The Path Toward Energy-Efficient Inference Engine Architectures on Scaled and Beyond-CMOS Fabrics

Ashkan Borna, Makoto Takamiya and Jan M. Rabaey

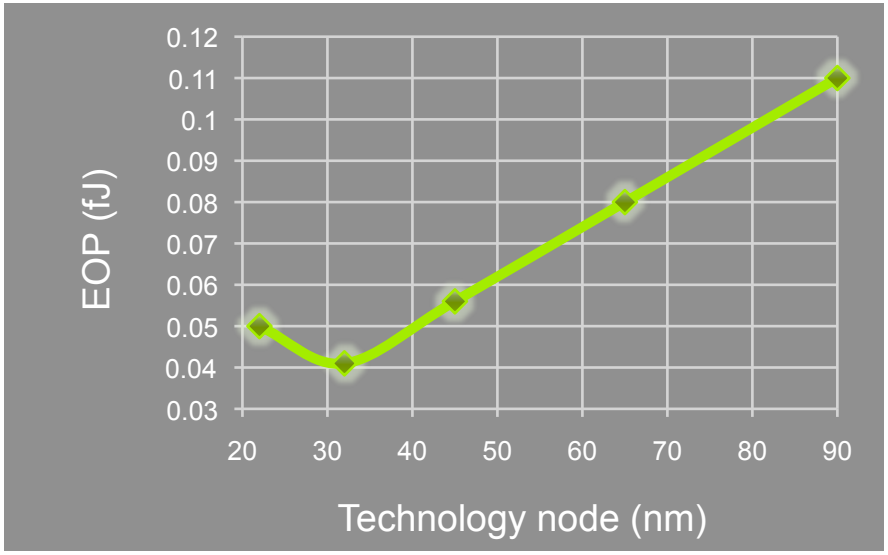
October 28th 2013

Neuro-Inspired Information Processing

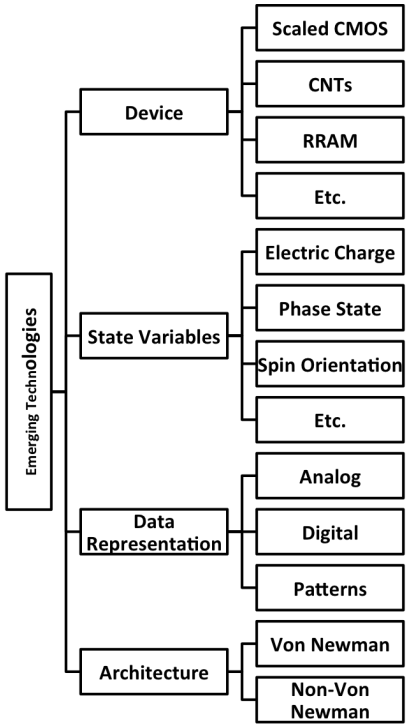
Why? Traditional semiconductor scaling is slowing down

- Energy, low signal-to-noise ratio and variability limit further scaling of semiconductor systems (end of "Moore's Law" ?)

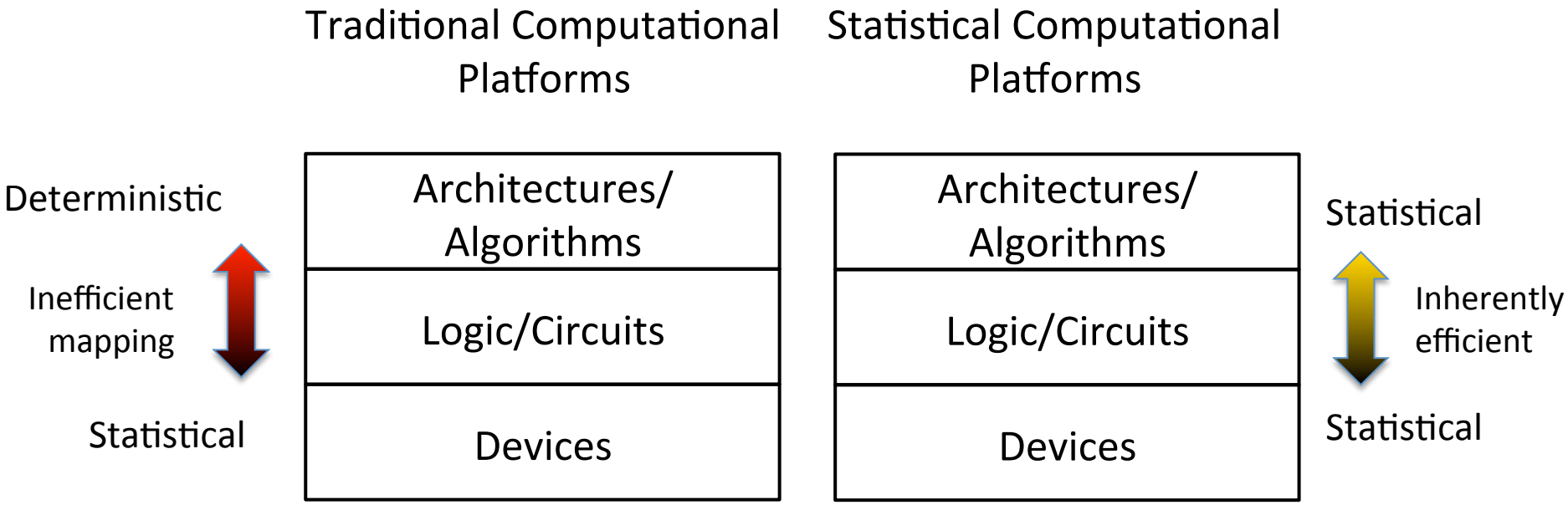
- Exploit properties of neural systems
  - Massively parallel, high density, major redundancy, and adaptivity (learning)
  - Robustness through exploitation of randomness and variability
  - Multiple signal representations in single integrated environment (analog, discrete, digital)



Reduction of energy/operation slowing down



Alternative Computational Paradigms



- Functional non-determinism present in many applications
  - ✓ Feature extraction, classification, recognition, decision making, learning

Metrics

- The means to compare neuro-inspired algorithms are similar to the traditional ones. However, given the difference in representations, metrics may have to be redefined
- Important properties to be measured:
  - Performance, latency
  - Power, energy
  - Robustness
  - Density: given the 3D nature of many of the envisioned implementations, density may be a better measure than area

Dimension of Representation

- A representation is minimal if dimensionality is just sufficient to represent the full signal coverage

- A representation is *hyperdimensional* when the number of dimensions is "much" (> 1000?) larger than needed to cover the space.
  - Hyperdimensional representations are redundant, and most often sparse
  - The axis' are chosen at random

Hyperspace

- Distance between nodes is a binomial distribution

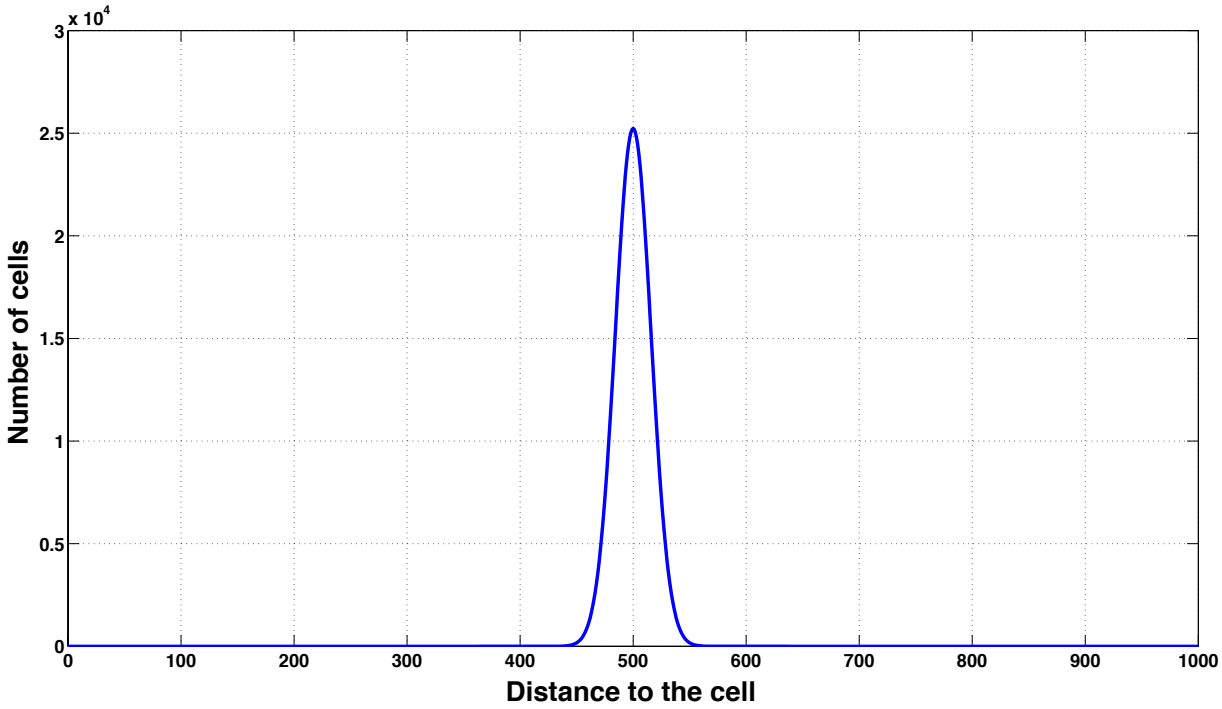
$$dist = (n,k)\, p^k q^{n-k}$$

- Approximated with a Gaussian:

$$Mean = \frac{n}{2}$$

$$STD = \sqrt{npq}$$

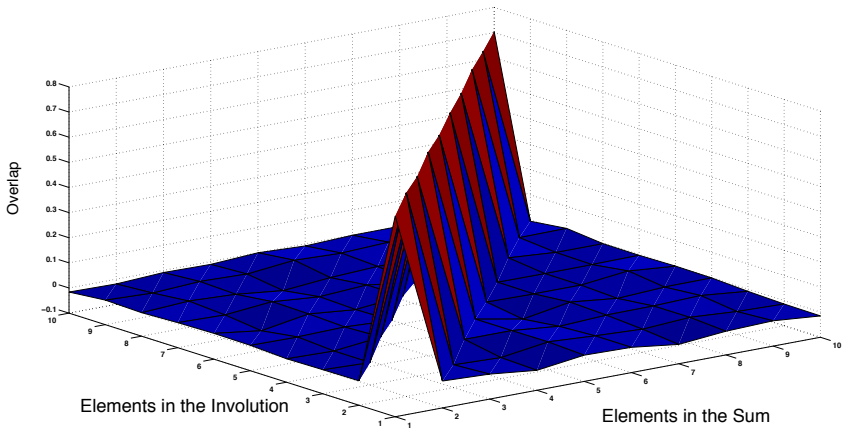
- Space is extremely clustered.

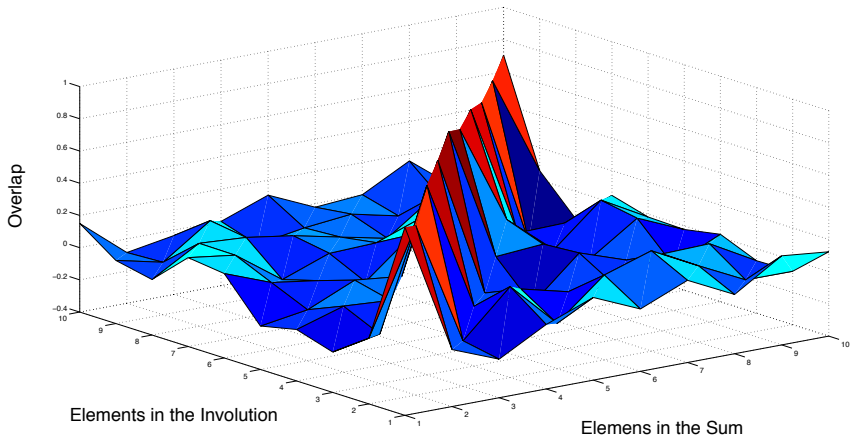


Thousand bit Vectors Space with One Million Nodes

- 1,000-dimensional space has 1,000 orthogonal vectors. But much more nearly orthogonal and that's why we like hyper space

Computation in Hyperdimension

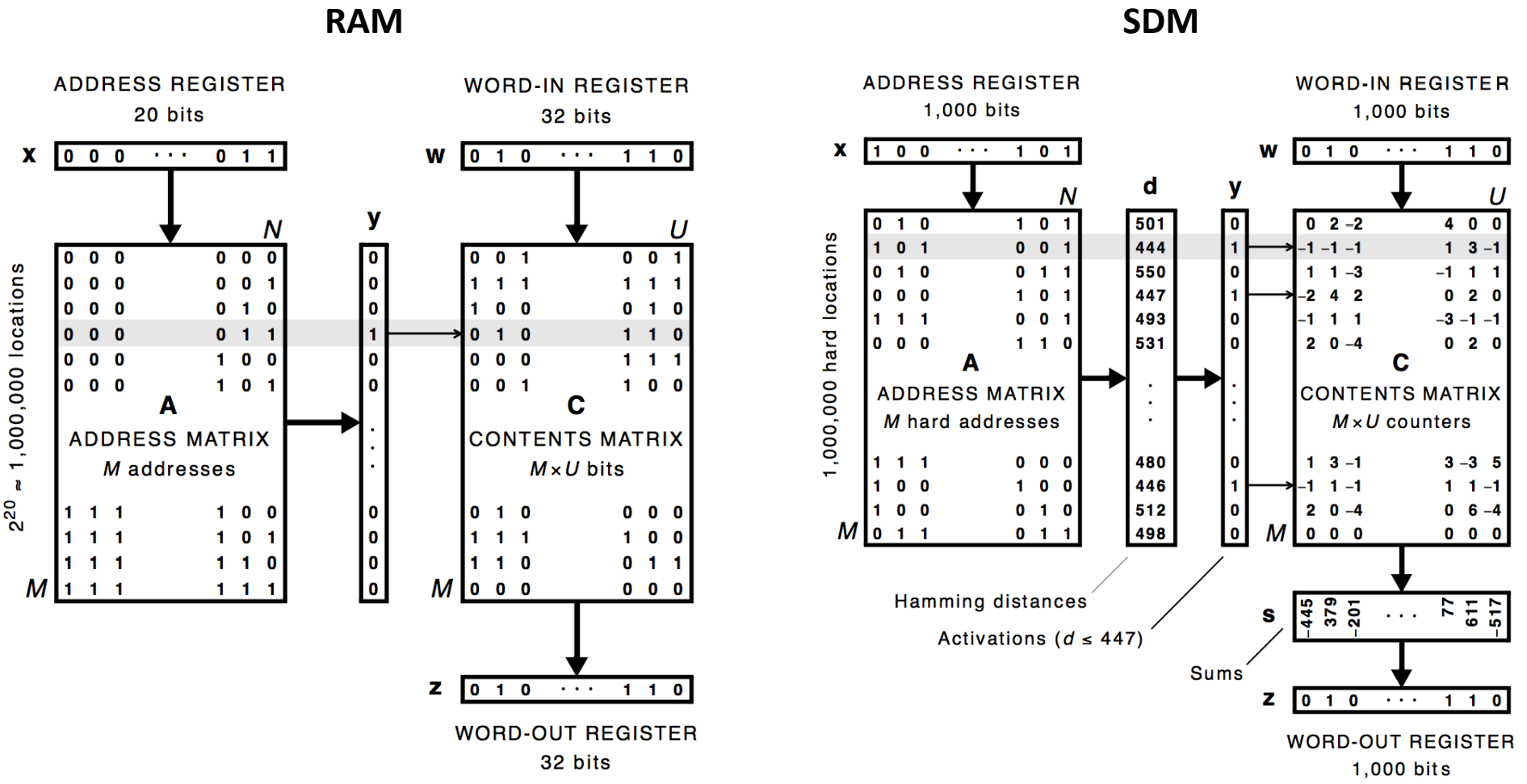




- Ten patterns in the library
- Each pattern has 10000 bits
- High SNR

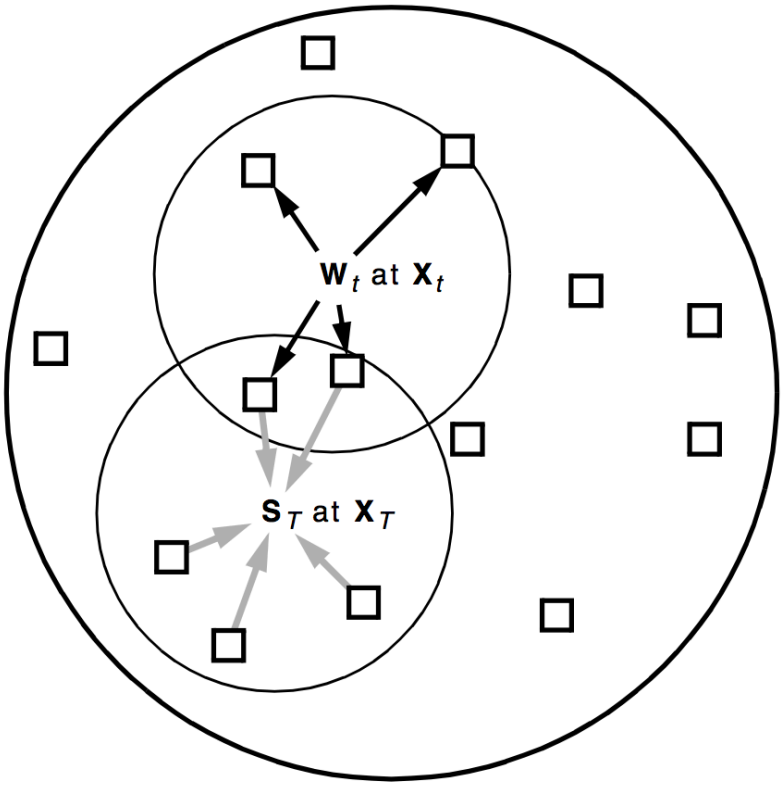
- Ten patterns in the library
- Each pattern has 100 bits
- Low SNR

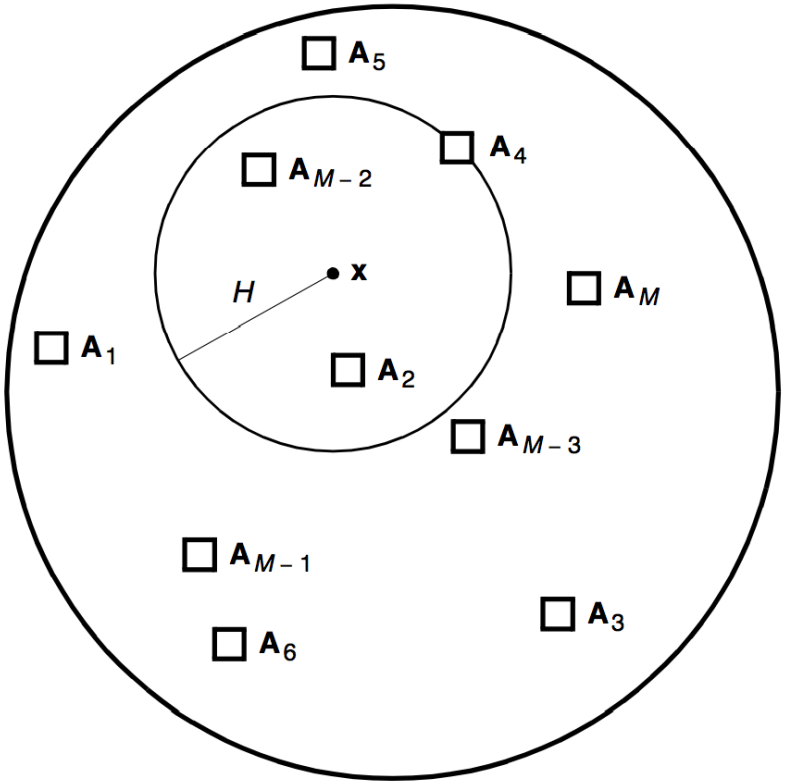
RAM vs. Sparse Memory



"Sparse distributed memory and related models", P. Kanerva

Read and Write Operations

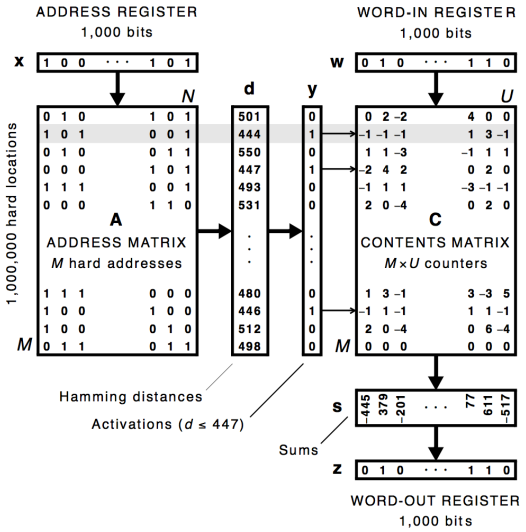




"Sparse distributed memory and related models", P. Kanerva

Fundamental Operations

- N is huge hence 2^N is practically infinite



- Basic Operations:

$$\mathbf{C} = \sum_{t=1}^{T} \mathbf{Y}_t \square \mathbf{W}_t = \sum_{t=1}^{T} y(\mathbf{A}\mathbf{X}_t) \square \mathbf{W}_t$$

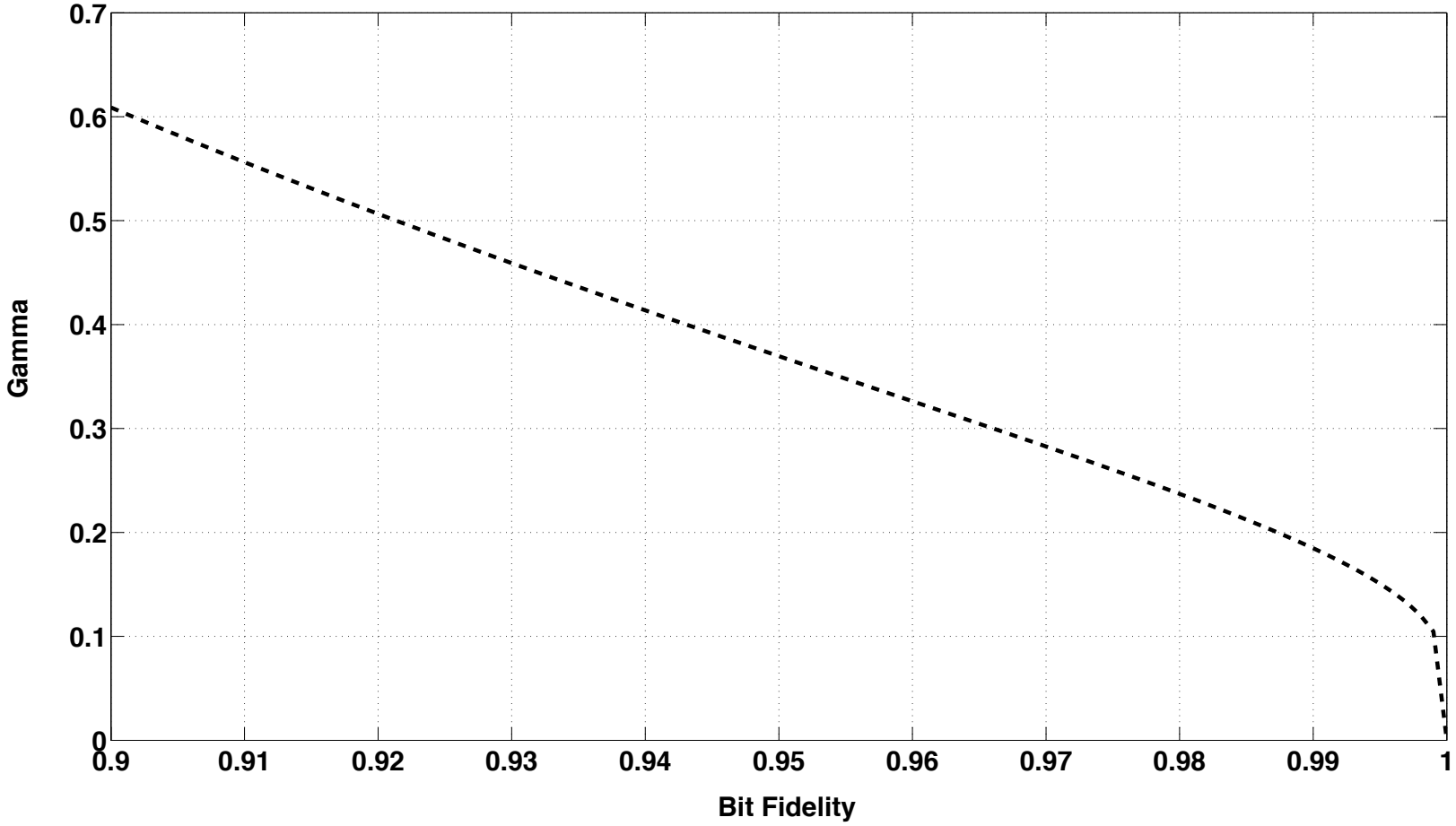
$$\mathbf{Z} - \mathbf{W} = z(\mathbf{Y}\mathbf{C}) - \mathbf{W}$$

$$= z(\mathbf{Y}\mathbf{Y}^{\mathrm{T}}\mathbf{W}) - \mathbf{W}$$

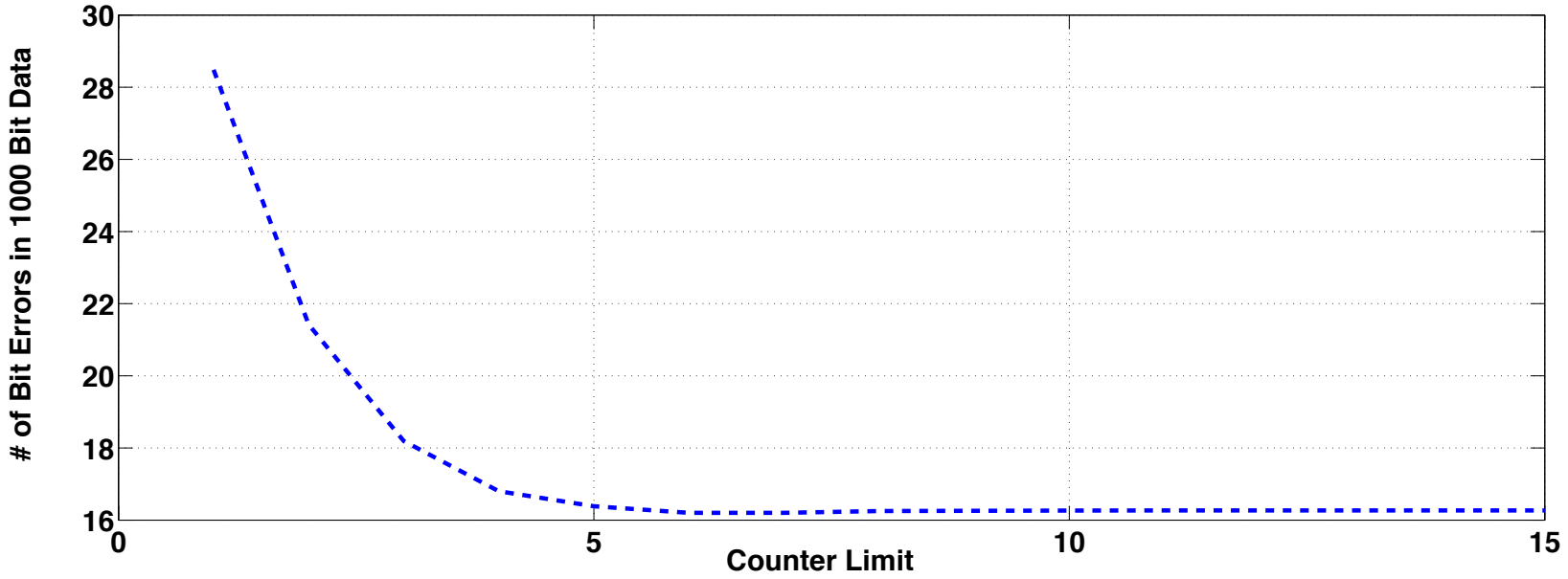
- Each time an address word would activate on the average ρM addresses. ρ would be determined based on bit recovery fidelity and number of available hard locations (M).
- T (Number of saved elements in the memory) is determined based on memory capacity (for the required bit fidelity) and hard locations.

$$\Gamma \approx \frac{1}{\left[\Phi^{-1}(\varphi)\right]} \qquad T = \Gamma M \qquad \rho = \frac{1}{\sqrt[3]{2MT}}$$

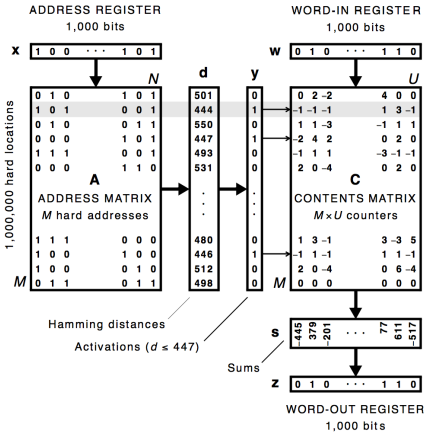
Ratio of Patterns to Physical Locations



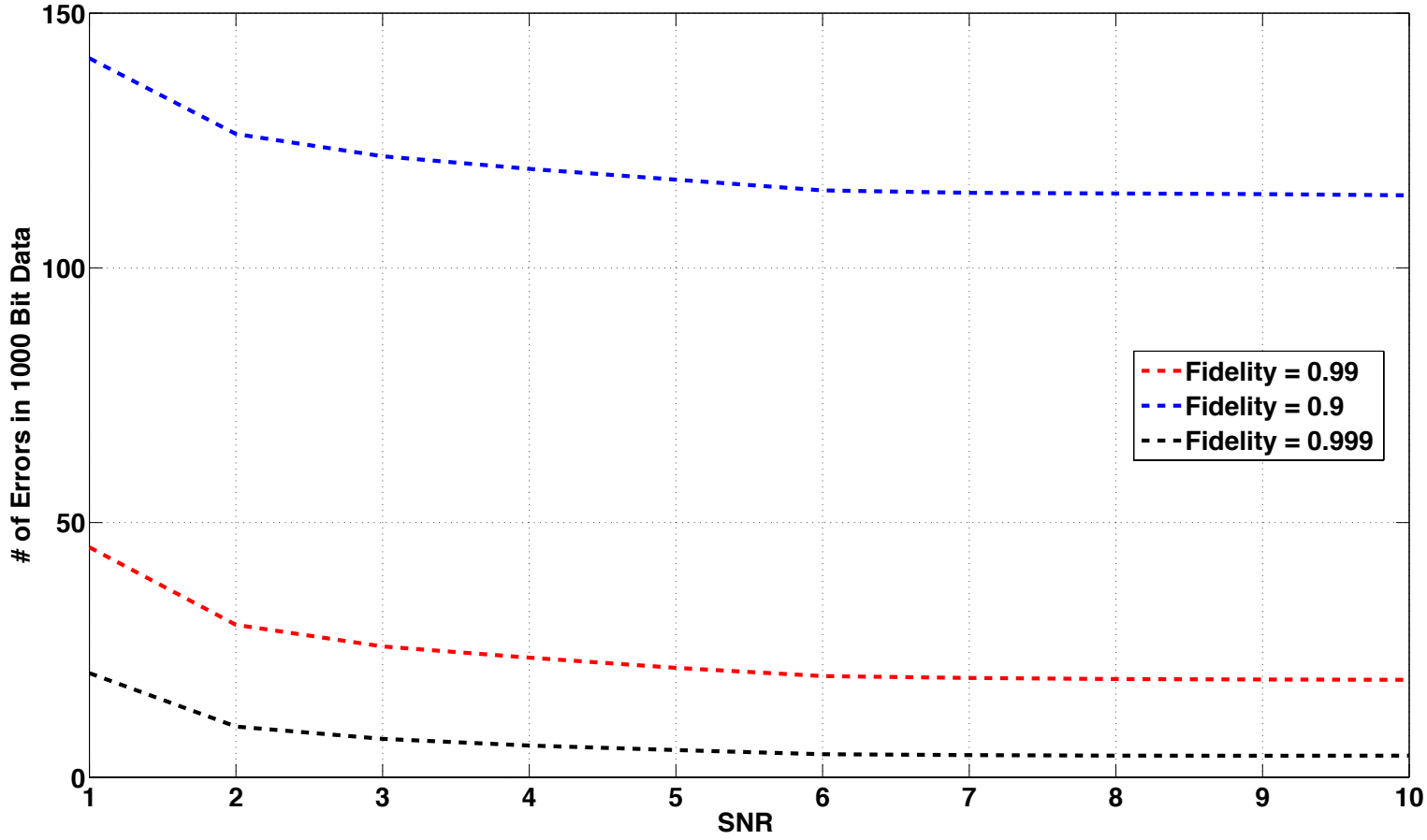
Non-Ideal Effects on SDM Performance (1)



- Number of discrepancies between stored and retrieved data for a 1000 bit data vs. counter limit for Fidelity of 0.99

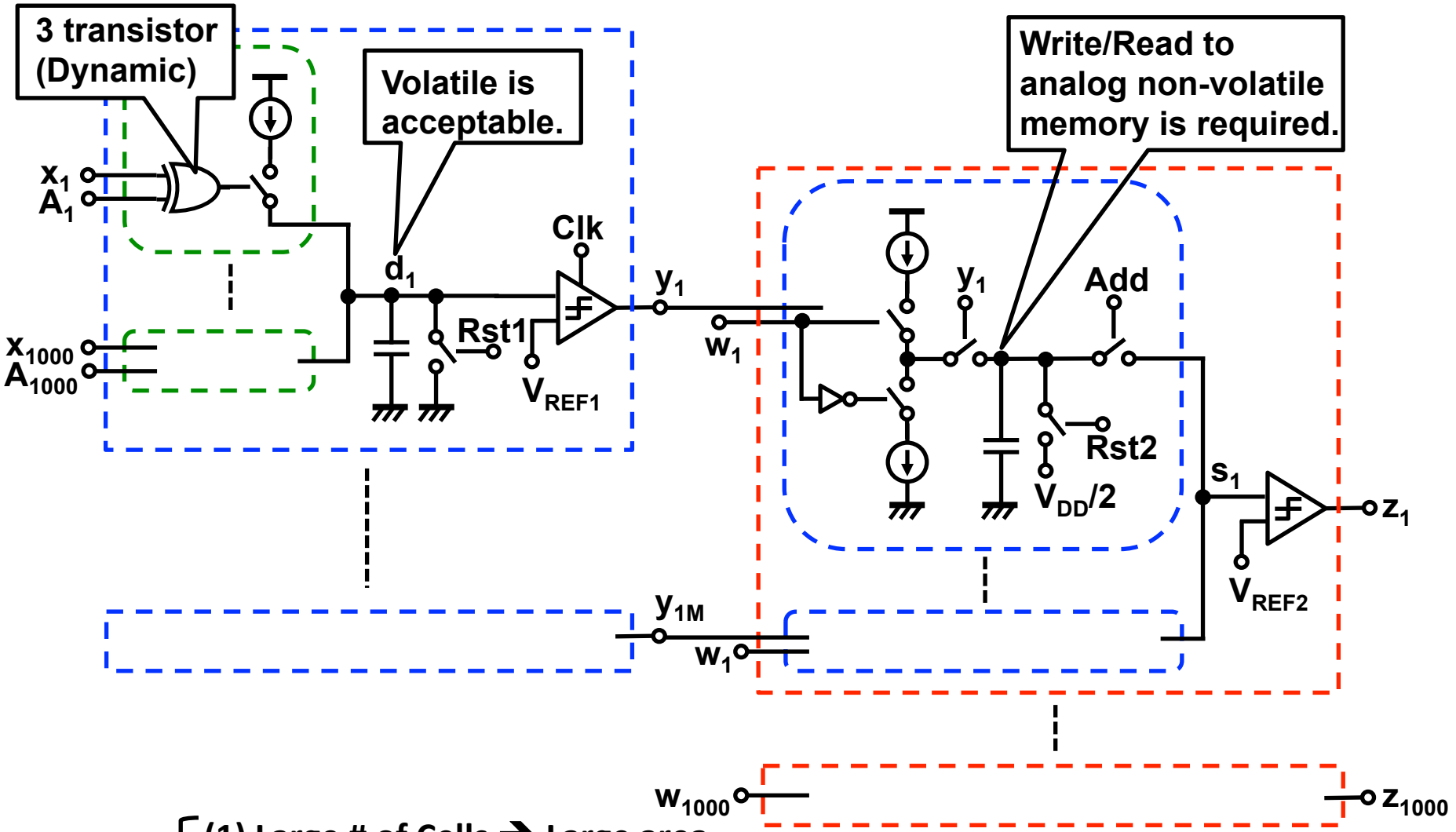


Non-Ideal Effects on SDM Performance (2)



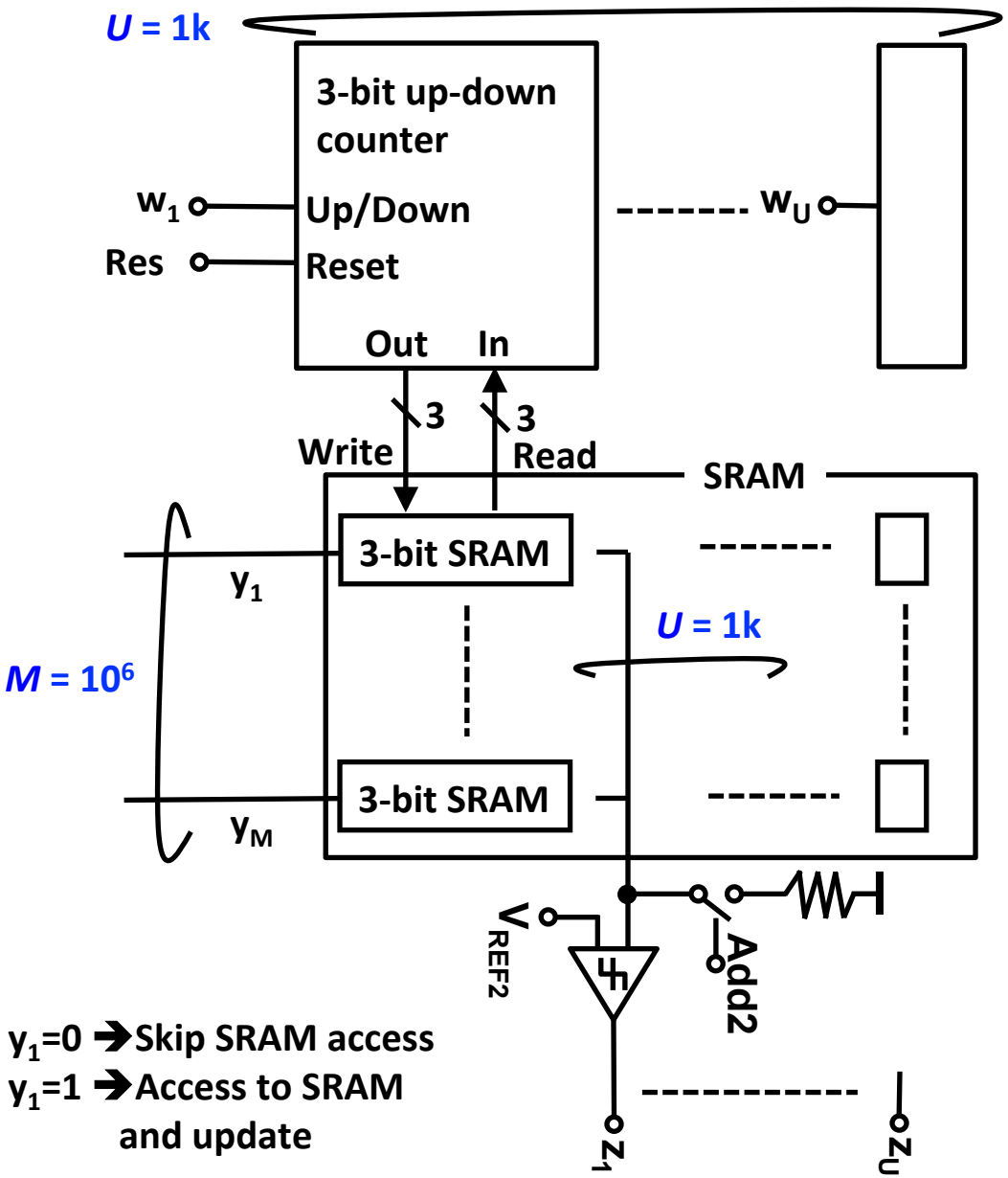
Effect of cells' SNR on the error rates for different fidelities

Circuit Implementation of SDM



Issues
- (1) Large # of Cells ➔ Large area
- (2) Analog DRAM ➔ Volatile memory
- (3) Sum of 1M ➔ Signal-to-noise ratio

Implementation of C Matrix



y_1=0 ➔Skip SRAM access
y_1=1 ➔Access to SRAM and update

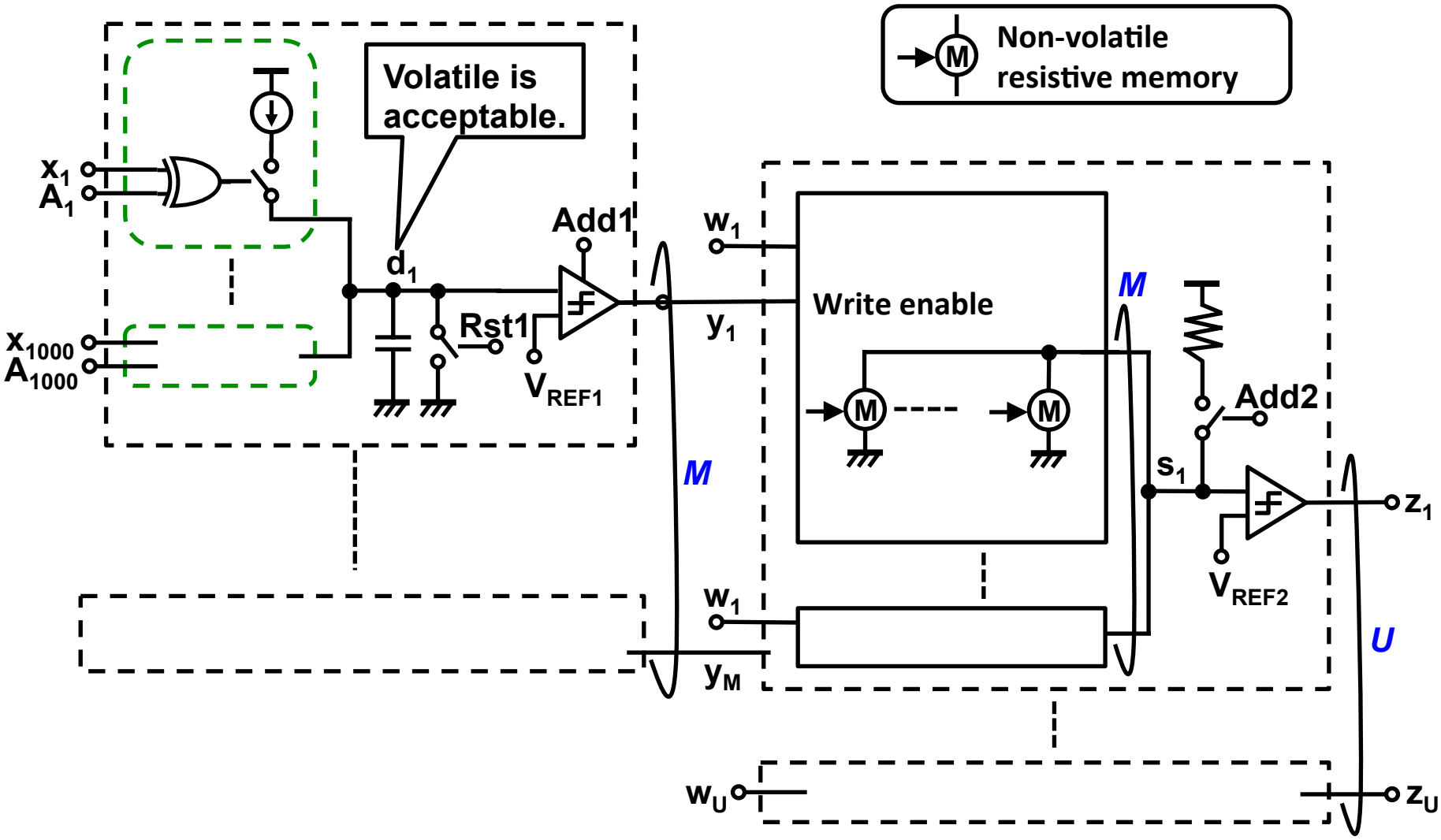
3 issues

(1) Sequential access to 1M row
➔ Long latency

(2) Large SRAM area
The smallest SRAM cell = 0.081mm^2
20nm CMOS (ISSCC2013)
➔20mm^2

(3) Volatile memory

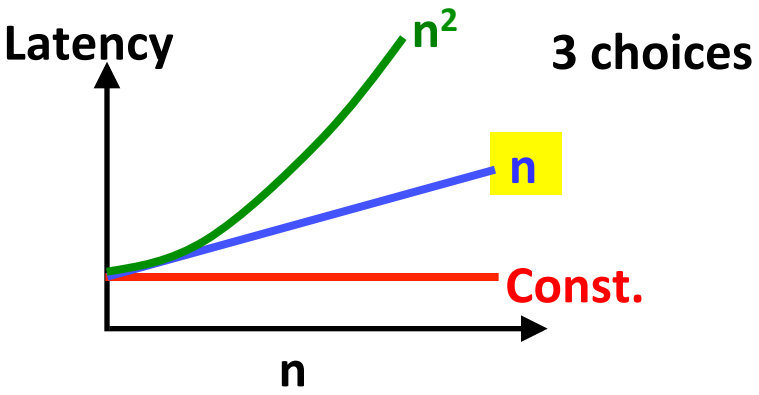
Circuit Implementation of SDM (2)

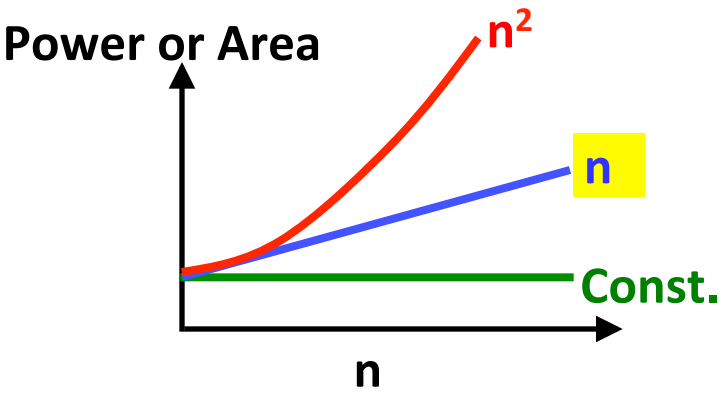


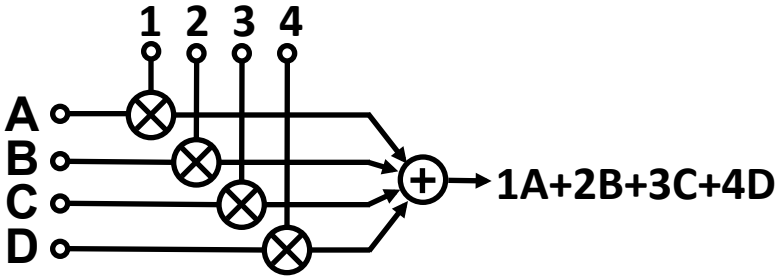
Matrix Multiplication

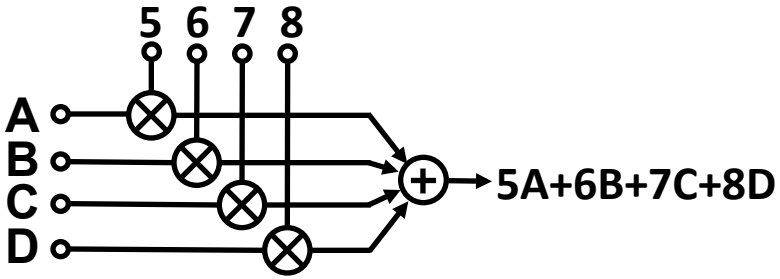
n x n matrix (e.g. n = 4)

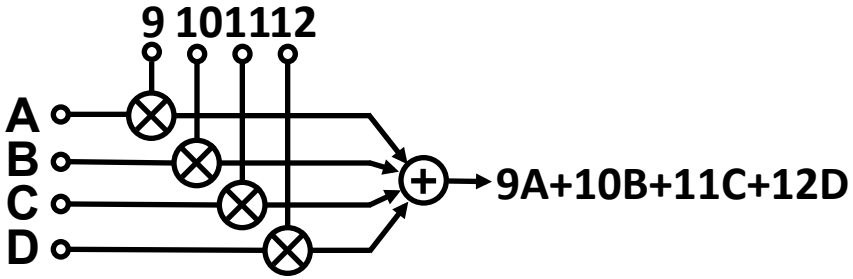
$$\begin{pmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \\ 9 & 10 & 11 & 12 \\ 13 & 14 & 15 & 16 \end{pmatrix} \begin{pmatrix} A \\ B \\ C \\ D \end{pmatrix} = \begin{pmatrix} 1A+2B+3C+4D \\ 5A+6B+7C+8D \\ 9A+10B+11C+12D \\ 13A+14B+15C+16D \end{pmatrix}$$

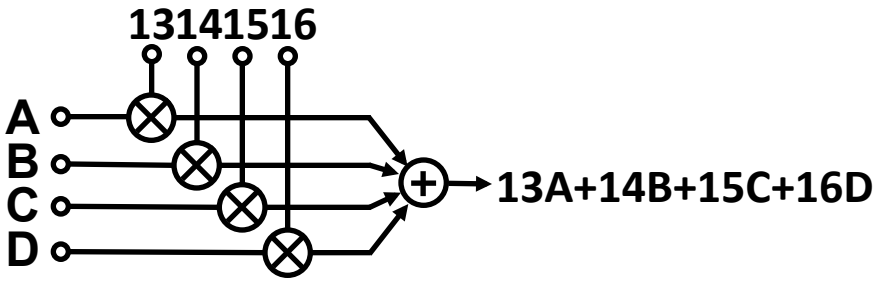






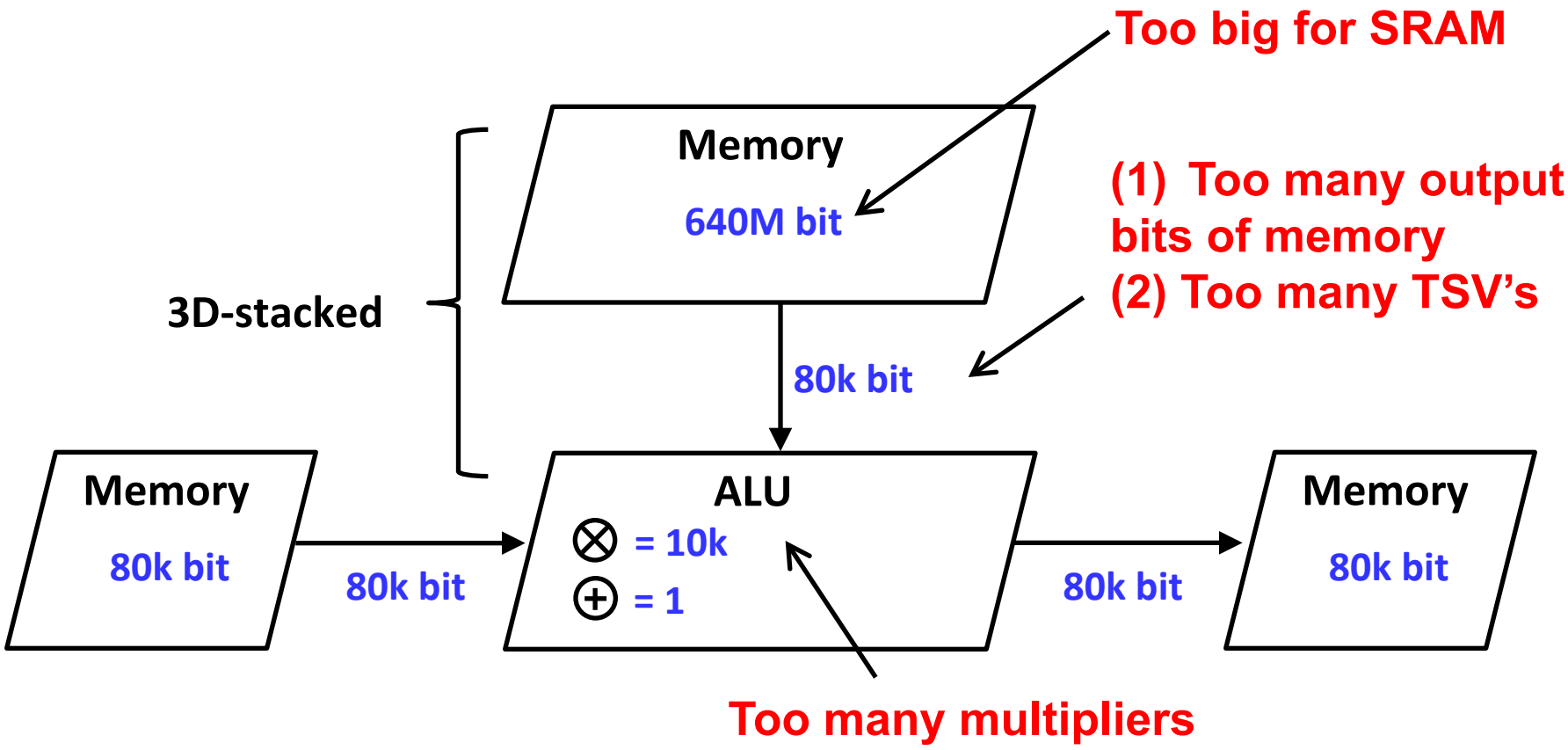




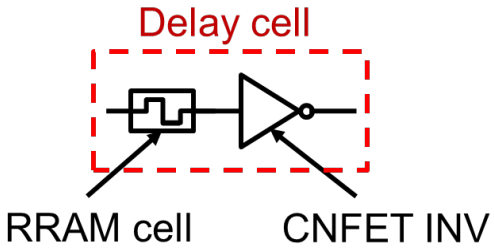


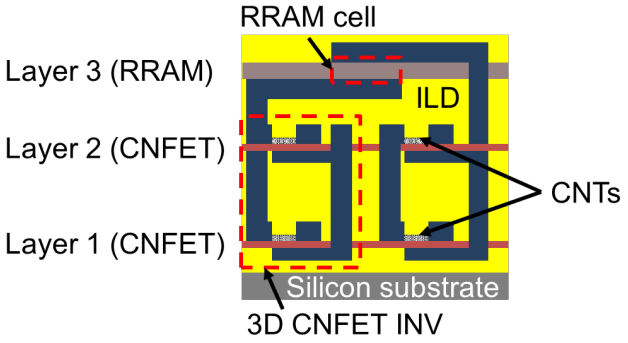
Hardware for Matrix Multiplication

n x n matrix ➔ 10000 x 10000 matrix
Each data = m-bit ➔ 8-bit (typical image)



Hyperdimension and Randomness





Random indexing: orthogonal transformation of data into hyper-dimensional space

Die	CNT density (CNT/µm)	Delay (µs)		Standard deviation (µs)	Std/(Mean-Min)
		Mean	Min		
1	1	0.73	0.21	0.18	86%
2	0.33	2.23	1.36	0.94	69%
3	0.11	6.79	4.82	2.41	50%

- CNT-RRAM combines helps to spread distributions
- 3D integration enables scalability
- Extremely low energy operation

[Collaborative Project with P. Wong and S. Mitra, Stanford]

Conclusion

- The need for defect-tolerant hardware due to noisy nanometer-scale devices is rising
- Large class of the next generation of applications could be categorized into recognition, mining and synthesis (RMS)
- Deploying heterogeneous systems will become inevitable to meet the specs (area, power, etc.)
- Emulating the algorithms with FPGAs and GPUs is ongoing before having these new devices in massive numbers