NYU

Lecture 13:
New Computation Paradigms

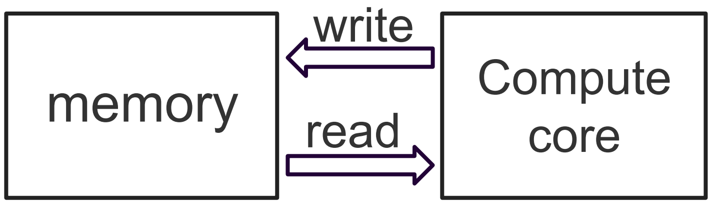
Recap

- FovealNet: Advancing AI-Driven Gaze Tracking Solutions for Efficient Foveated Rendering in Virtual Reality
- FovealSeg: Efficient Gaze-driven Instance Segmentation for Augmented Reality

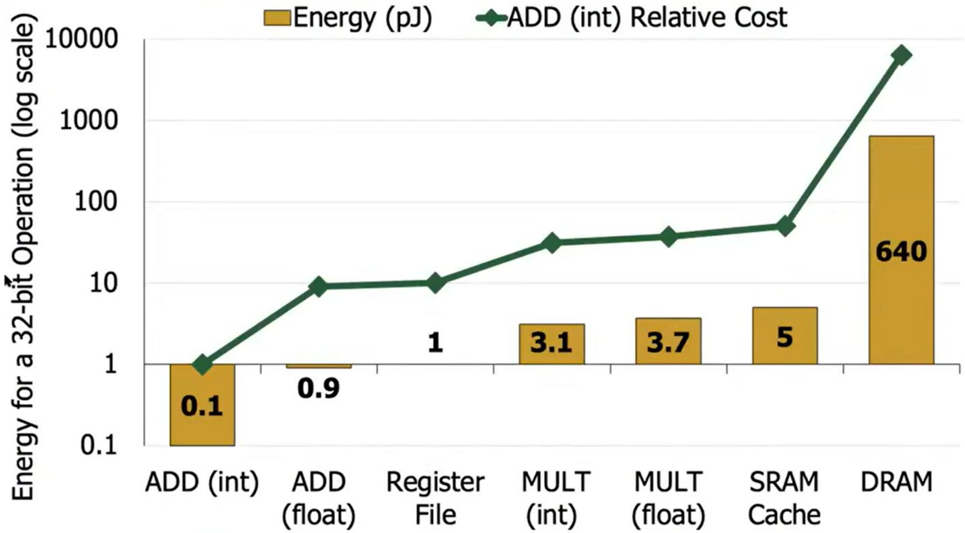
Topics

- In-memory computing
- Stochastic computing

Data Movement Cost vs. Computation Cost



- Retrieving a single element from memory is more costly than computing it.

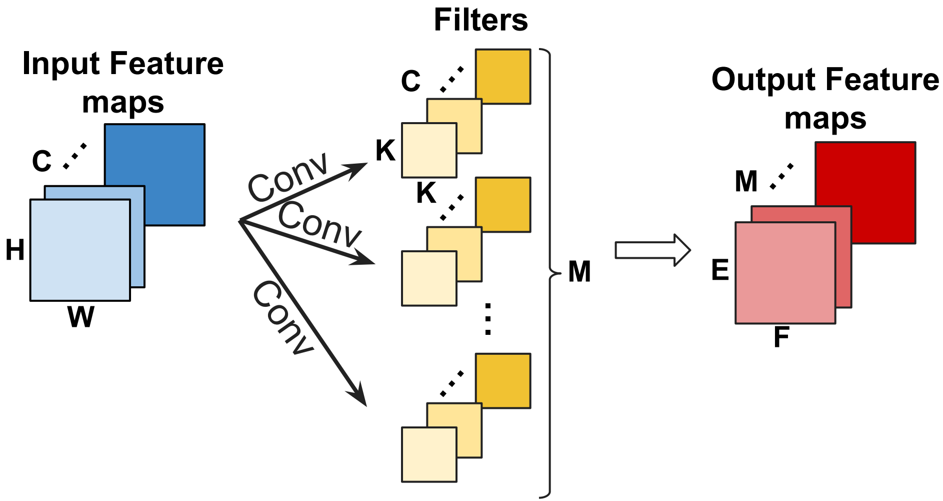


Onur Mutlu, "ISCA 2023 Tutorial: Real-world Processing-in-Memory Systems for Modern Workloads."

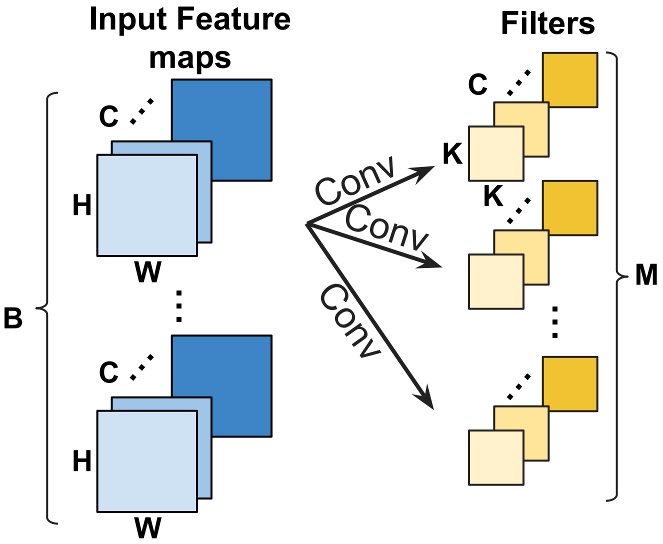
Data Movement Cost vs. Computation Cost

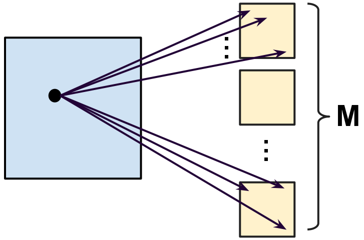
- **Arithmetic intensity**: the ratio of total floating-point operations to total data movement (bytes)

$$\frac{\text{Total FLOPs}}{\text{Total data movement}}$$



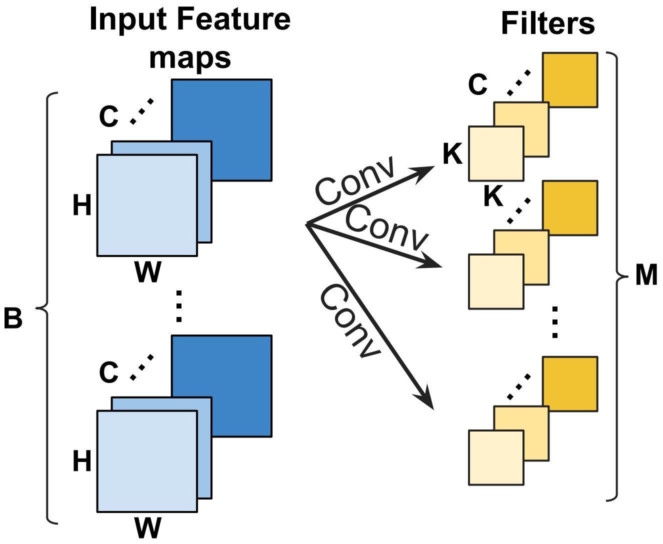
Data Movement Cost vs. Computation Cost

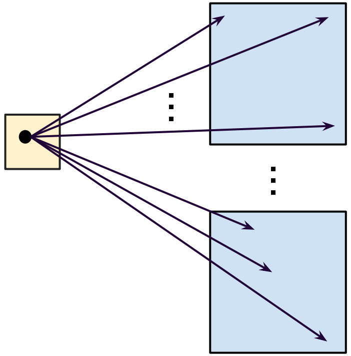




- For each single element within the input feature maps, the maximum amount of reuse = K^2M.

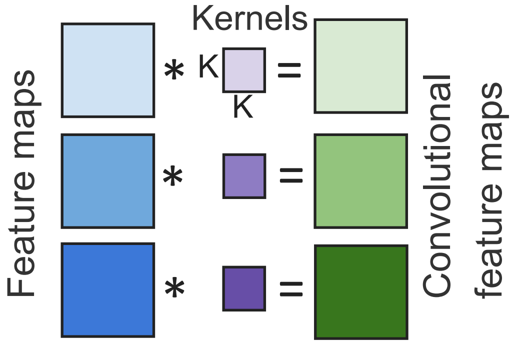
Data Movement Cost vs. Computation Cost



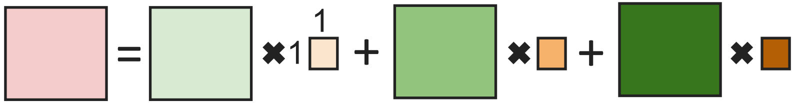


- For each single element within the weight kernel, the maximum amount of reuse = BHW.
- For standard convolution, the arithmetic intensity is high.

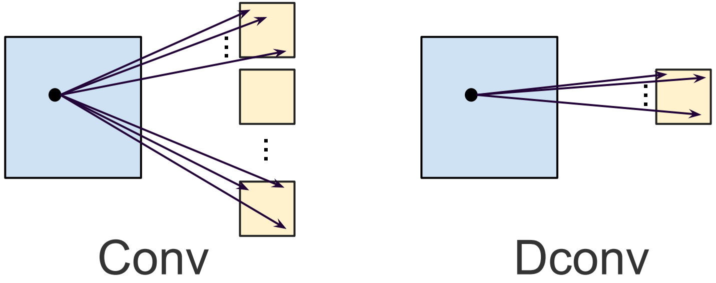
Data Movement Cost vs. Computation Cost



Step 1 Depthwise Convolution

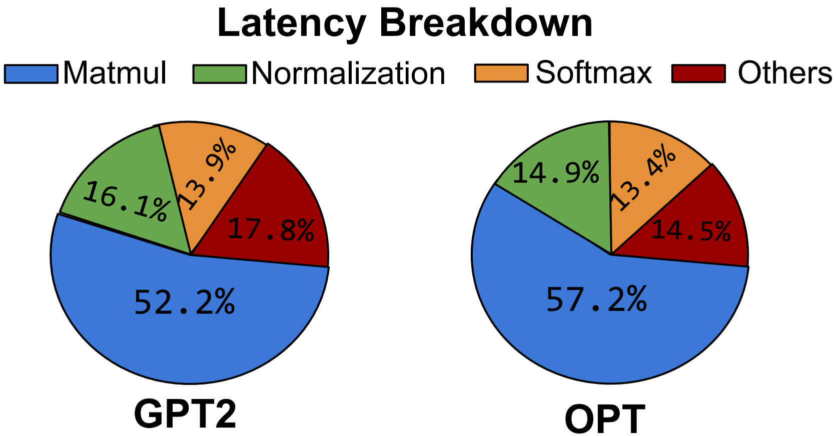


Step 2 Pointwise Convolution



- For each single element within the input activation, the maximum amount of reuse = K^2 for Dconv.

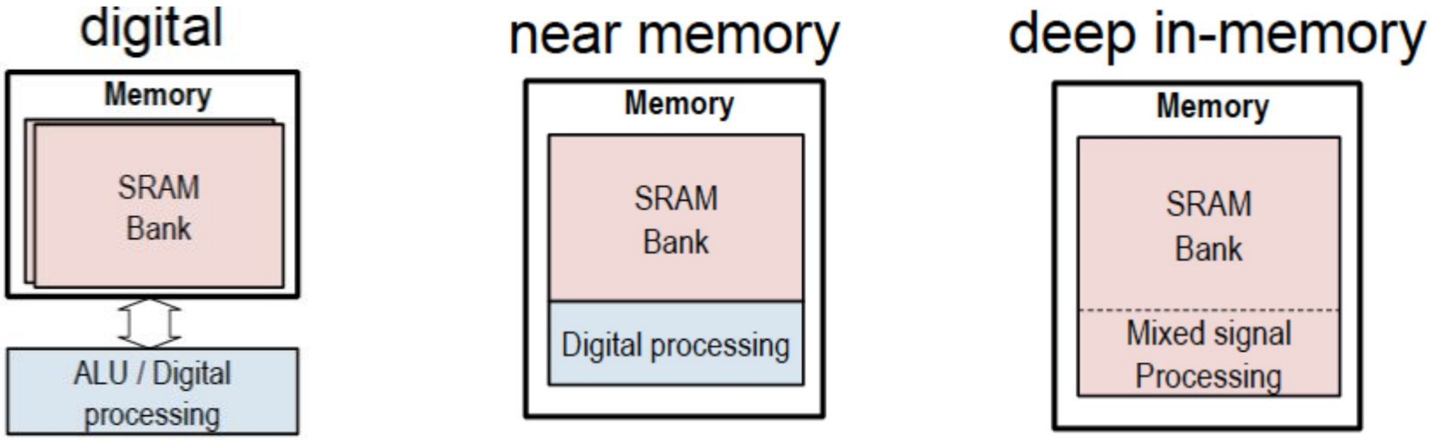
Breakdown on Computational Cost



- Matmul still contributes to majority of the overall latency.
- Nonlinear operations are not negligible.
- Also other operations (e.g., transposition, reshape) also contributes to a great portion of the overall latency.

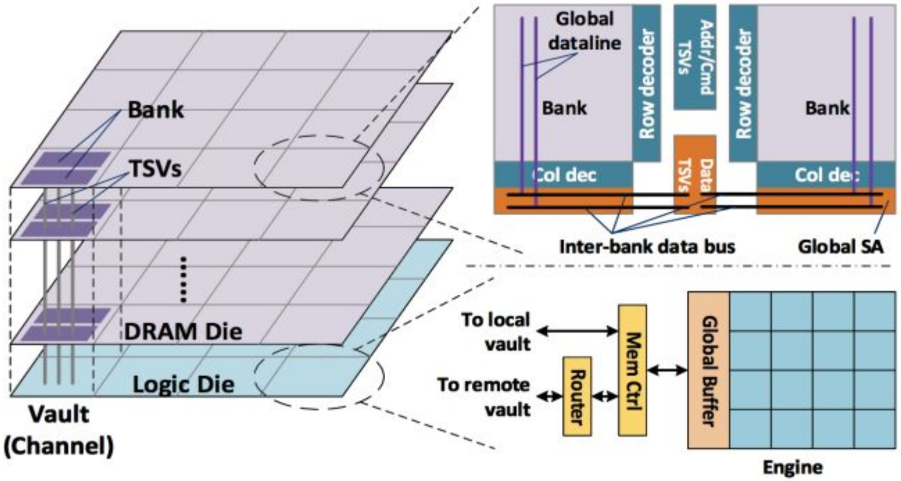
NYU SAI LAB

Near/In-Memory Processing

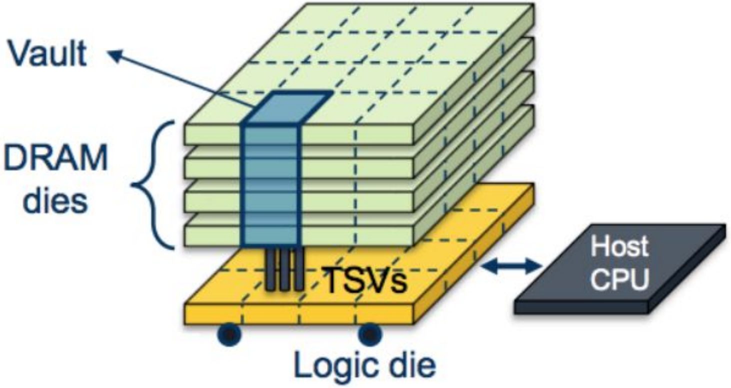


- Near memory computing has a higher BW, and analog in-memory computing integrate the computation with the memory access.
- Analog PIM brings compute closer to the memory.

Near Memory Processing



Eyeriss design



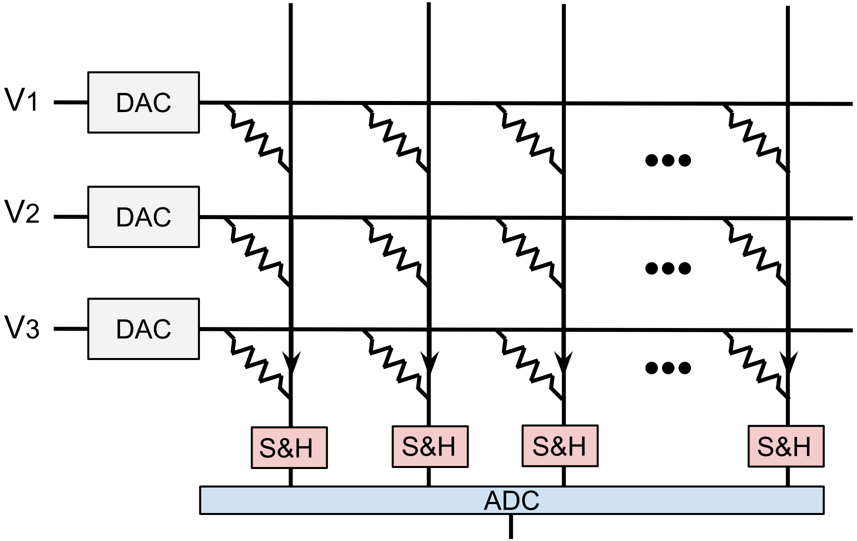
Tetris

NeuroCube

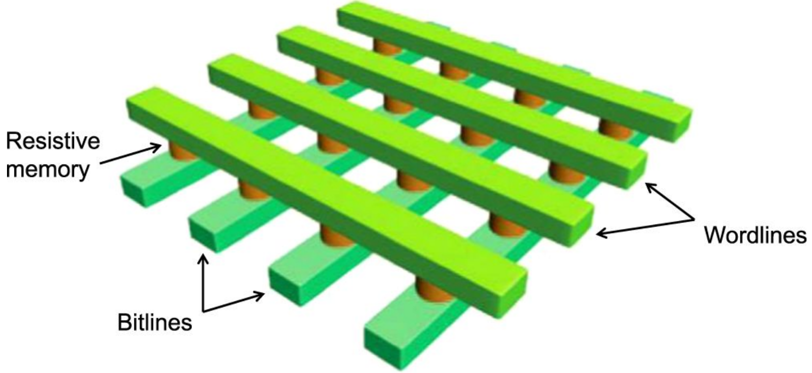
NYU SAI LAB

Gao, Mingyu, et al. "Tetris: Scalable and efficient neural network acceleration with 3d memory." *Proceedings of the Twenty-Second International Conference on Architectural Support for Programming Languages and Operating Systems*. 2017.

Resistive Memory



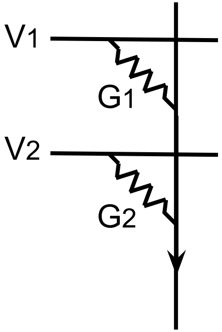
- Resistive RAM (ReRAM or RRAM) is a type of non-volatile RAM that works by changing the resistance across a dielectric solid-state material, often referred to as a memristor.

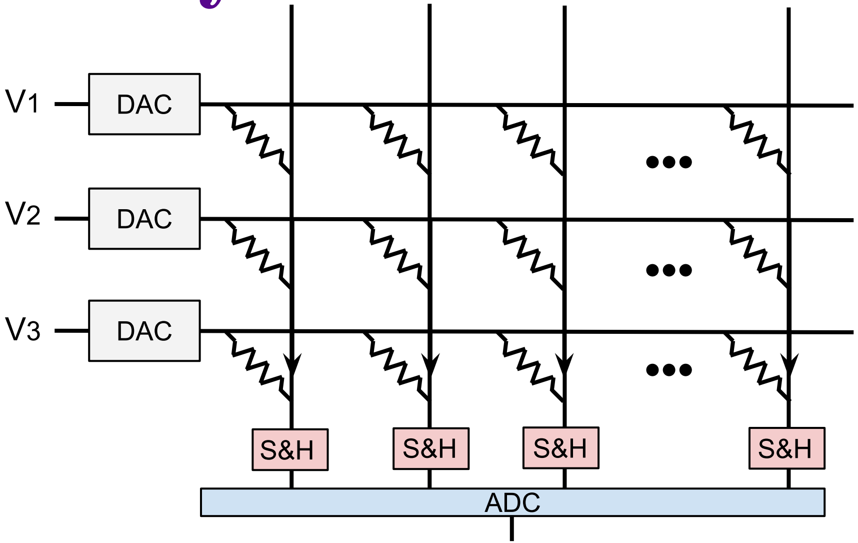


NYU SAI LAB

Processing In Memory

$I = V_1G_1+V_2G_2$

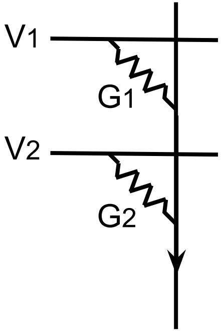


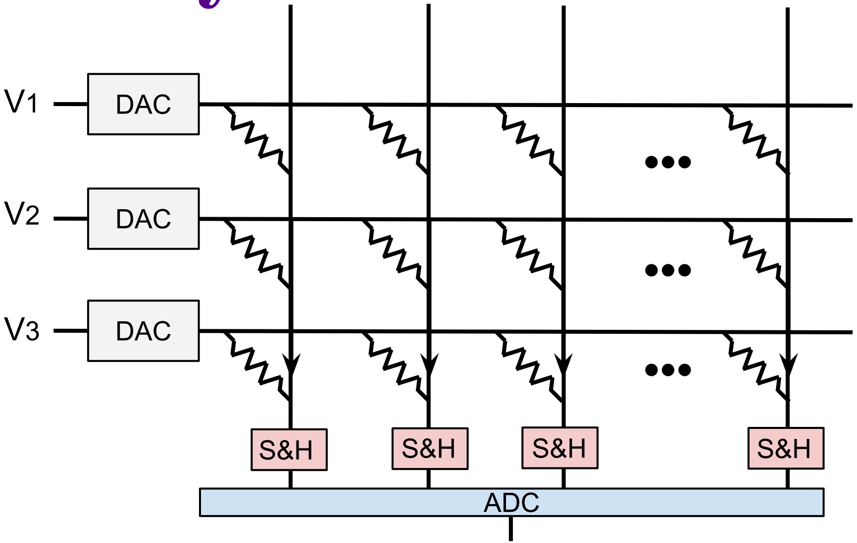


- The digital input are first passed to the DAC and converted to the analog input voltages.
- The voltages are applied to each of the rows in the crossbar array.

Processing In Memory

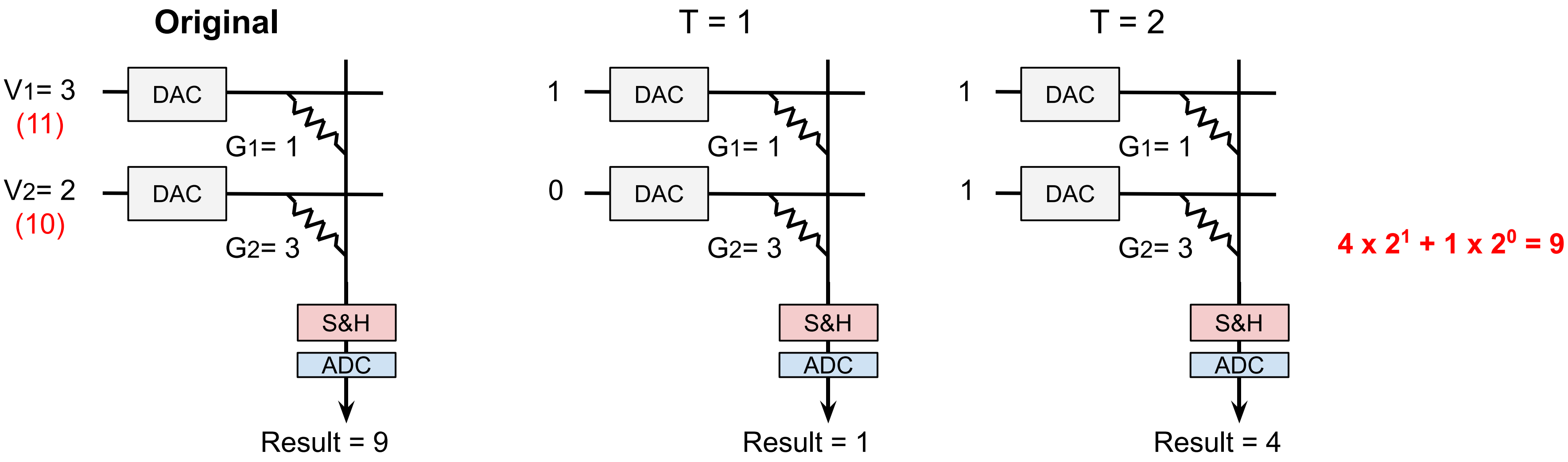
$I = V_1G_1+V_2G_2$





- The output current accumulated at the bottom of each column is the dot product between the voltages and the conductances across the rows.
- A sample-and-hold (S&H) circuit receives the bitline current and feeds it to a shared ADC unit

Processing In Memory

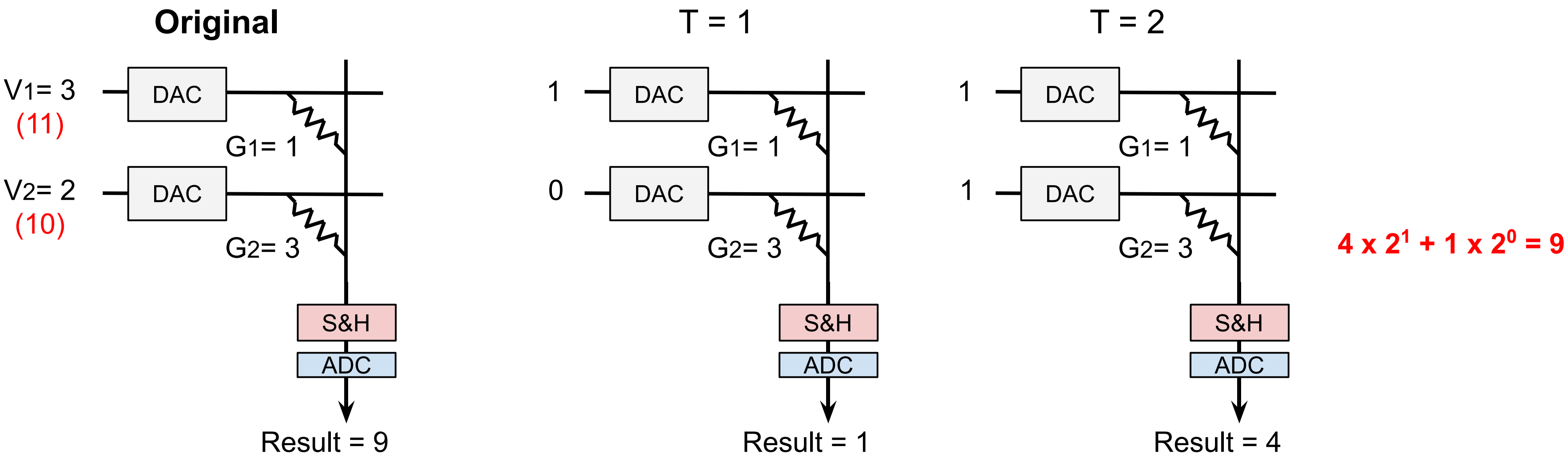


$4 \times 2^1 + 1 \times 2^0 = 9$

- Assume both inputs and weights are 16 bits, we need a 16-bit DAC to provide input voltage, 2^{16} resistance levels in each cell, and an ADC which can handle over 16 bits, which leads to a significant overhead.

NYU SAI LAB

Processing In Memory

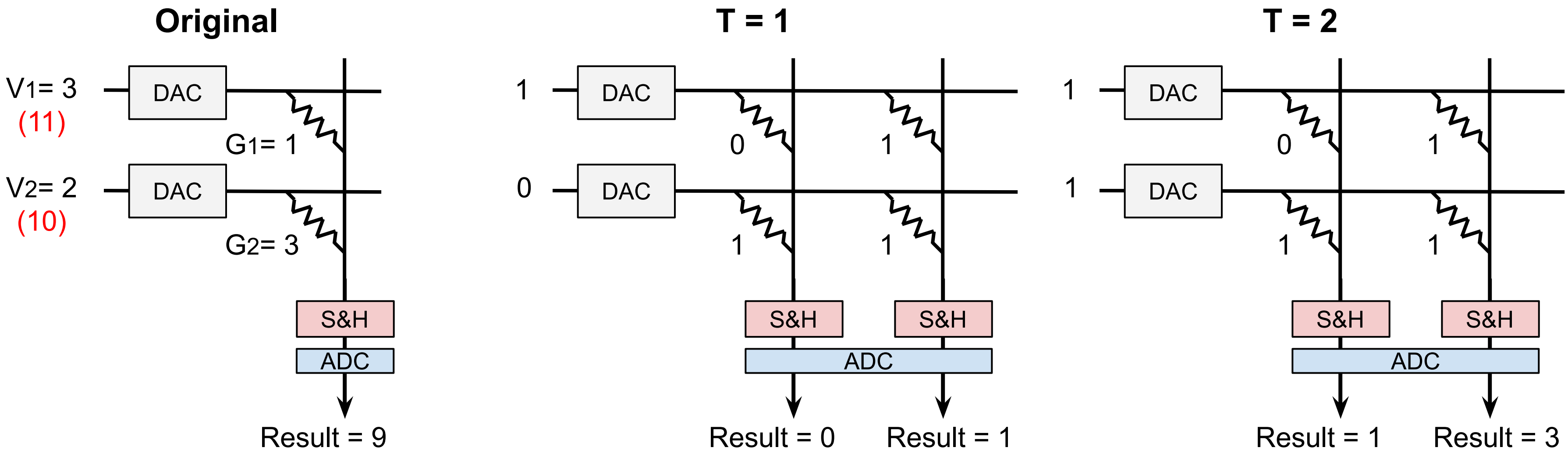


$4 \times 2^1 + 1 \times 2^0 = 9$

- Instead, the digital input enters the crossbar in a bit-serial manner, the intermediate results are buffered in the register. Shift-Add operation is them performed after all the input bits entering the crossbar.

NYU SAI LAB

Processing In Memory



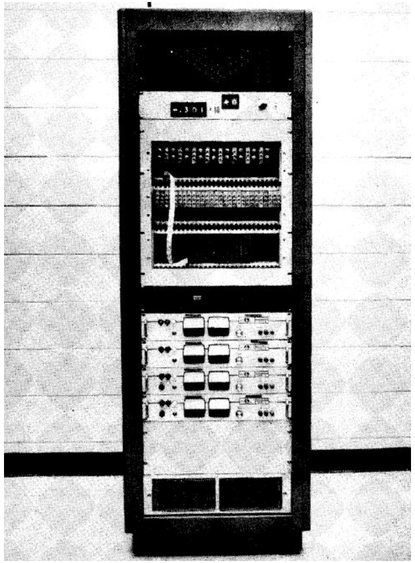
$1 \times 2^0 + 0 \times 2^1 + 1 \times 2^1 + 3 \times 2^1 = 9$

Topics

- Processing in memory
- Stochastic Computing

Stochastic Computing

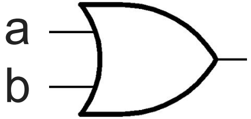
- Stochastic computing is a computational approach that utilizes random bit streams to perform numerical calculations, offering benefits in power efficiency and hardware simplicity, particularly for error-tolerant applications.
- Introduced by John von Neumann in 1953.



The RASCEL stochastic computer, circa 1969

Stochastic Computing

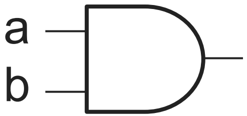
- $a = 0.5$, $b = 0.5$
  - ○ $a = 00111100$ $p_a(1) = 0.5$
  - ○ $b = 11000011$ $p_b(1) = 0.5$



11111111 = c

$p_c(1) = 1$

- $a = 0.5$, $b = 0.5$
  - ○ $a = 11001010$ $p_a(1) = 0.5$
  - ○ $b = 01010011$ $p_b(1) = 0.5$



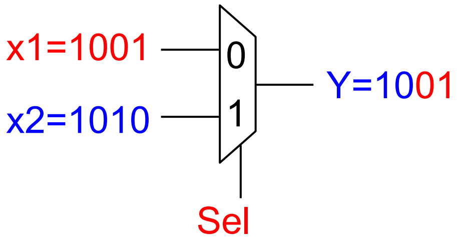
01000010 = c

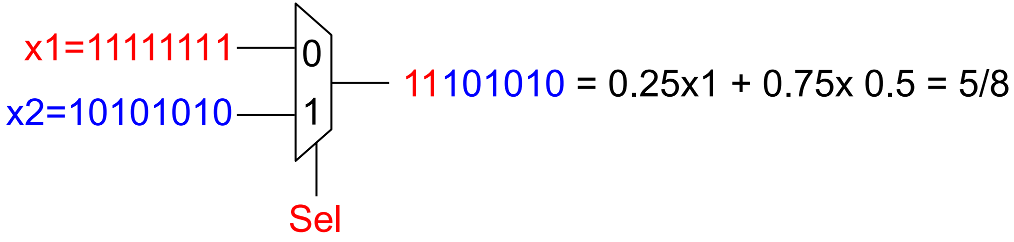
$p_c(1) = 0.25$

- As the input stream lengthens, the multiplication process will become more accurate.

Addition with Stochastic Computing

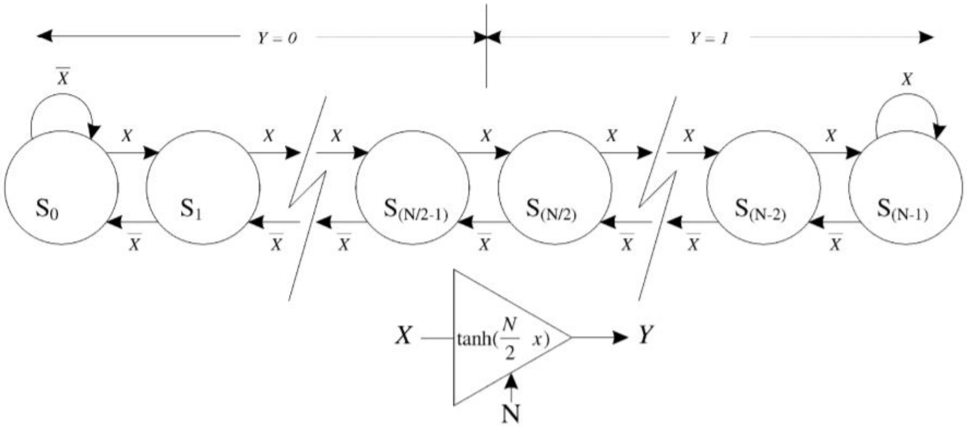
- MUX implementation
  - By adjusting Sel over time, the output of the multiplexer will equal to the weighted sum of the input bit streams.
  - The accuracy gets worse when the number of inputs to the MUX is large.





Nonlinear Operation with Stochastic Computing

- The tanh function is highly suitable for SC-based implementations because i) it can be easily implemented with a K-state finite state machine (FSM) in the SC domain.



- The major advantage of stochastic computing is the significantly lower hardware cost for a large category of arithmetic calculations.