SPICE
SPIN PHENOMENA
INTERDISCIPLINARY CENTER

Tutorial: *p*-bits for Probabilistic Spin Logic

Kerem Y. Camsari
Postdoctoral Researcher

PURDUE UNIVERSITY

NSF

CAPSL
Computing Advances by Probabilistic Spin Logic

SRC® JUMP
Joint University Microelectronics Program

SRC® nCORE
Semiconductor Research Corporation

p-bits for Probabilistic Spin Logic (PSL)

Zhihong Chen

Supriyo Datta

(1) Camsari et al., ***Phys. Rev. X (2017)***
(Computing with p-bits)
(2) Camsari et al., ***IEEE EDL (2017)***
(Stochastic MTJ-based p-bit)
(3) Camsari et al., ***arXiv:1810.XXXX (2018)***
(Quantum Annealing with p-bits)
(4) Camsari et al., ***arXiv:1809.04028(2018)***
(Review of PSL)

Punyashloka Debashis

Ramtin Zand (UCF)

Ron DeMara (UCF)

Sayeef Salahuddin (UC Berkeley)

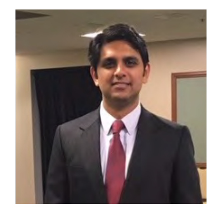
Anirudh Ghantasala

Shuvro Chowdhury

Behtash Behin-Aein (GF)

Brian M Sutton

Zeeshan Pervaiz

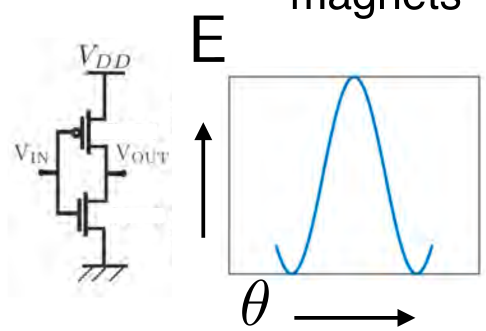


Rafatul Faria

Orchi Hassan

bits, *p*-bits and *q*-bits

transistors / Stable magnets

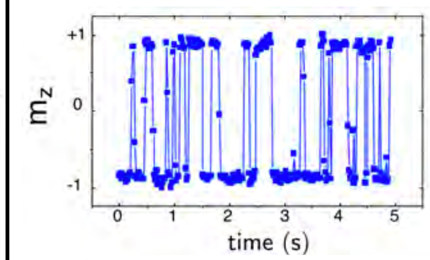


bits

Deterministic
“0” or “1”

digital circuits

Unstable magnets



***p*-bits**

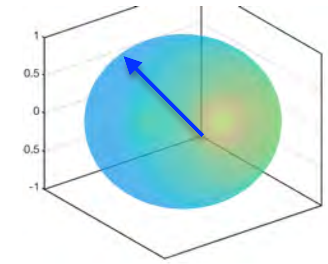
Stoc
“0” or “1

Feynman
1982

p-circuits

p-computer

Single
Spin



***q*-bits**

S erposition

“0” and “1”

q-circuits

q-computer

Digital electronics .. notion of a “bit” , charge on a capacitor

Experimental data by Debashis / Chen

I will make a concrete p-bit to q-bit connection

Application space for p-bits

Quantum Computing inspired

Adder ⟷ Subtractor

(1) Subset Sum Problem

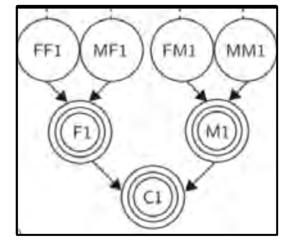
Multiplier ⟷ Divider

(2) Prime Factorization

Camsari et al., ***Phys. Rev. X, (2017)***

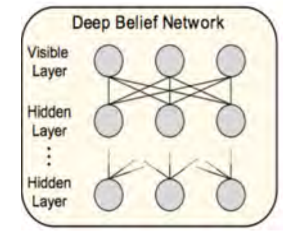
Machine Learning inspired

Bayesian Networks



Faria et al., ***AIP Advances (2018)***

DBN's using p-bits

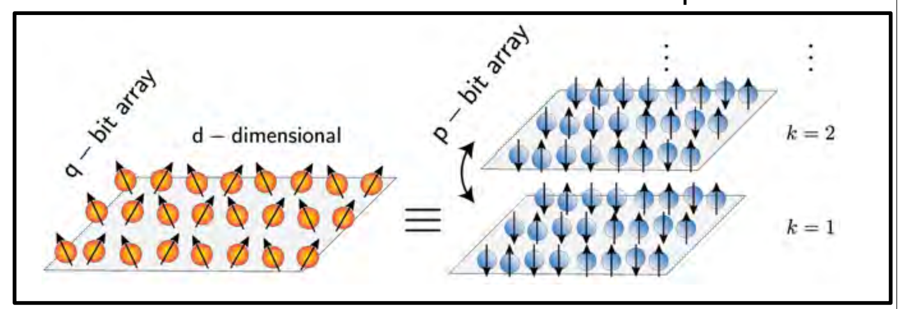


Zand et al., ***GLSVLSI (2018)***

Combinatorial Optimization (e.g. TSP)

Sutton et al., ***Sci. Rep., (2017)***

Quantum circuits emulated with p-bits



Camsari et al., ***arXiv:1810.XXXX***

- **How to build a p-bit?**
- **Quantum Computing-inspired (Reciprocal Networks)**
- **Machine Learning-inspired (Reciprocal and Non-reciprocal Networks)**

p-bit is a binary stochastic neuron

Hardware-specific features

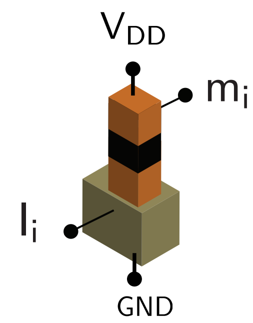
* ***Hardware p-bit is a three-terminal device***

so that it can be interconnected as a transistor-like building block

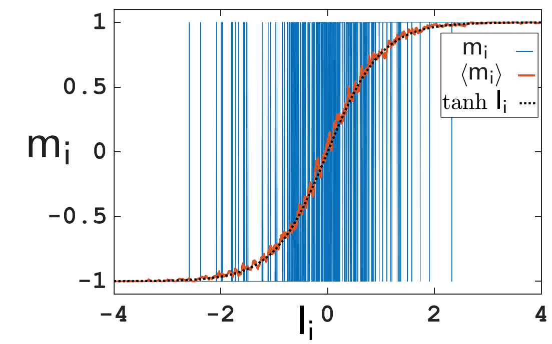
*****update order of p-bits***: needs to be sequential or *asynchronous*

******speed of synapse***: -needs to be much faster than *p*-bit

Need for *hardware* *p*-bits



***p*-bit**



Basic Equations

$$m_i = \text{sgn}[\text{rand}(-1,+1) + \tanh(I_i)] \quad (1)$$

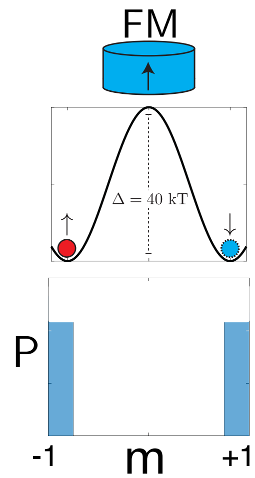
p-bit: Stochastic binary neuron

$$I_i = I_0 \left(\sum J_{ij} m_j + h_i \right) \quad (2)$$

Synapse: Linear weighted sum

*** Update order: RBM qualification

Low-barrier magnets (LBM) are natural p-bits

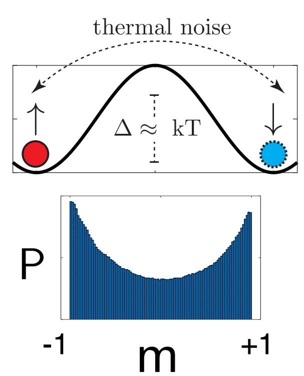


$\tau = \tau_0 \exp(\Delta/kT)$

$\tau_0 \approx$ ps – ns

$\Delta = 40 - 60$ kT

$\tau \approx$ years



$\Delta = kT$

$\tau \approx$ ps – ns

$\Delta \propto H_K \; (M_s \mathrm{Vol.})$

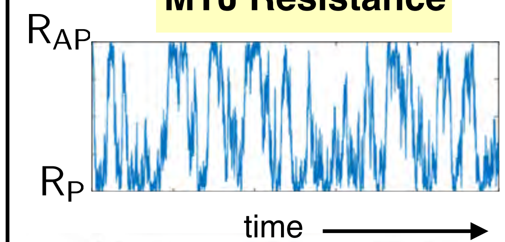
Reducing anisotropy

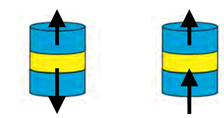
- > PMA thickness
- Circular

Reducing magnetic moment

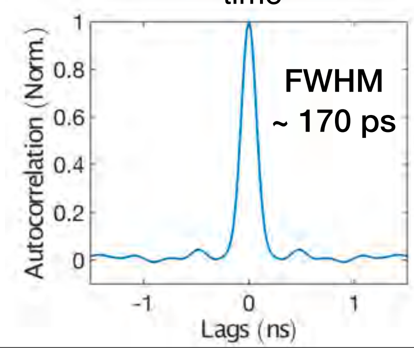
- Low volume
- Low sat. mag.

MTJ Resistance





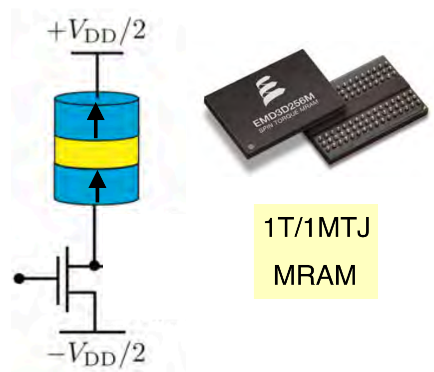
$R_{AP} > R_P$



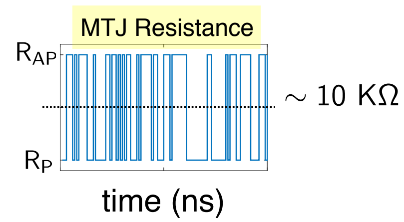
- Circular low barrier magnets can be ***fast***

$\tau^{-1} \propto \alpha\gamma H_D$

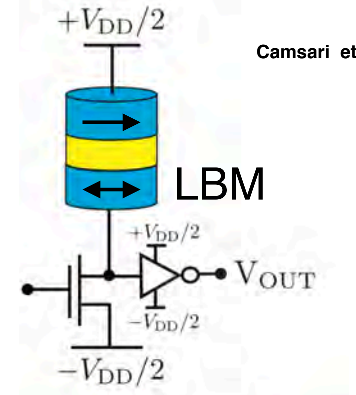
Hardware mapping for *p*-bit



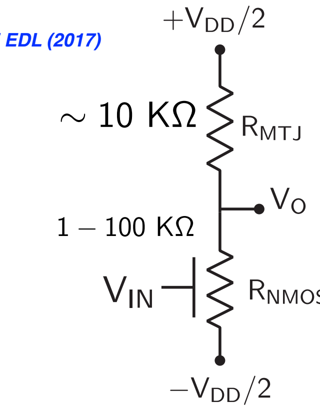
1. Replace stable Free Layer with LBM

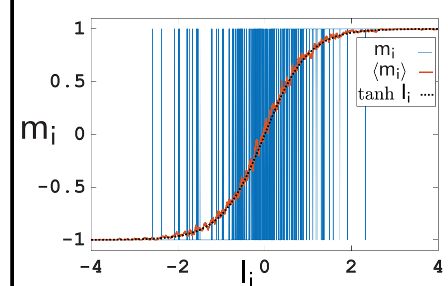


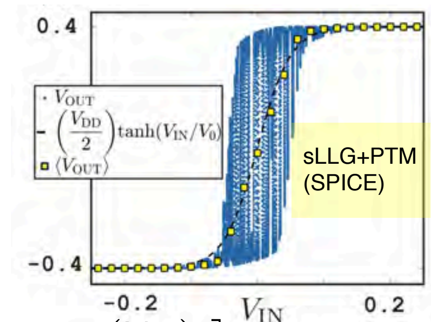
2. Add inverter to DRAIN (1 PMOS/1NMOS)



Camsari et al., *IEEE EDL (2017)*





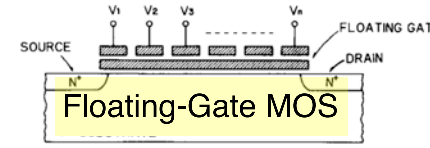


$$\underbrace{\left(\frac{V_{OUT}}{V_{DD}/2}\right)}_{m_i} \approx \mathrm{sgn}\left[r + \tanh \underbrace{\left(\frac{V_{IN}}{V_0}\right)}_{I_i}\right]$$

Possibilities for hardware synapse

$m_i = \mathrm{sgn}[\mathrm{rand}(-1,+1) + \tanh(I_i)]$ (1)

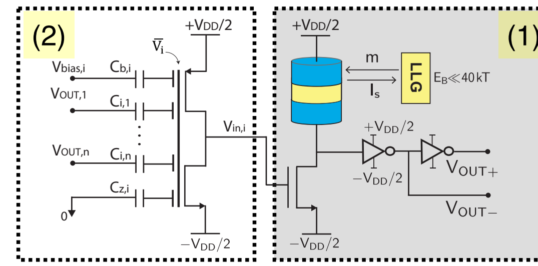
$$I_i = I_0 \left(\sum J_{ij} m_j + h_i \right) \quad (2)$$



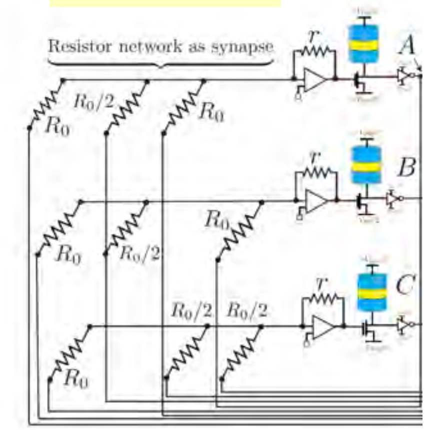
Neu-MOS, Ohmi (1992), IEEE TED

Capacitive cross-bar $[C]_{ij}$

Hassan et al., *arXiv:1801.0926*

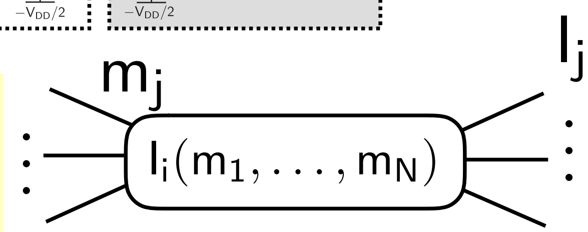


(mem)resistor cross-bar $[G]_{ij}$



Camsari et al., IEEE *EDL (2017)*

CMOS circuits

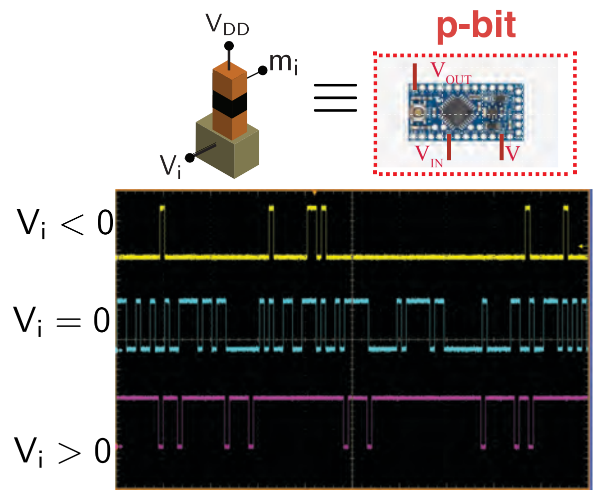


might enable ***non-linear synapses,*** *e.g.* $I_j \propto m_1 m_2 W_{21j}$

->

Non-magnetic *p*-bits/*p*-circuits

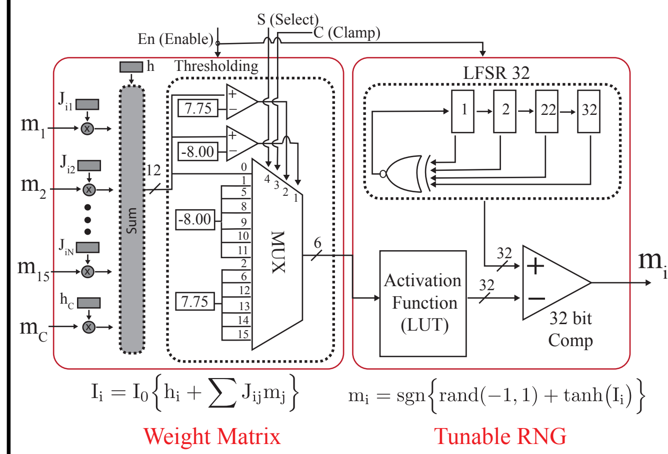
Micro-controller Implementation



Pervaiz et al., ***Sci. Rep. (2017)***

- Demonstrated invertible p-circuits up to ~50 p-bits: 4-bit Integer Factorization

FPGA implementation



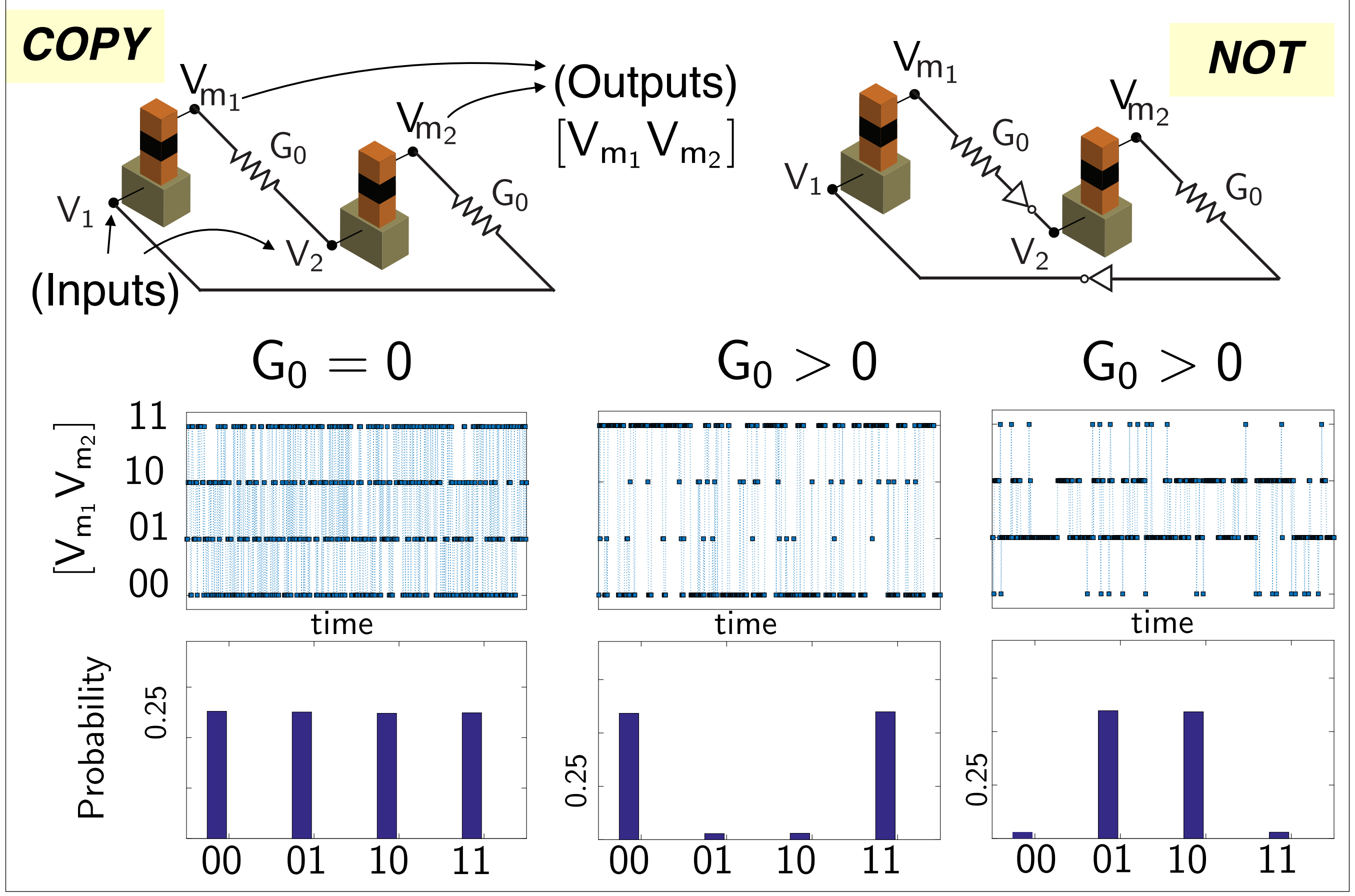
Pervaiz et al., ***IEEE TNNLS (2018)***

- Demonstrated p-circuits up to ~500 p-bits: Solved a 16 bit Subset Sum Problem

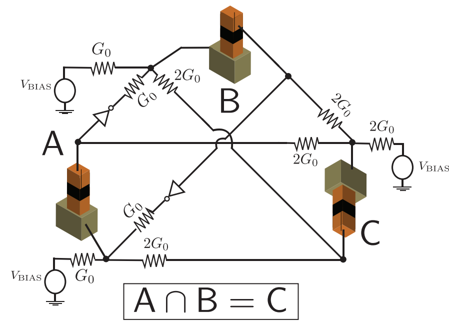
To appear in TNNLS

- How to build a p-bit?
- **Quantum Computing-inspired (Reciprocal Networks)**
- Machine Learning-inspired (Reciprocal and Non-reciprocal Networks)

Correlated *p*-bits: *p*-circuits



Correlated *p*-bits: *p*-circuits



$$m_i = \mathrm{sgn}[\mathrm{rand}(-1,+1) + \tanh(I_i)]$$

$$I_i = I_0\left(\sum J_{ij} m_j + h_i\right)$$

Basic equations

Truth Table

A	**B**	**C**
0	0	0
0	1	0
1	0	0
1	1	1

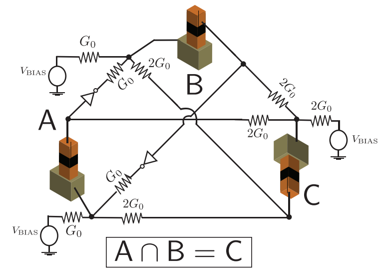
↓

$[J_{AND}]$

↓

$[G]_{ij}$ network

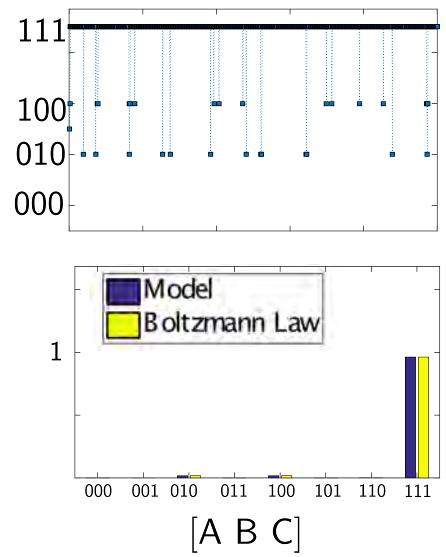
Invertible Boolean Logic

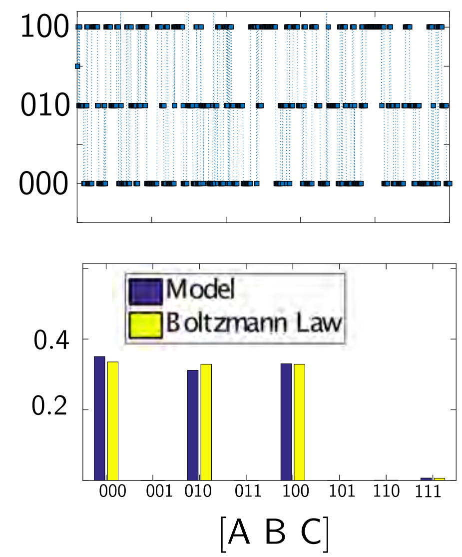


$$m_i = \text{sgn}[\text{rand}(-1,+1) + \tanh(I_i)]$$

$$I_i = I_0 \left(\sum J_{ij} m_j + h_i \right)$$

{h} used as a handle to "clamp" inputs ***and outputs!***





$h_A \rightarrow +1$

$h_B \rightarrow +1$

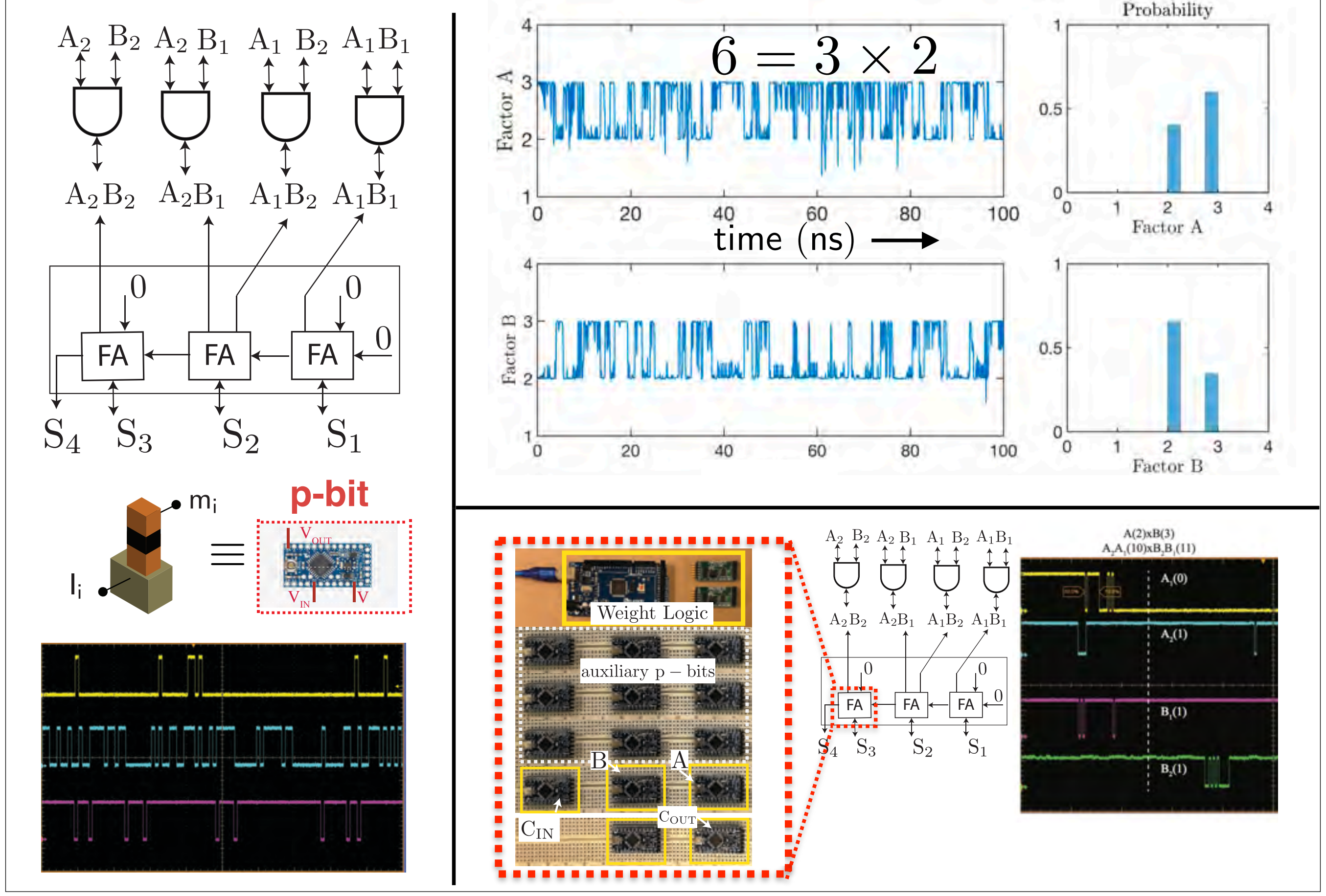
$A \rightarrow 1$

$B \rightarrow 1$

$h_C \rightarrow -1$

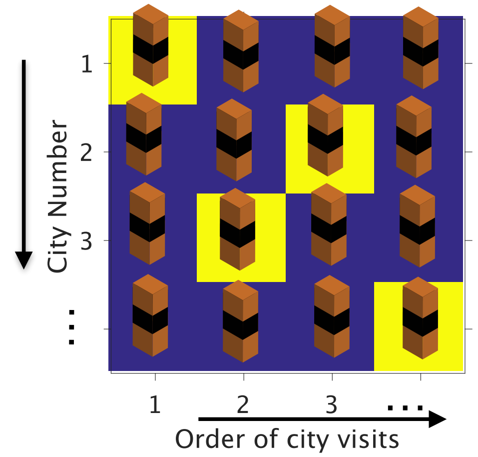
$C \rightarrow 0$

Factorization as *Inverse* Multiplication



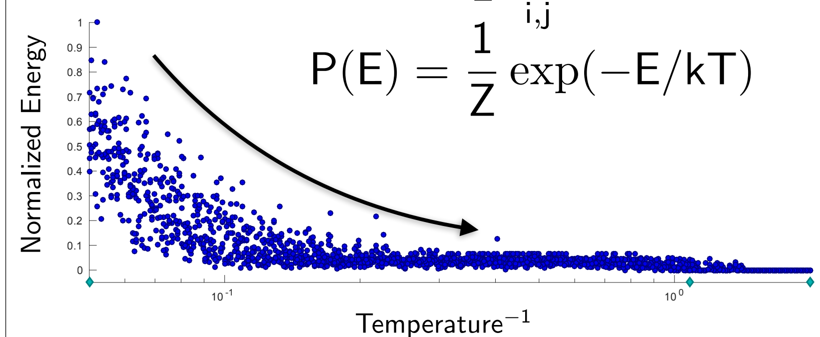
D-Wave reference here

Combinatorial Optimization



$$E(m_1, m_2, \ldots, m_i) = \frac{1}{2}\sum_{i,j} J_{ij} m_j m_i$$

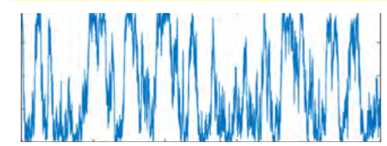
$$P(E) = \frac{1}{Z}\exp(-E/kT)$$



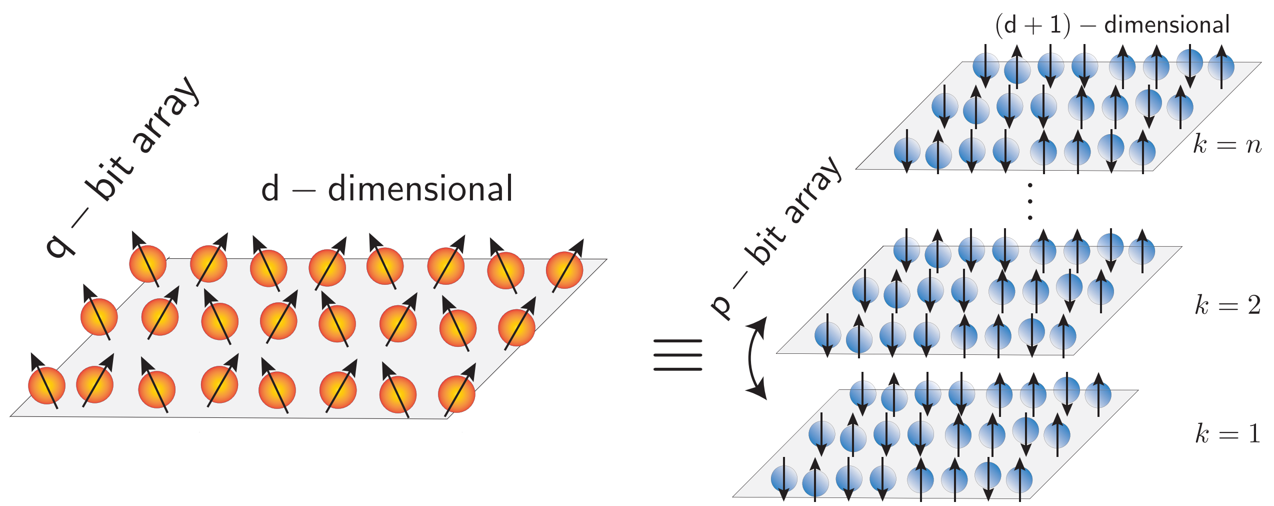
Sutton et al., *Sci. Rep. (2017)*

- **Needs sparse, quantized J**

- **Searches the phase space at *GHz rates***



An *exact* mapping of q-bits to p-bits



$$H_{\text{quantum}} = -\left(\sum_{i<j} J_{i,j}\sigma_i^z\sigma_j^z + \Gamma_x \sum_i \sigma_i^x\right)$$

$$H_{\text{classical}} = -\left(\lim_{n\to\infty}\sum_{k=1}^{n}\sum_{i<j}(J_{\parallel})_{i,j}\, m_{i,k}m_{j,k} + J_{\perp} m_{i,k}m_{i,k+1}\right)$$

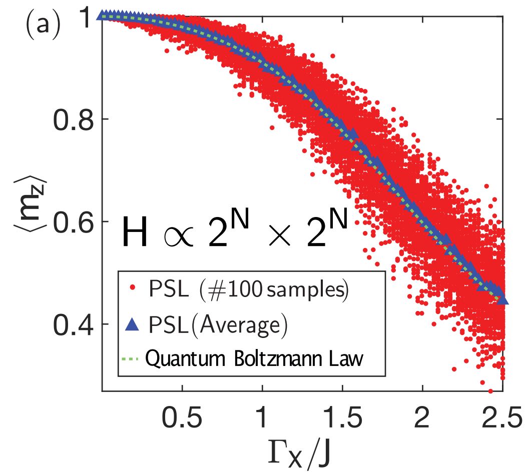
$$J_{\parallel} = J/n \quad J_{\perp} = -\frac{1}{2\beta}\log\, \tanh(\beta\Gamma/n)$$

- **Trotter-Suzuki Decomposition**

- **Efficient for "*stoquastic*" Hamiltonians**

Quantum Annealing with p-bits

1D spin-chain with M=8 q-bits (J=+1)



$$H_{quantum} = -\left(\sum_{i<j} J_{i,j}\sigma_i^z\sigma_j^z + \Gamma_x \sum_i \sigma_i^x\right)$$

$$\langle m^z \rangle = \frac{Tr.[m^z \exp(-\beta H)]}{Tr.[\exp(-\beta H)]}$$

- **"Quantum" Boltzmann Law**

n =250 replicas

$$H_{classical} = -\left(\lim_{n\to\infty} \sum_{k=1}^{n} \sum_{i<j} (J_{\|})_{i,j}\ m_{i,k}m_{j,k} + J_{\perp} m_{i,k} m_{i,k+1}\right)$$

- **Replicated p-bits vs Exact solution: Averages and *correlations match***

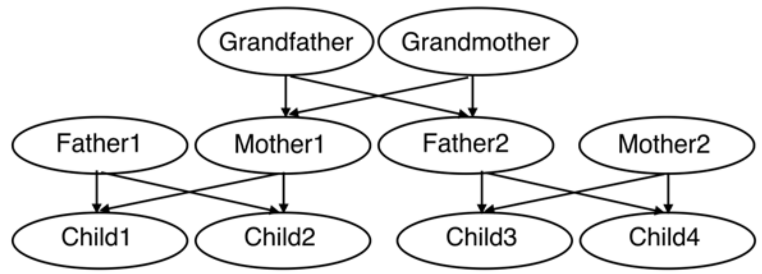
Comment: SPICE simulation

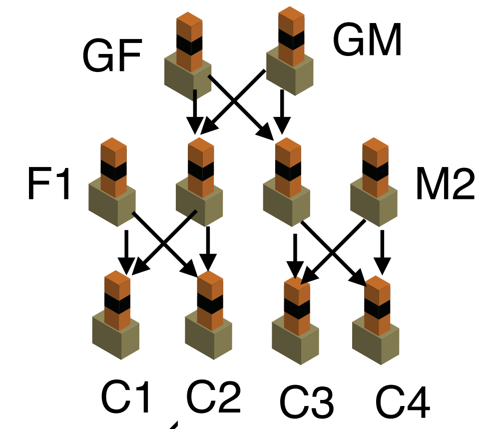
Comment: Size of matrix 2^N by 2^N

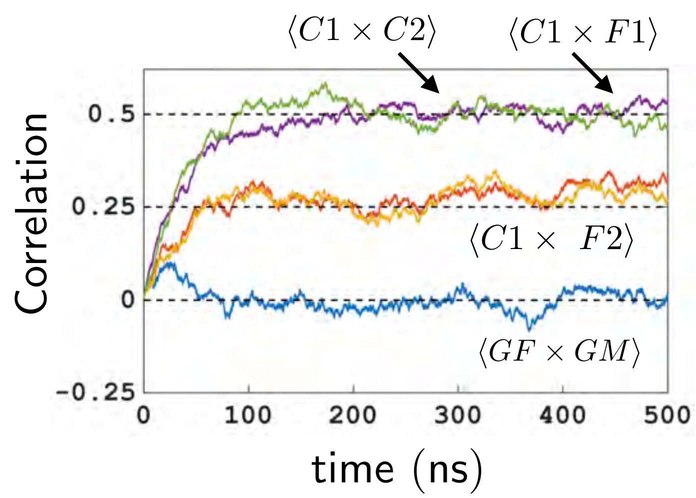
- How to build a p-bit?
- Quantum Computing-inspired (Reciprocal Networks)
- **Machine Learning-inspired (Reciprocal and Non-reciprocal Networks)**

Bigger outline showing previous parts

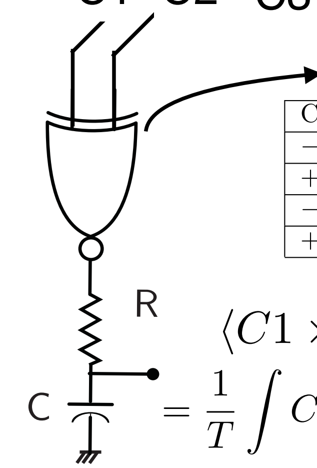
Inherently probabilistic circuits: Bayesian Networks







Faria et al., *AIP Advances (2018)*

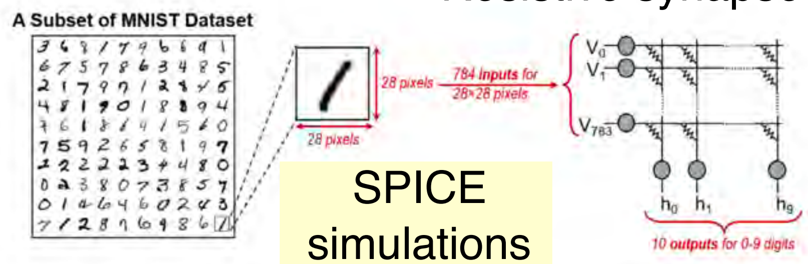


C1	C2	OUT
−1	−1	+1
+1	−1	−1
−1	+1	−1
+1	+1	+1

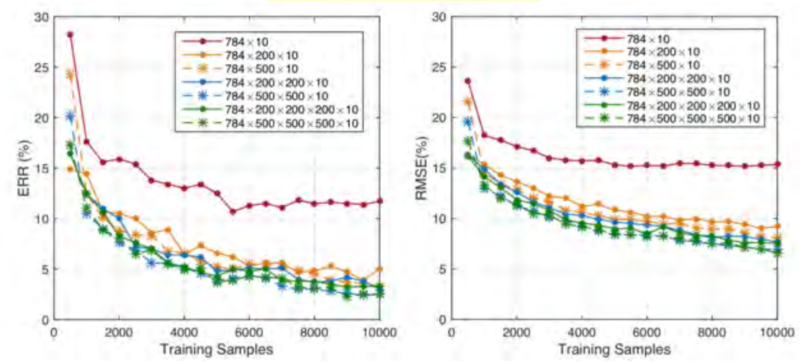
$$\langle C1 \times C2 \rangle = \frac{1}{T}\int C1(t)\ C2(t)dt$$

Low-power DBN: Inference with p-bits

Resistive synapse



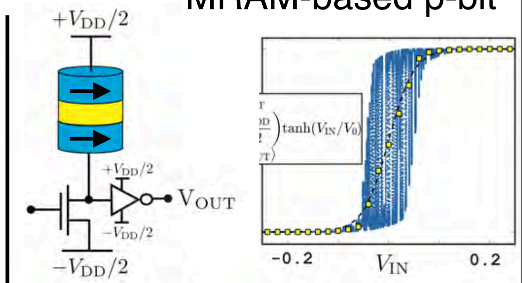
SPICE simulations



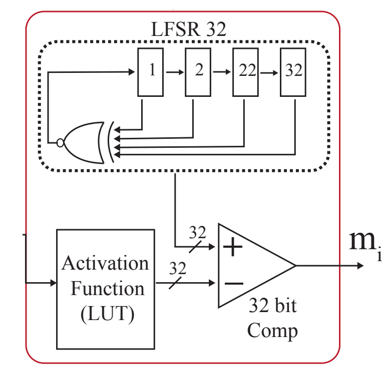
- Energy x Accuracy Tradeoff via different topologies

Ramtin Zand (UCF) et al. (2018)

MRAM-based p-bit



p-bit as **Neuron**: Area & Energy savings compared to a *digital* neuron



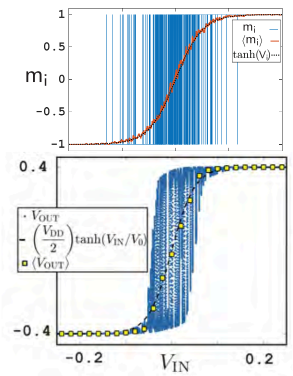
True North reference: LFSR's are expensive
Of course, digital neurons are more reliable and probably offer better scaling / fully thresholded.

Full SPICE could be misleading.

Energy X 1000, could it be defended?

Summary

p-bit as hardware building block

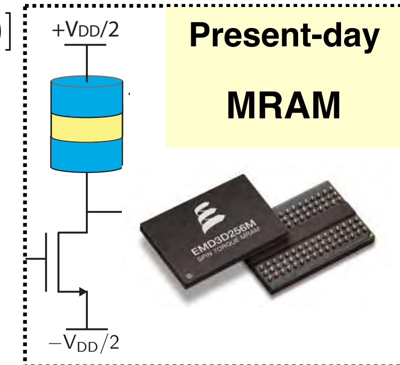


$$m_i = \mathrm{sgn}[\mathrm{rand}(-1,+1) + \tanh(I_i)]$$

$$I_i = I_0\left(\sum J_{ij} m_j + h_i\right)$$

$\approx$ one-to-one mapping

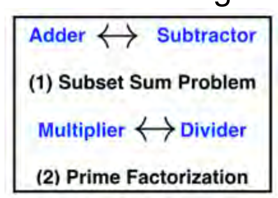
Synapse
$[C]_{ij}$ network
$[G]_{ij}$ network
CMOS



Quantum Computing

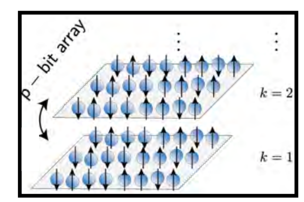
Machine Learning

Invertible Logic

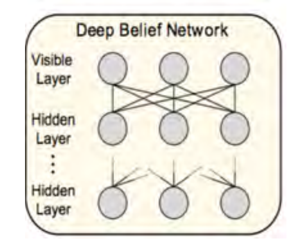


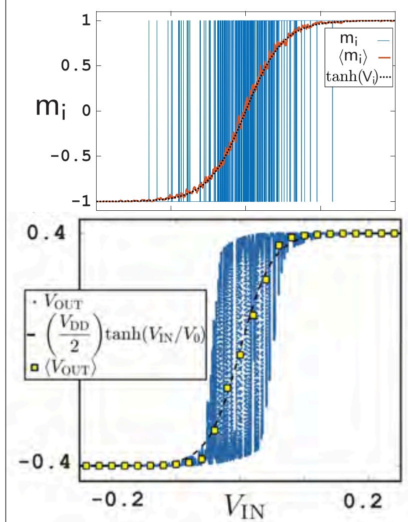
Optimization

Quantum Circuits



RBM, Bayesian, …





p-bits for PSL

$$m_i = \mathrm{sgn}[\mathrm{rand}(-1,+1) + \tanh(I_i)]$$

$$I_i = I_0 \left(\sum J_{ij} m_j + h_i\right)$$

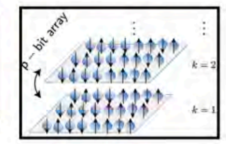
Quantum Computing | *Machine Learning*

Invertible Logic

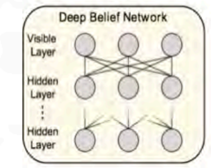
Adder ⟷ Subtractor
(1) Subset Sum Problem
Multiplier ⟷ Divider
(2) Prime Factorization

Optimization

Quantum Circuits



RBM, Bayesian, …



Applications/Algorithms: Quantum Computing / Machine Learning

The End of Moore's Law:
A New Beginning for Information Technology

T. N. Theis and P. Wong, 2016

***p*-bits & *p*-circuits as**

Novel devices and circuits for "***the algorithms and applications of the coming decades***"

p-bits for Probabilistic Spin Logic (PSL)

Zhihong Chen

Supriyo Datta

(1) **Camsari et al.,** ***Phys. Rev. X (2017)***
(Computing with p-bits)
(2) **Camsari et al.,** ***IEEE EDL (2017)***
(Stochastic MTJ-based p-bit)
(3) **Camsari et al.,** ***arXiv:1810.XXXX (2018)***
(Quantum Annealing with p-bits)
(4) **Camsari et al.,** ***arXiv:1809.04028(2018)***
(Review of PSL)

Punyashloka Debashis

Ramtin Zand (UCF)

Ron DeMara (UCF)

Sayeef Salahuddin (UC Berkeley)

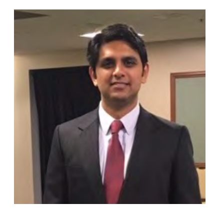
Anirudh Ghantasala

Shuvro Chowdhury

Behtash Behin-Aein (GF)

Brian Sutton

Zeeshan Pervaiz



Rafatul Faria

Orchi Hassan