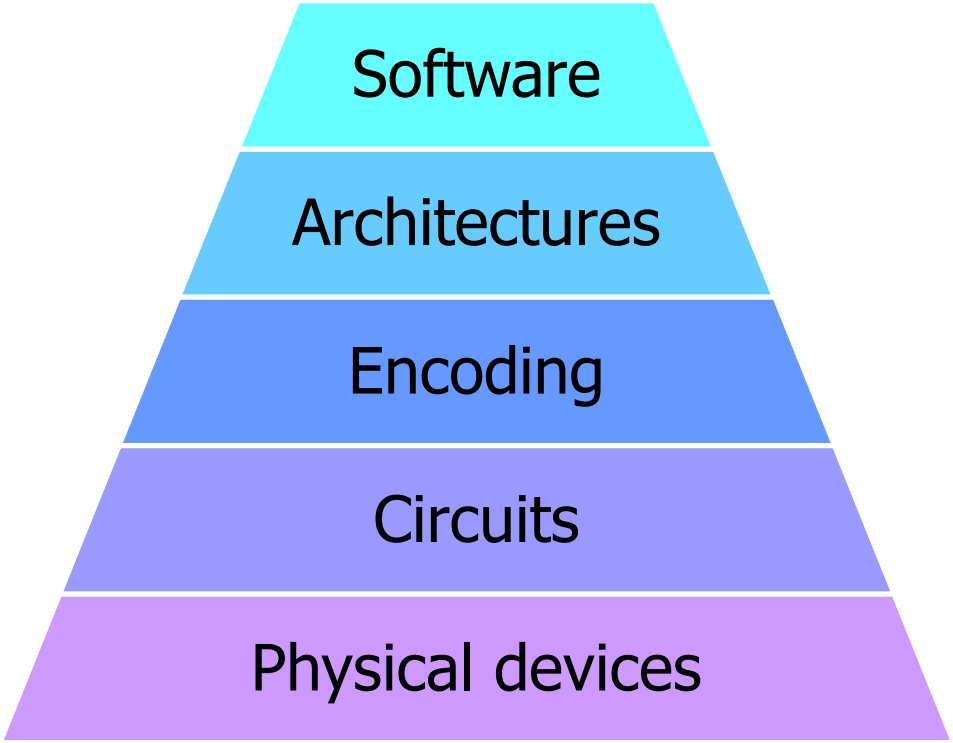
Write energy

$$Energy \propto \Delta E_w{}^2$$

Example:
For 3% precision
→ 54 neurons in each population (2916 weights)
→ ΔE_w = 12 $k_B T$



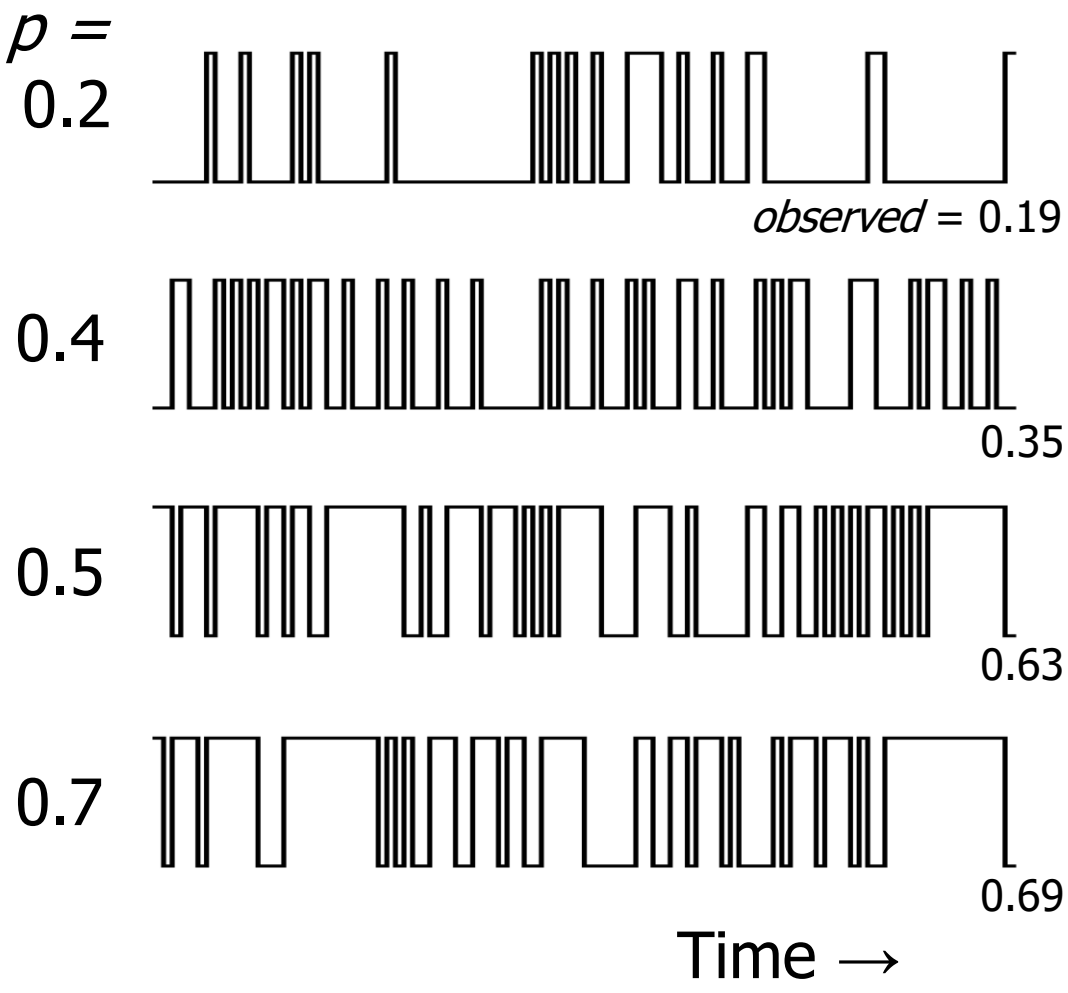
Stochastic computing with superparamagnetic tunnel junctions



- Neural network
- Stochastic bit streams
- AND and OR gates, PCSA
- Superparamagnetic tunnel junctions

M. W. Daniels, **A. Madhavan**, P. Tatatchian, A. Mizrahi, and M. D. Stiles, Physical Review Applied, 13, 034016 (2020).

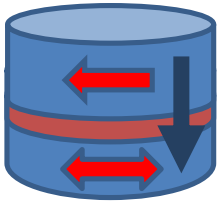
Stochastic computing represents numbers as probabilistic bit streams



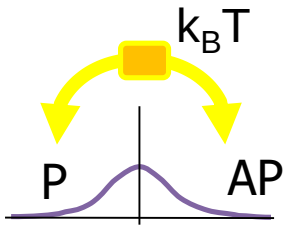
- Resilient to white noise
  - Incorrect bit ⟹ error $\approx O(1)$
  - Compare to binary: error $\approx O(2^N)$
  - Naturally coherent with bilevel devices (magnetic tunnel junctions).
- Interpreting these "spike trains" as **rates** or **probabilities** restricts us to $0 \leq p \leq 1$.

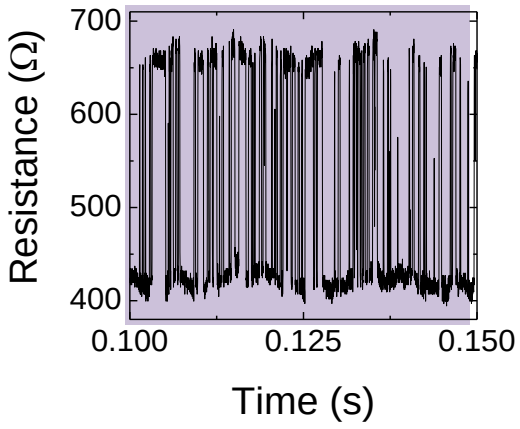
Unbiased superparamagnetic tunnel junction plus CMOS for low energy random bitstreams

Superparamagnetic tunnel junction

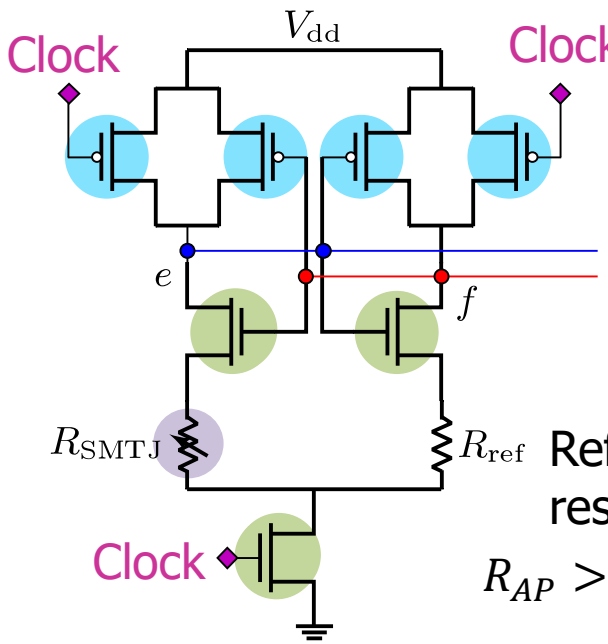


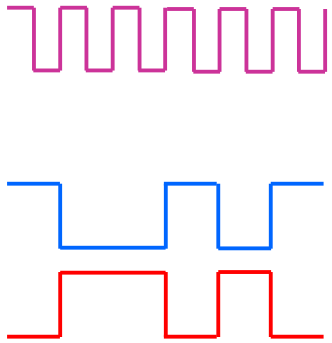
$I_{MTJ} << I_C$





Pre-Charge Sense Amplifier (PCSA)





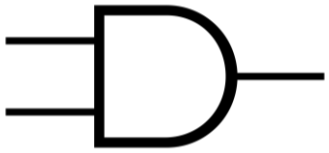
Reference resistor

$R_{AP} > R_{ref} > R_P$

Low energy random bits

$p = 0.5$

AND gates for multiplication
– Programmable bitstream generator: 10 fJ per bit



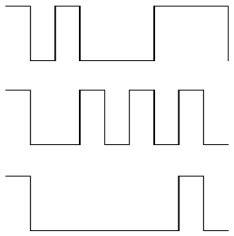
AND gate (multiply)

Example

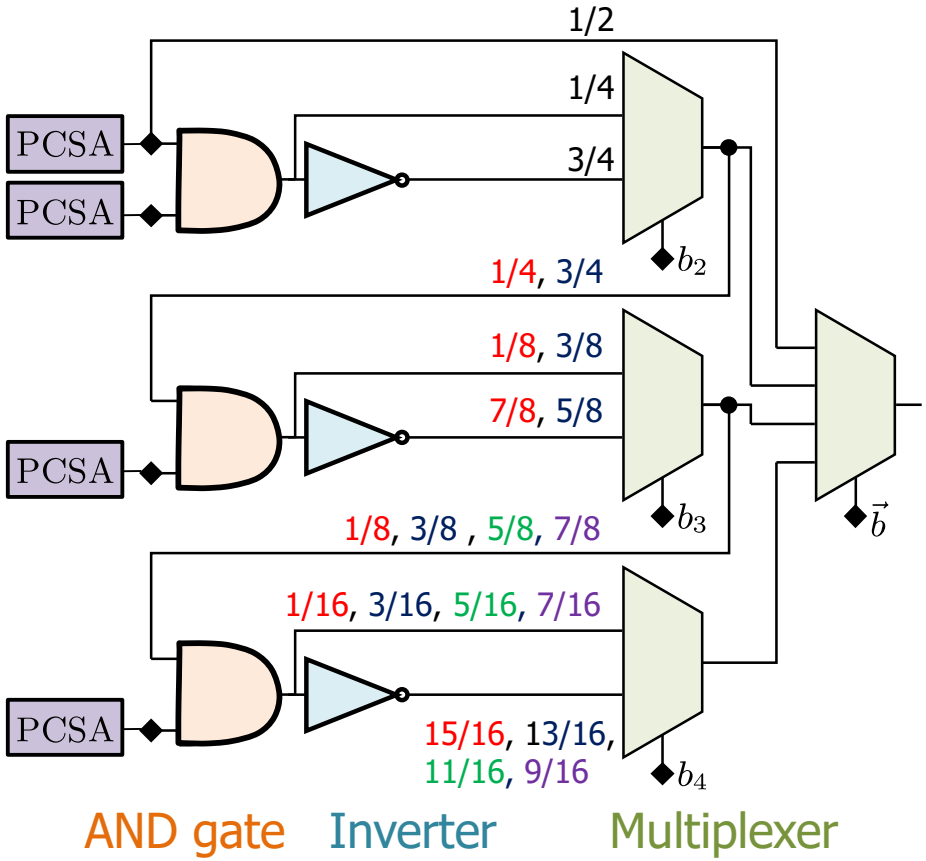
1001010100 (0.4)

x 1010001110 (0.5)

= 1000000100 (0.2)

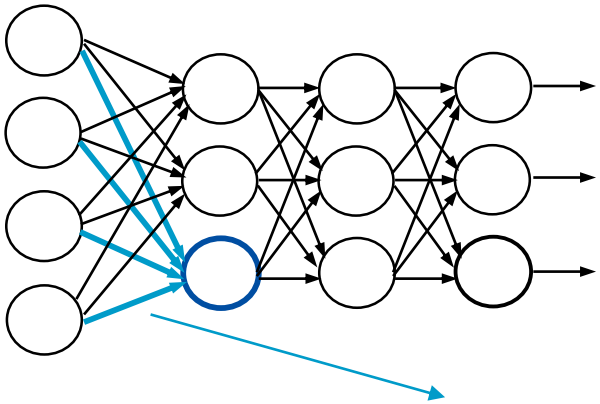


$$p_{\mathrm{AND}} = p_a p_b$$



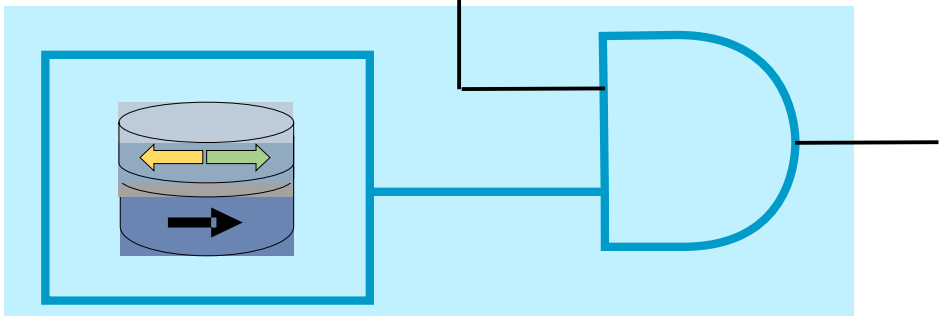
Typical stochastic computing uses Linear Feedback Shift Registers
→ Short period pseudorandom numbers

Synapses from bitstream generators (weights) and AND gates (multiplication)



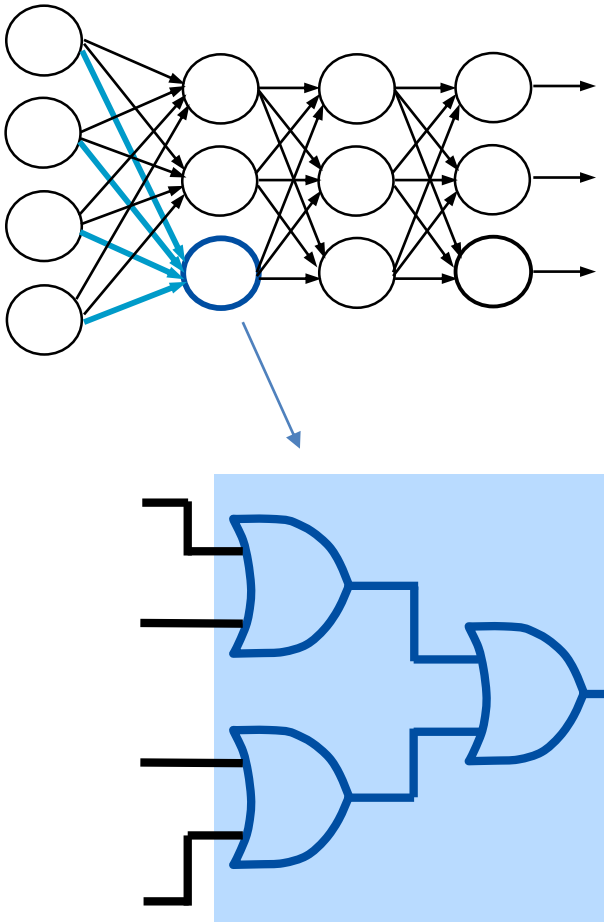
AND-gate synapse
Synaptic weight:

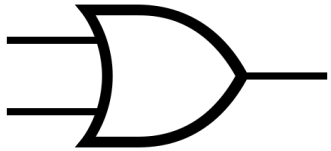
Bit stream
from neuron



Programmable Bit stream generator

Neurons from OR gates for nonlinear summation





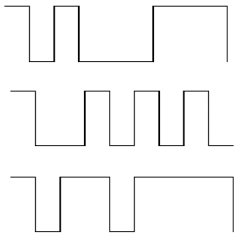
OR gate (nonlinear add)

Example

1001010100 (0.4)

+ 1010001110 (0.5)

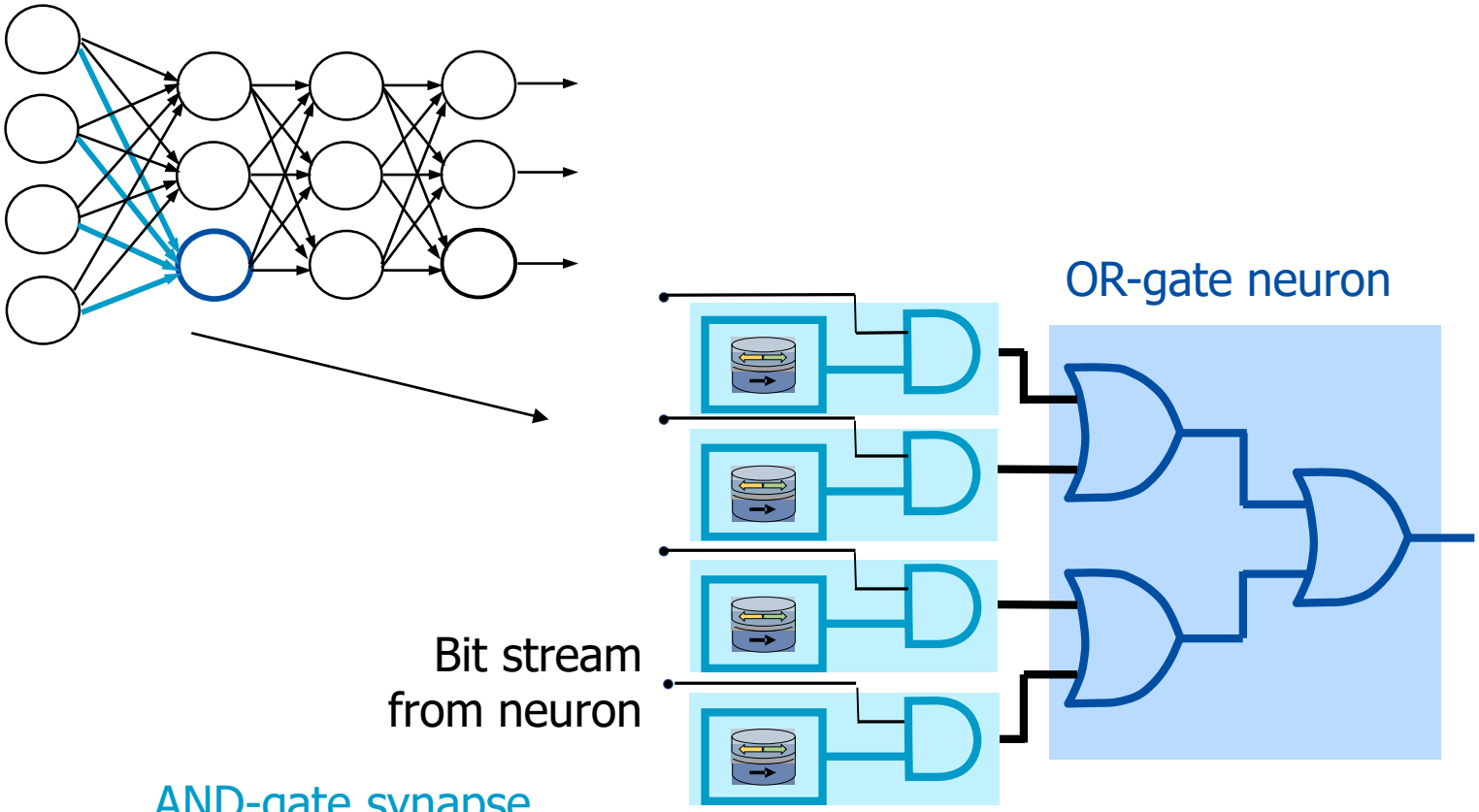
= 1011011110 (0.7)



$$p_{\mathrm{OR}} = p_a + p_b - p_a p_b$$

OR-gate neuron does not work with correlated bit streams

Low area, high efficiency neural network with stochastic information throughout the calculation



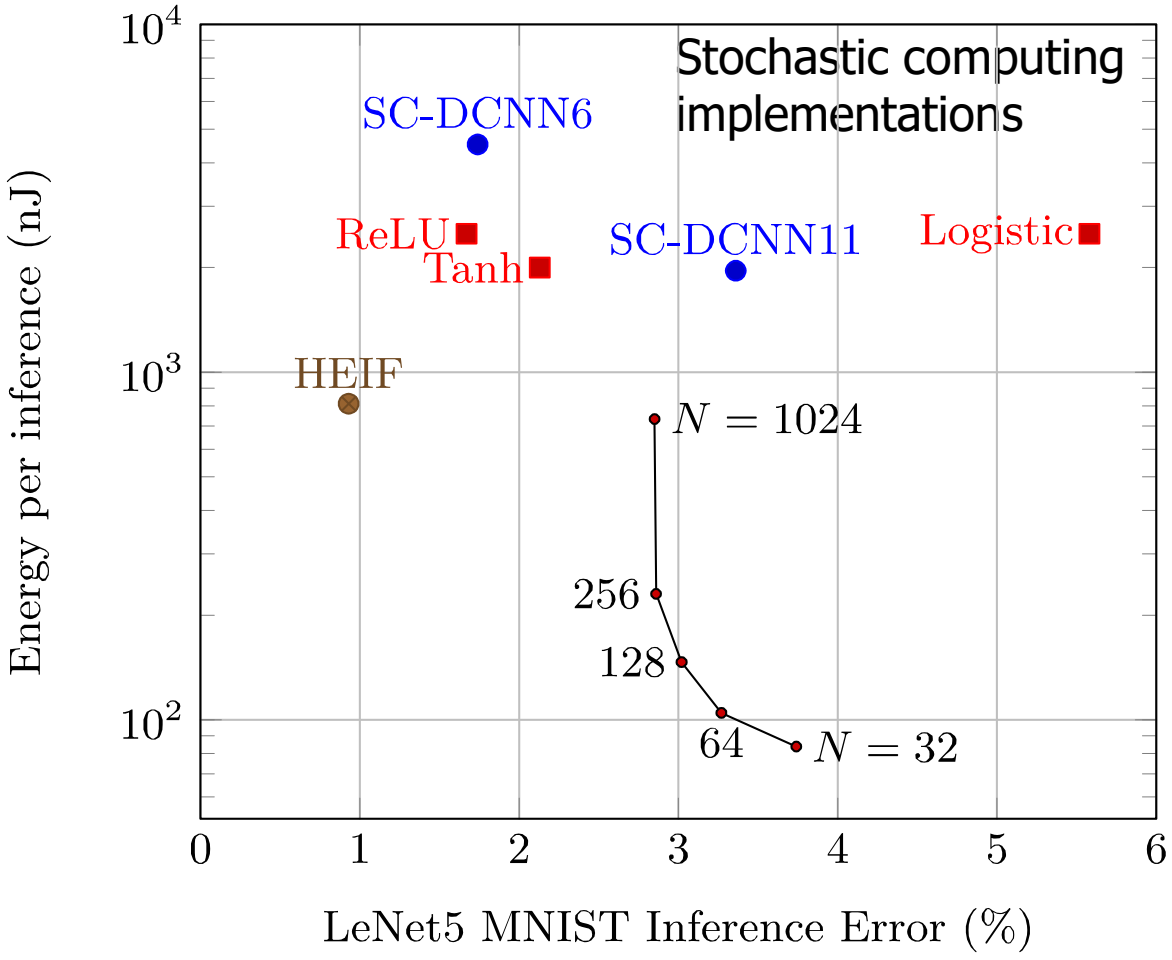
AND-gate synapse

Synaptic weight: Bit stream from programmable SMTJ generator

Efficient random bitstream generation and avoiding domain translation – energy efficiency on a standard problem

- Energy/performance tradeoff by tuning circuit parameters.
- Running for longer or shorter total time (~N) gives energy/performance tradeoff.

Future: implement primitives to check feasibility with realistic MTJs



A standard deep neural network structure

Standard test dataset

Collaborators

NIST National Institute of Standards and Technology U.S. Department of Commerce

Brian Hoskins, **Matthew Daniels**, Guru Khalsa, Mark Anders, Jonathan Goodwill, Jabez McClelland, Nikolai Zhitenev, William Rippard, Matthew Pufall, Emilie Jué, Mark Stiles

University of Maryland

Alice Mizrahi, Advait Madhavan, Philippe Talatchian

The George Washington University, Washington, DC

Siyuan Huang, Gina Adam

UC Santa Barbara

George Tzimpragos, Tim Sherwood

cnrs Thales

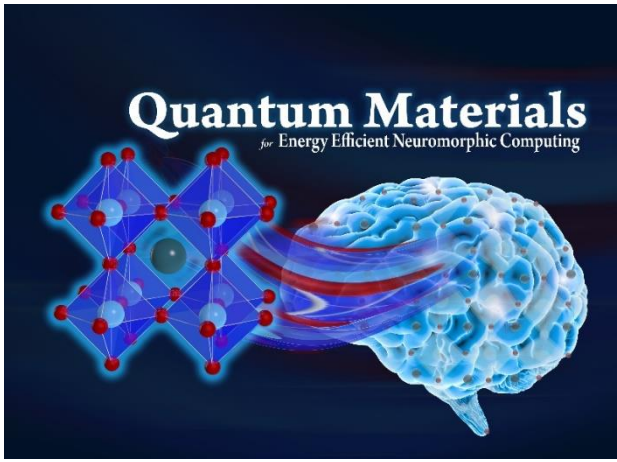
Jacob Torrejon, Mathieu Riou, Flavio Abreu Araujo, Paolo Bortolotti, Vincent Cros, Julie Grollier

C2N

Tifenn Hirtzlin, Damien Querlioz

AIST National Institute of Advanced Industrial Science and Technology (AIST)

Sumito Tsunegi, Kay Yakushiji, Akio Fukushima, Hitoshi Kubota, Shinji Yuasa

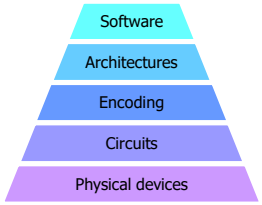


Energy Frontier Research Center
Department of Energy

Vivek Amin, Jonathan Gibbons, Paul Haney, Axel Hoffmann, Julie Grollier

Using magnetic tunnel junctions to compute like the brain

- Emulating features of the brain can increase efficiency for cognitive computing.
- Applications require designing across the computational stack
  - Architecture
  - Encoding
  - Circuitry
  - Devices



- CMOS is likely to play an important role in any room-temperature computing scheme.
- Magnetic tunnel junctions
  - Multifunctional
  - Already in CMOS Fabs

Architecture	**Encoding**	**Devices**
Population coding	Fluctuation rates	Superparamagnetic tunnel junctions
Synaptic weights	Binary	Unstable magnetic tunnel junctions
Neural network with stochastic computing	Random bitstreams	Superparamagnetic tunnel junctions, Repurposed CMOS Logic gates