Analog Compute-near-RRAM CNN Accelerator for Fast-Inference Applications

Aya G. Amer, Minghan Chao, Gage Hills,
Anantha P. Chandrakasan, Max Shulaker

May 1st , 2024

Energy
Efficient
Circuits and
Systems
@ MIT

Outline

- **Background and Motivation**
- **Proposed NMC CNN Accelerator**
  - ➢ Proposed Fully-Parallel Architecture
  - ➢ Proposed RRAM Cell and Pulsed RRAM Read
  - ➢ Proposed multi-bit MAC Cell
- **Proposed Analog Computations**
  - ➢ CONV, FC, POOL layers Implementation
  - ➢ Low-Power Analog Circuits
- **Experimental Results**
- **Conclusions**

Motivation: AI Applications & Edge Computing

Security & Surveillance

Smart TV

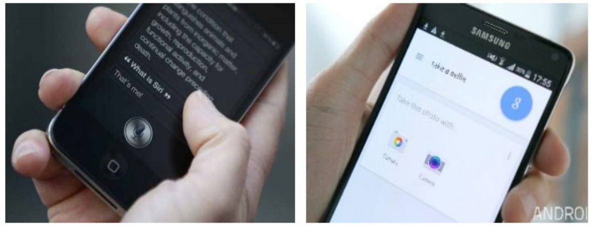
Autonomous Cars

Artificial Intelligence at home
Open Source Code Robot & Linux OS
Qbo

Home Robotics

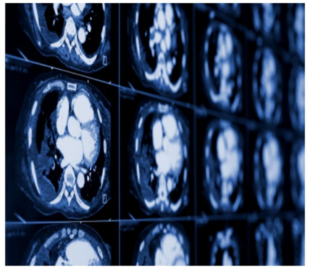
VR & Gaming

Face recognition



Speech Recognition

Wearable Medical Devices



Medical Imaging

- Low Latency ✓
- Low Cost ✓
- Security ✓

Motivation: Computing Limitations

Huge amounts of data processing in real-time.

~ 6 GB of data every 30 seconds

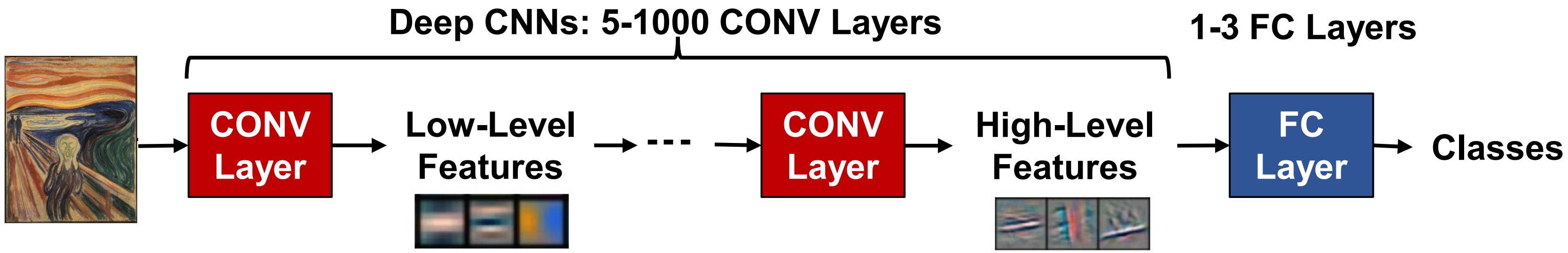
Prototypes use ~ **2500 Watts** !!

~ 60 k searches every second

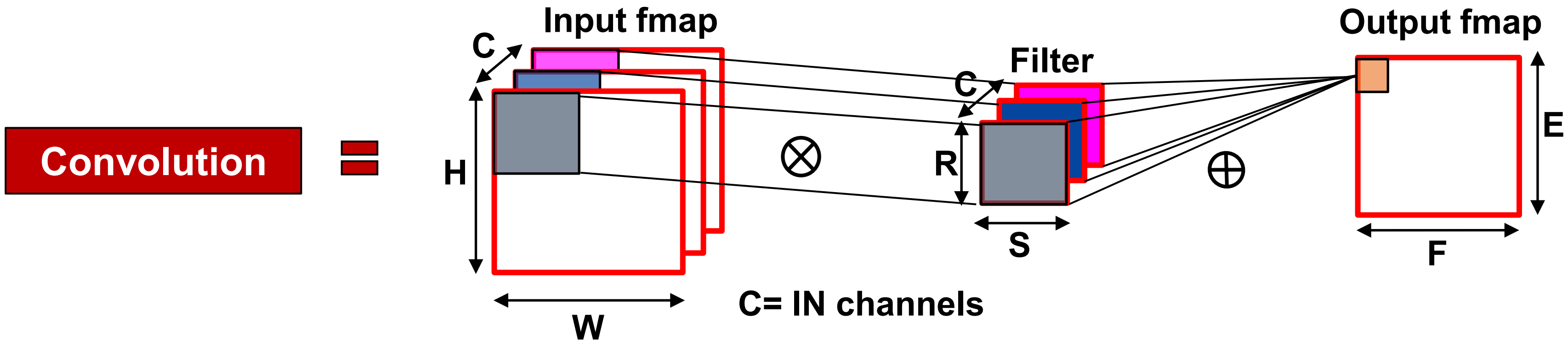
1 search = Energy of 60W light bulb for 17 seconds !!

Convolutional Neural Networks: CNNs

- CNNs provide state-of-the-art accuracy for wide range of tasks.



- CNNs basic operation is dot product or multiply and accumulate (MAC).



CNNs: Major Challenges

- **Computation Intensive**
  - Millions of MAC operations
  - Energy Intensive

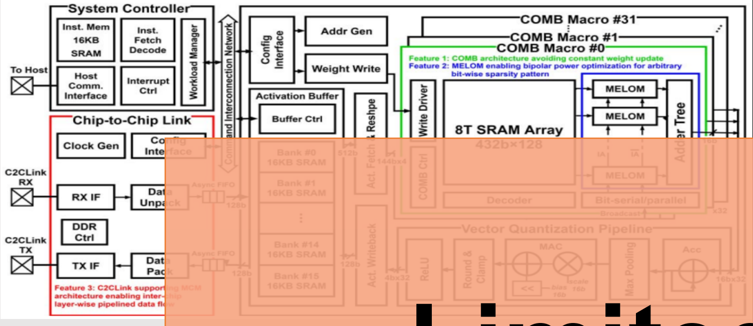
- **Data movement to/from memory**
  - Massive continuous data movement between computation and memory
  - Substantial energy and speed penalty

- **Hardware Reuse**
  - Long latency
  - High energy

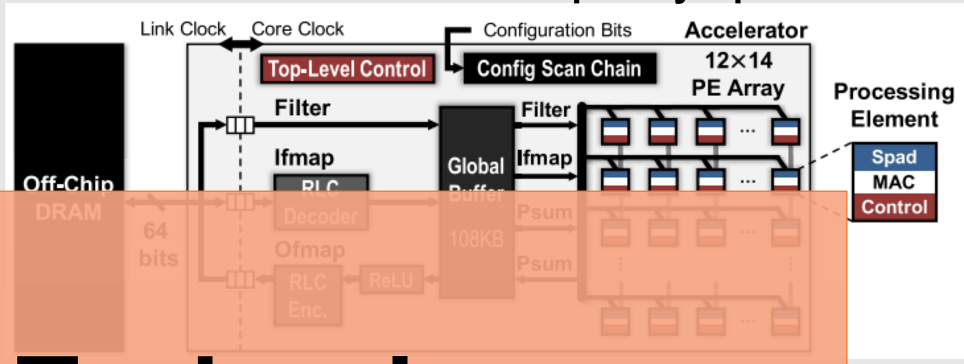
How to Improve Computations?

Improved Circuits, System Architectures

COMB-MCM, ISSCC’22
Scalable Multi-Chiplet-Modules + Sparsity Optimization

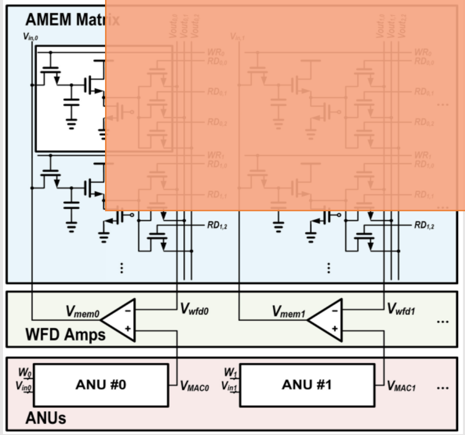


Eyeriss, JSSC’17
Data Flow + Data Statistics + Sparsity Optimization

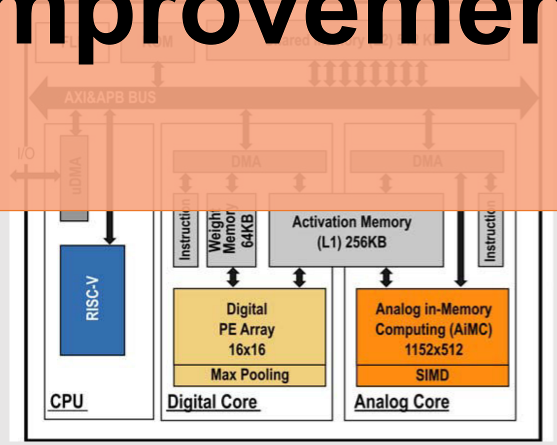


Limited by Technology Improvement

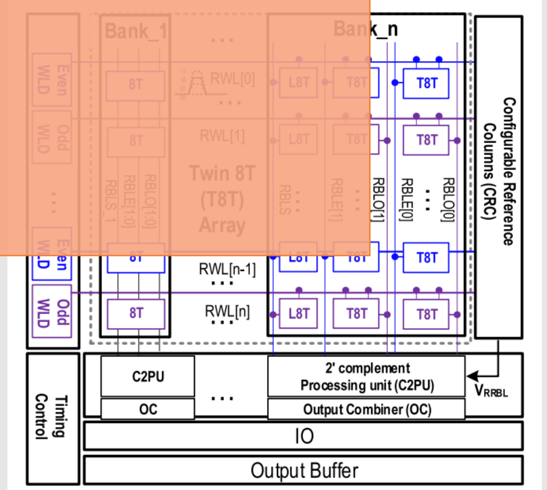
ARCHON, ISSCC’22
Analog CNN + Analog Memory



DIANA, ISSCC’22
Multi-Core ML (Analog + Digital)

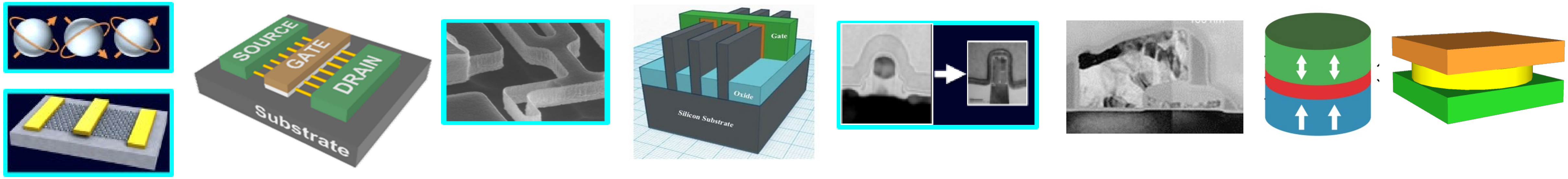


T8T SRAM IMC, ISSCC’19

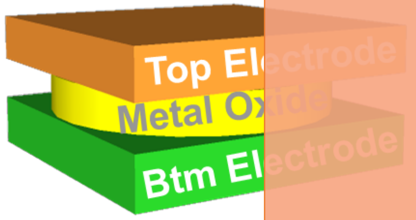


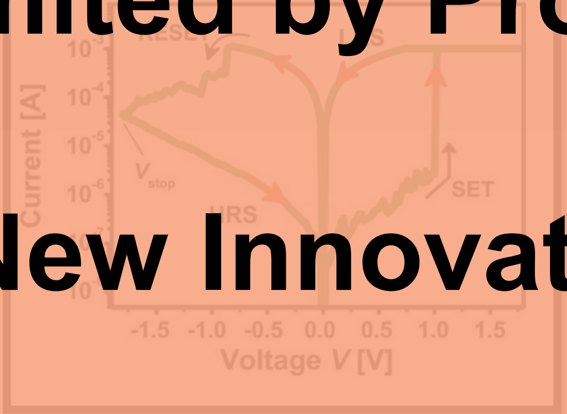
How to Improve Computations?

Enhanced / New Devices



NV- RRAM Technology





- Dense
- Reduced Leakage
- Improved Read/Write Speed.
- Compatible with CMOS process.
- Energy-Efficient storage and computing inside memory.

Limited by Process Variations

⬇

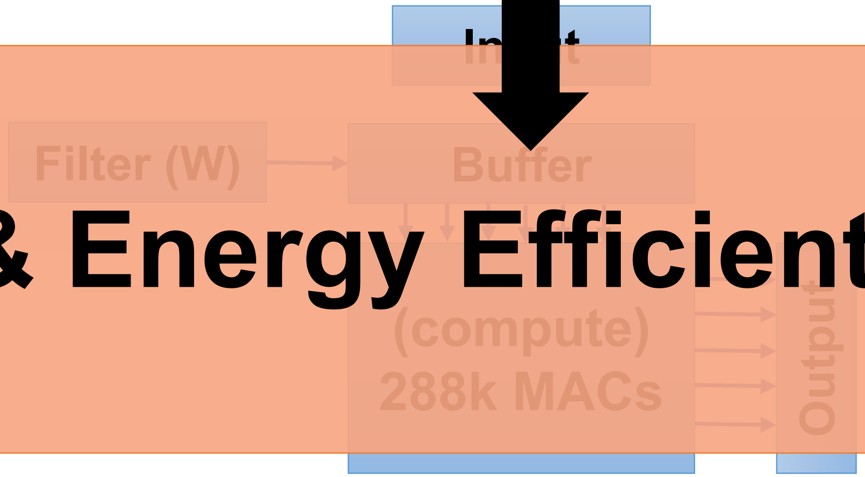
New Innovations Required

Proposed Fully Parallel CNN Hardware

- **Architecture Parallelism**
  - Map all storage and computation hardware on chip to generate the outputs in 1 clock cycle
  - Remove bottleneck of data transfer to/from memory

Layer	Input size	# Filters	Filter size	Output size	Multiplications = Filter Size * Output size
CONV	16x16x128	1	3x3x128	16x16x1	288 k

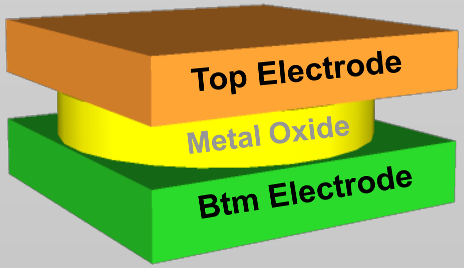
Area – Speed tradeoff



Dense & Energy Efficient MAC Cells

Proposed Solution

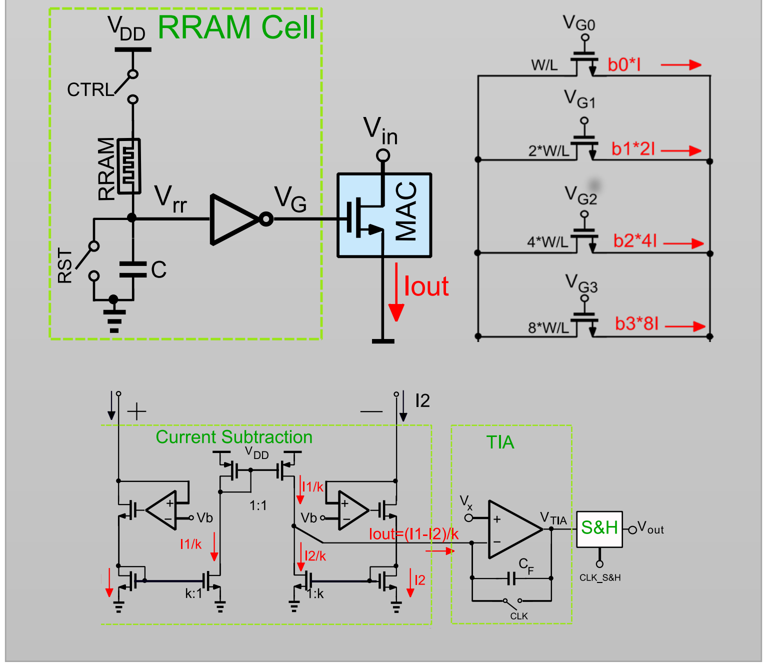
RRAM Technology



- Non-Volatile
- Dense
- Energy-Efficient
- Compatible with Si CMOS process

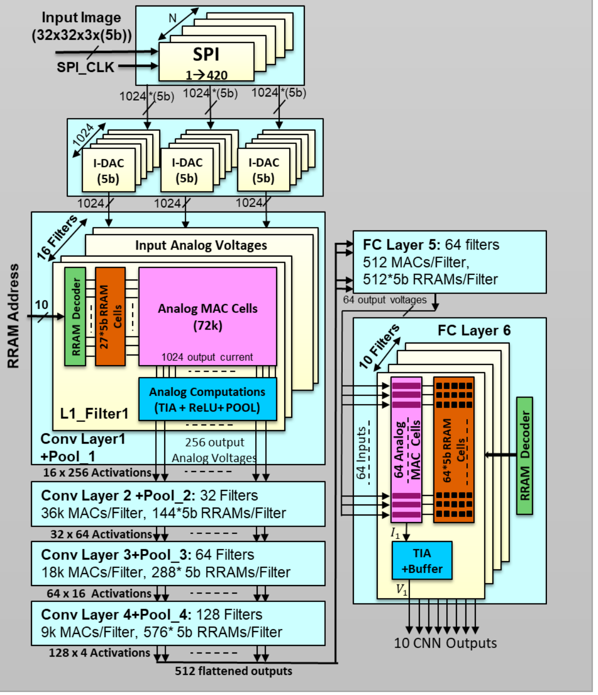
➕

Novel Analog Circuits



🟰

Parallel Architecture



Conv Layer 2 +Pool_2: 32 Filters
36k MACs/Filter, 144*5b RRAMs/Filter

Conv Layer 3+Pool_3: 64 Filters
18k MACs/Filter, 288* 5b RRAMs/Filter

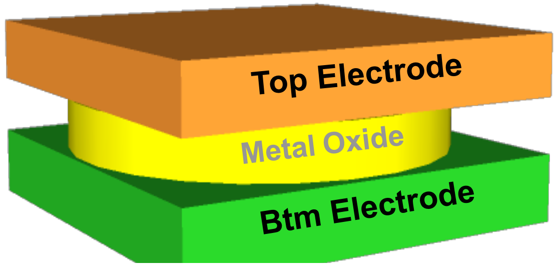
Conv Layer 4+Pool_4: 128 Filters
9k MACs/Filter, 576* 5b RRAMs/Filter

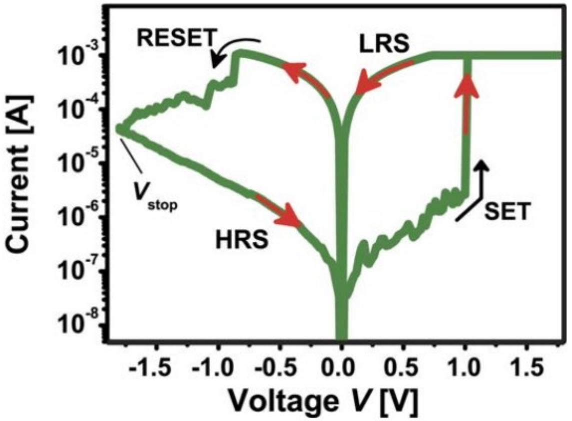
FC Layer 5: 64 filters
512 MACs/Filter,
512*5b RRAMs/Filter

> 300x EDP Benefit

New Device: Resistive RAM (RRAM)

- **Dense (< 60nm x 60nm)**
- **Non-volatile**
- **Programmability into one of 2 states**
  - High Resistance State (HRS > 100k)
  - Low Resistance State (LRS < 1k)

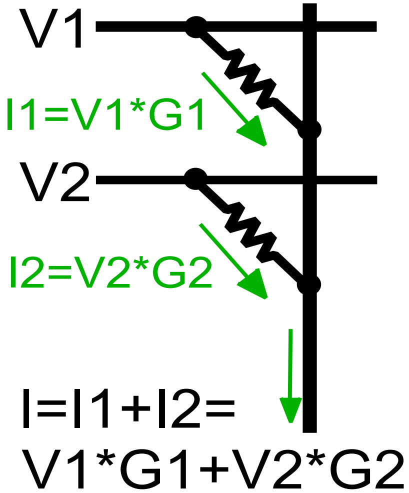




- **Digital memory device** (store CNN weights)
  - HRS → Low Conductance → Weight = 0
  - LRS → High Conductance → Weight = 1

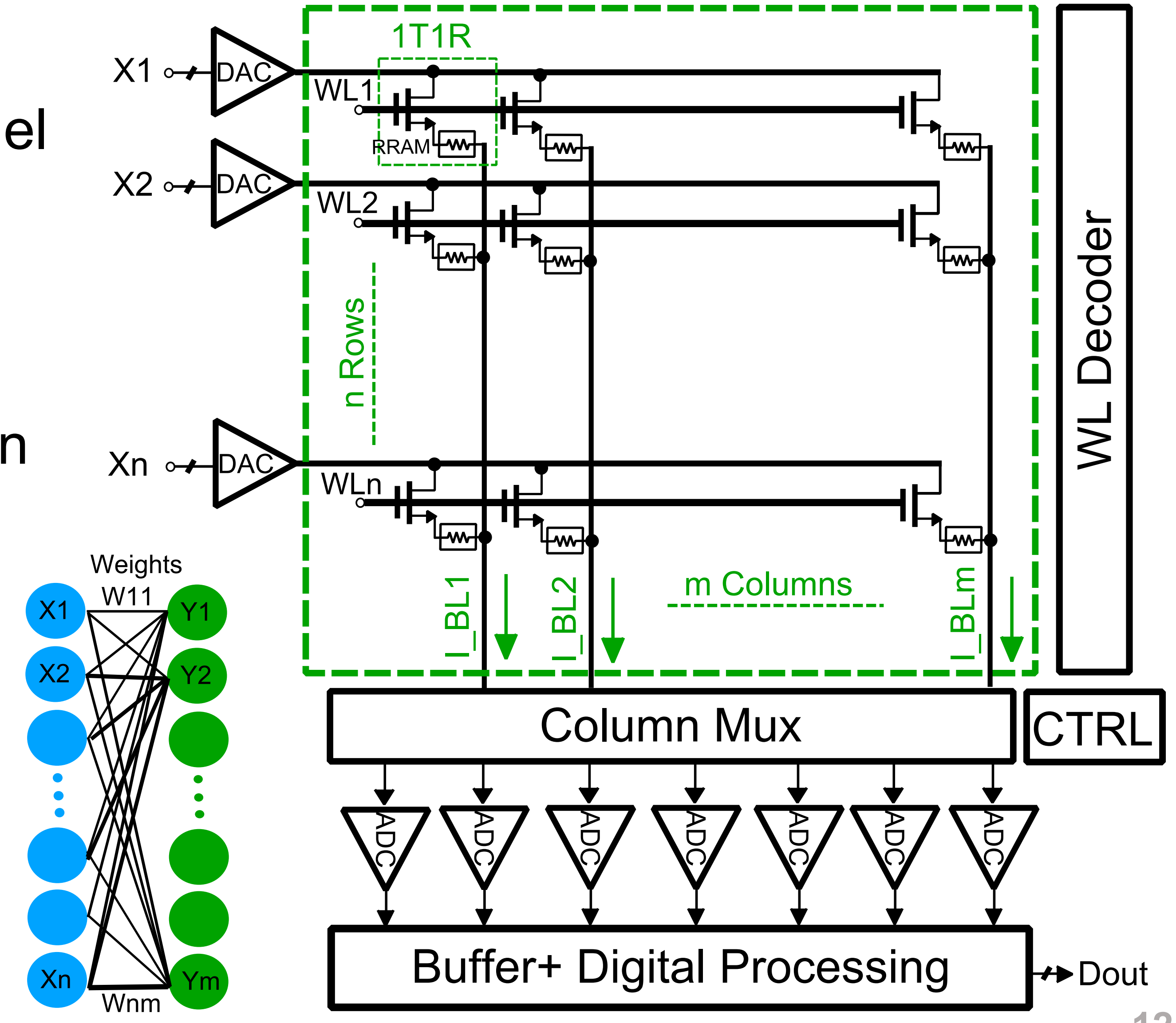
- **Perform MAC operations in parallel**

$$I = V_1 * G_1 + V_2 * G_2$$



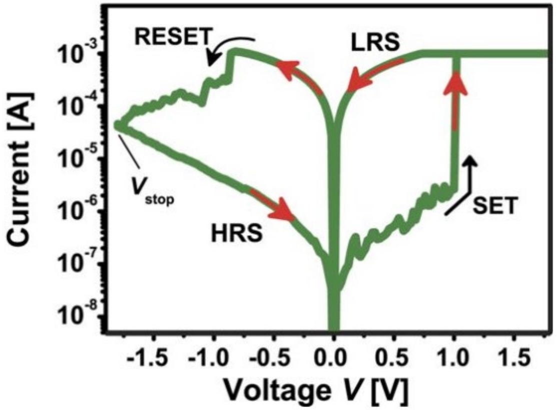
Computation in RRAMs (1T1R Configuration)

- **RRAM in crossbar array**
  - ➢ Perform MAC operations in parallel
  - ➢ Energy efficient
- **Access Transistors**
  - ➢ Address RRAM cells by activation of word- and bit-lines.
- **ADCs/DACs**
  - ➢ Interface array with buffers and digital processing.
  - ➢>80% of system power and area.



Major RRAM Challenges

- **I-V Non-linearity**
  - Limits overall accuracy
  - Difficult for multibit weight implementation

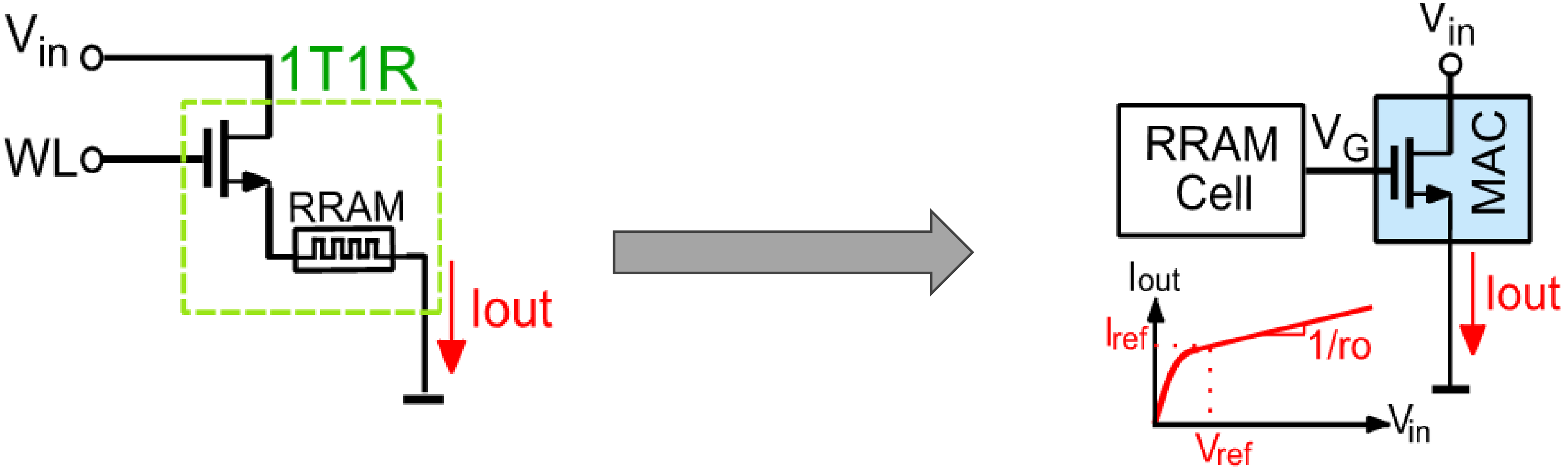


- **Resistance Variations**
  - LRS= 800Ω -2kΩ, HRS = 10kΩ -100kΩ
  - Finite Roff/Ron ratio (~10)

- **Limited Endurance**
  - < 10^6 set-reset cycles before permanent write failure

Proposed RRAM + MAC Cell

Conventional IMC 1T1R **Proposed NMC**



$I_{out}\,\alpha\,V_{in} * W$

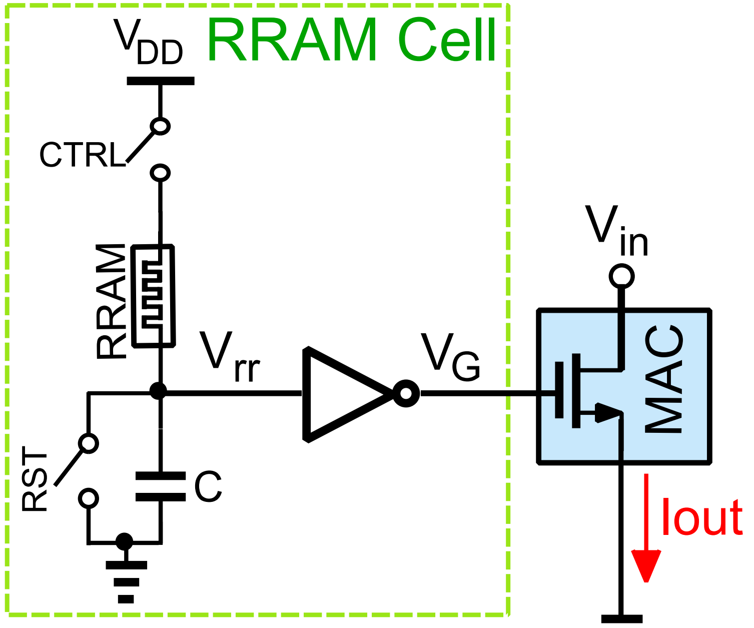
- **Multiplication with a transistor**
- **RRAM cell turns on/off MAC transistor.**

RRAM HRS (W=1) → $V_G = 1 \rightarrow I_{out}\,\alpha\,V_{in} * W$

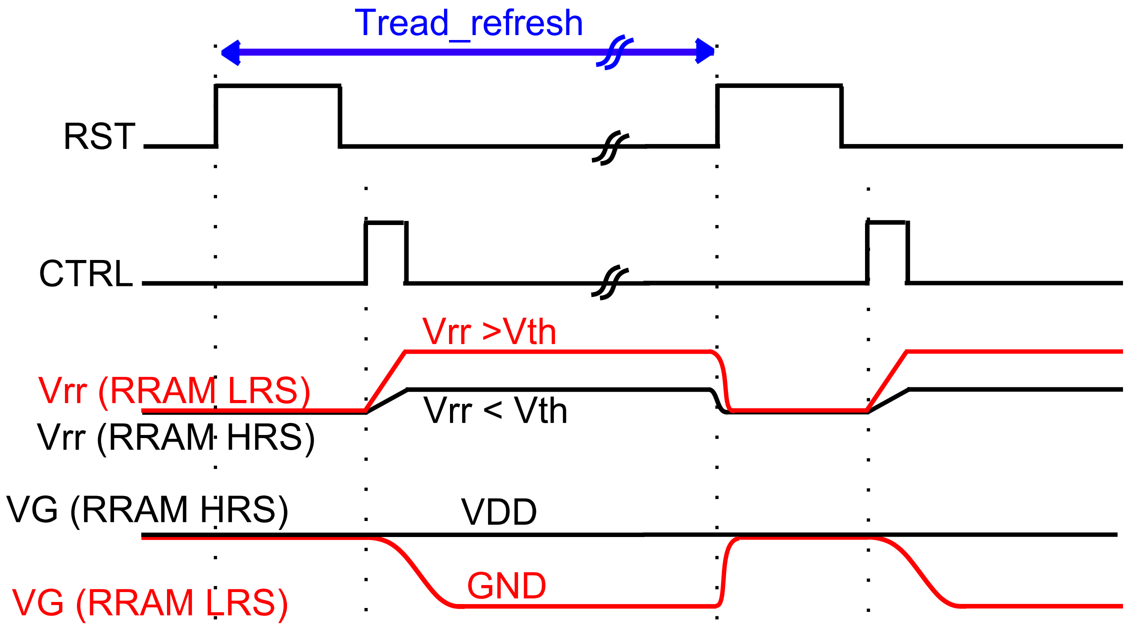
RRAM LRS (W=0) → $V_G = 0 \rightarrow I_{out} = 0$

Proposed RRAM Cell

- **RRAM cell turns on/off MAC transistor.**
  - ➢RRAM Cell Output = VDD/0 depending on the RRAM State (HRS/ LRS)

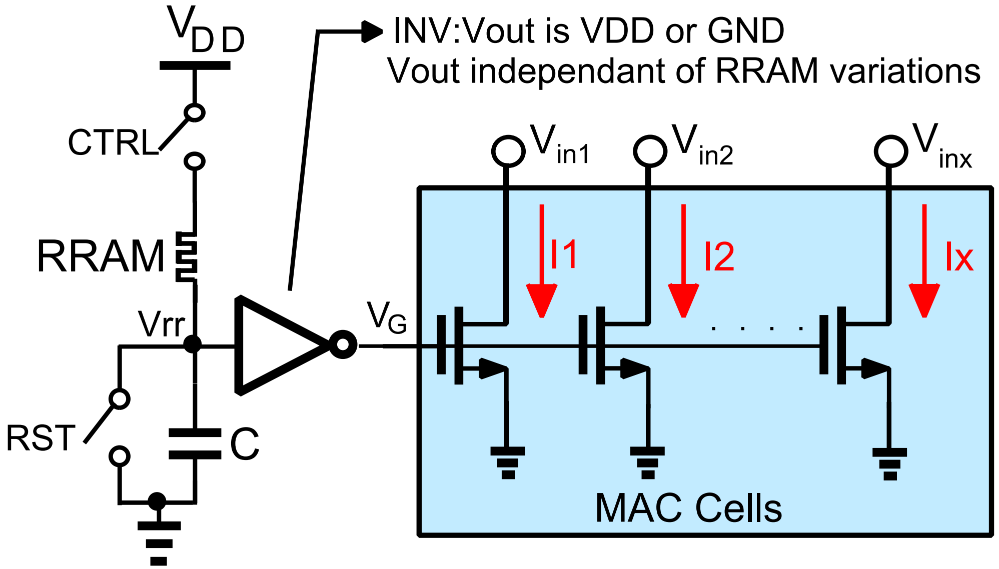


Switches + Capacitor + Inverter



Duty-Cycled Read
→ Low Power Consumption

Proposed Pulsed RRAM Read



- **Read once at start of the operation**

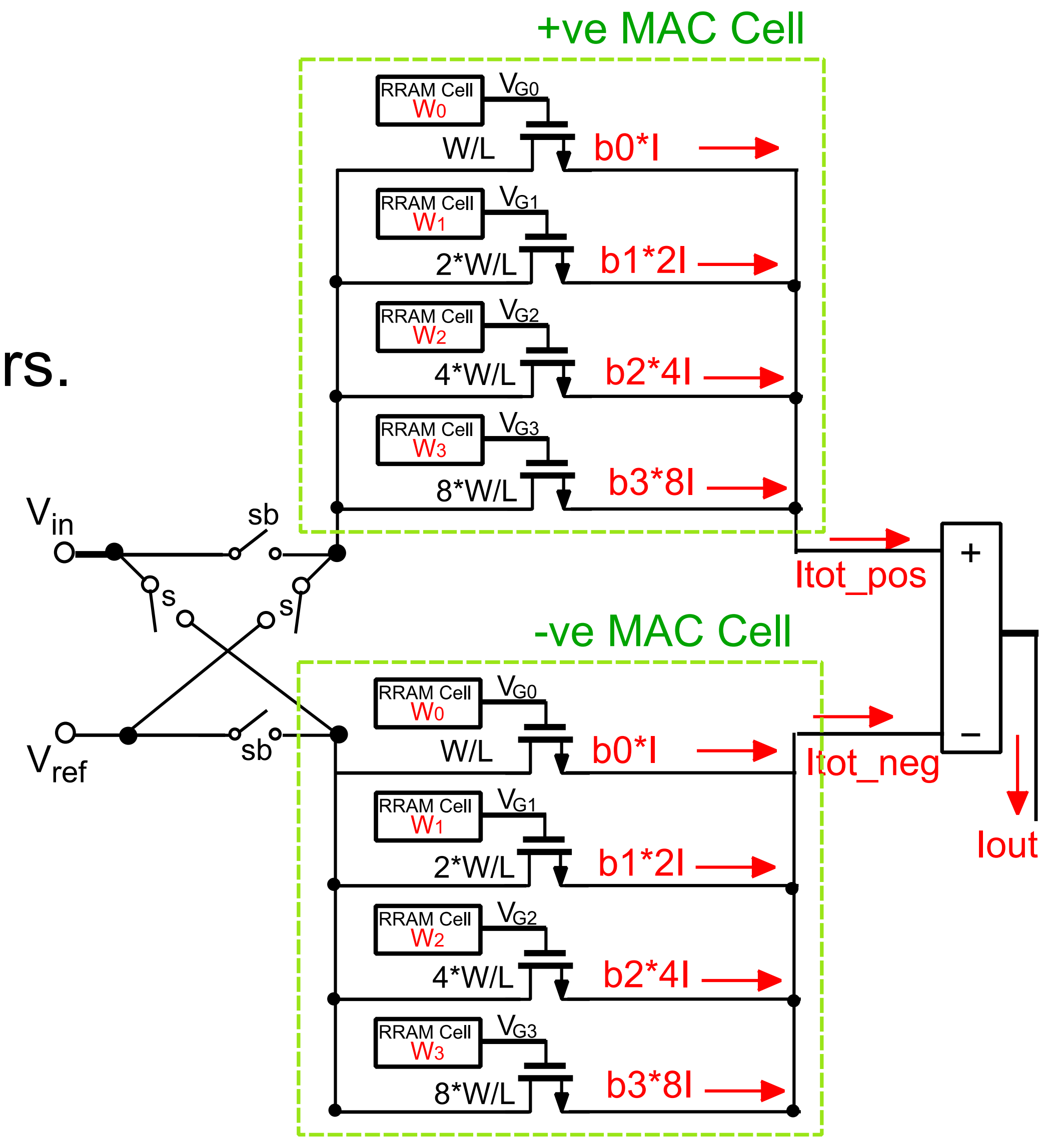
  No static power and leakage power.

- **RRAM Cell output can be shared with many MAC cells.**
  - ➢ Reduce number of RRAM cells on chips.

- **RRAM Cell output independent of RRAM variations**
  - ➢ Works with 50% RRAM variations and Roff/Ron down to 5

Analog Multiplication by n-bit Signed Weight

- **Differential MAC cells + Current Subtraction**
  - ➢ The sign bit connects Vin to either the positive or negative MAC cell.
  - ➢ Weights control both MAC cells transistors.
  - ➢ Removes offset currents.



- **Binary-Weighted Current DAC**
  - ➢ Scale MAC transistors W/L

- **Low-Power MAC**
  - ➢ Power scales with input/weight values.
  - ➢ W=0 → $I_{out} = 0$

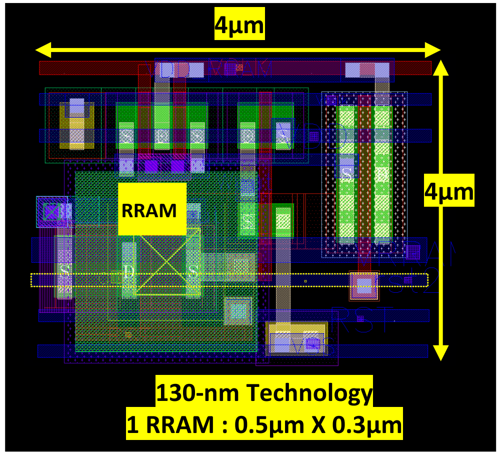
Proposed RRAM + MAC Cells

- **Linear Operation**
  - Improves overall MAC accuracy.

- **Robust against Variations**
  - Works with 50% RRAM variations.
  - Works with finite Roff/Ron ratio (>5).

- **Low Power**
  - Duty-cycled RRAM read operation
  - MAC power scales down with IN/W sparsity.

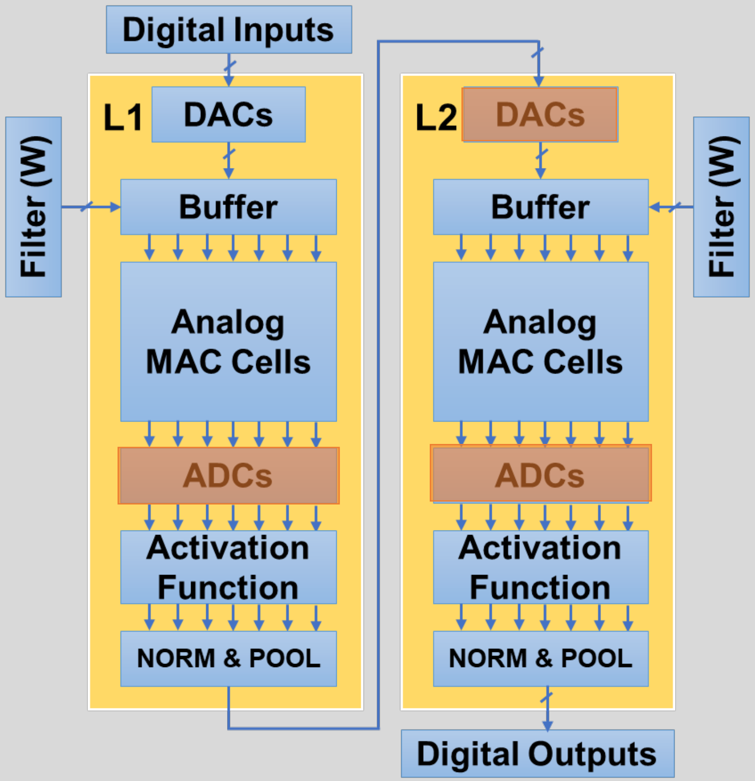
- **Small Area**
  - Dense → implement large number on-chip.
  - RRAM cell output is shared with many MAC cells.



130-nm Technology
1 RRAM : 0.5µm X 0.3µm

Analog Computations

Conventional Analog-Mixed CNN Computations



Digital Inputs

L1

DACs

Buffer

Filter (W)

Analog MAC Cells

ADCs

Activation Function

NORM & POOL

L2

DACs

Buffer

Filter (W)

Analog MAC Cells

ADCs

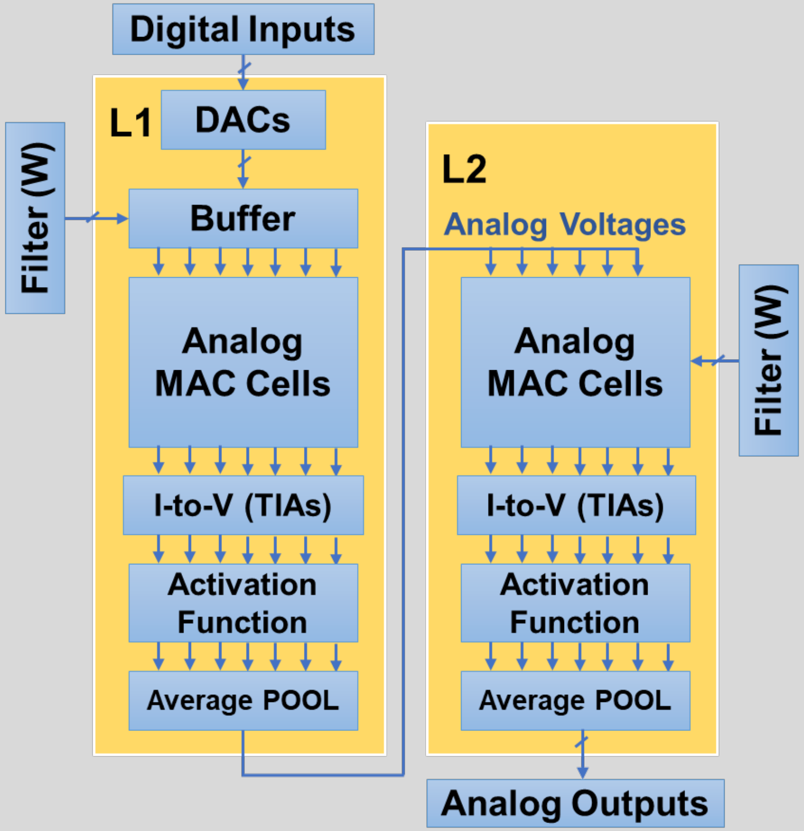
Activation Function

NORM & POOL

Digital Outputs

ADCs/DACs
→ Energy, Delay, Area Overhead

Proposed Analog CNN Computations



Digital Inputs

L1

DACs

Buffer

Filter (W)

Analog MAC Cells

I-to-V (TIAs)

Activation Function

Average POOL

L2

Analog Voltages

Analog MAC Cells

Filter (W)

I-to-V (TIAs)

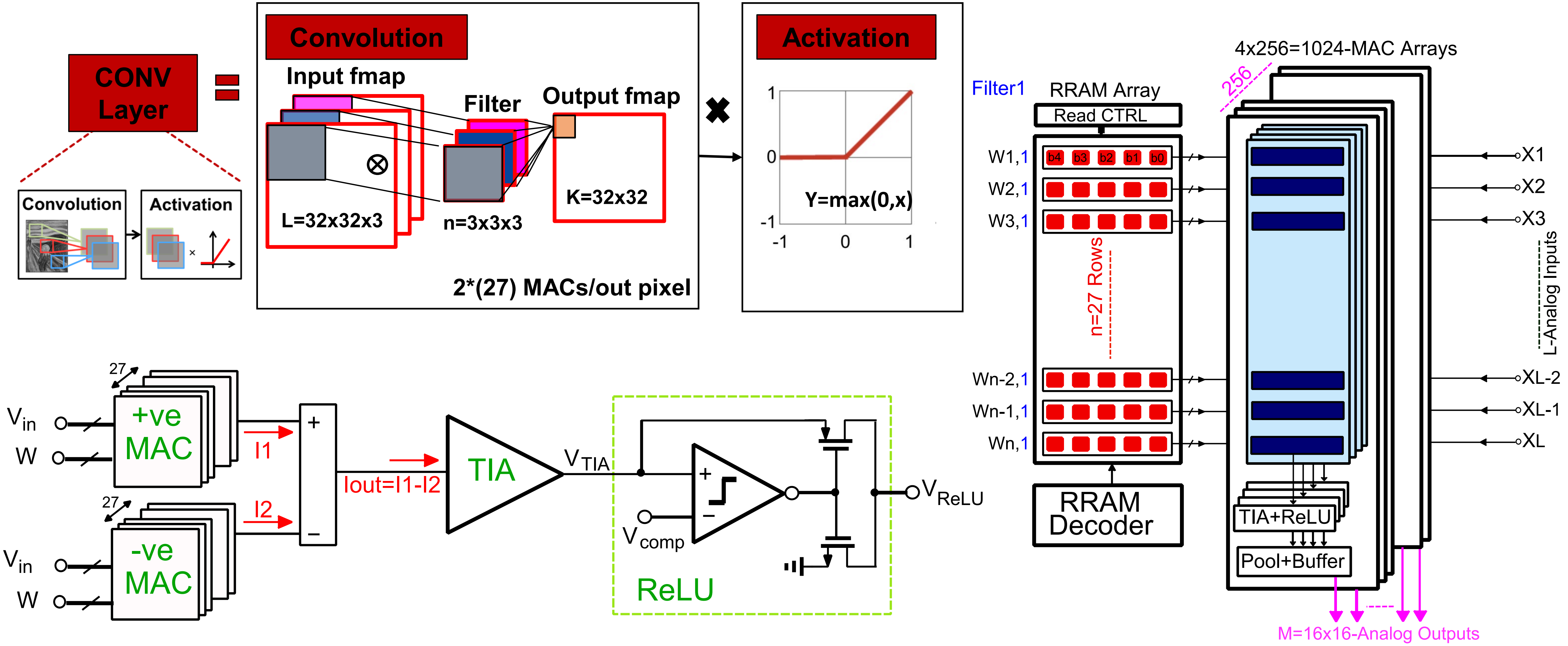
Activation Function

Average POOL

Analog Outputs

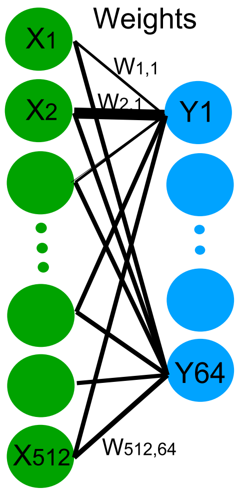
No ADCs/DACs between layers
→ Lower Power, Higher speed

Convolution Layer Implementation

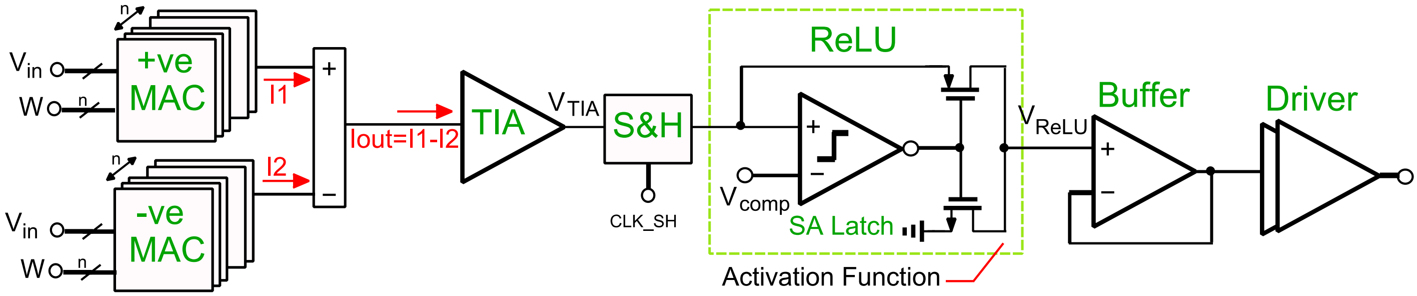


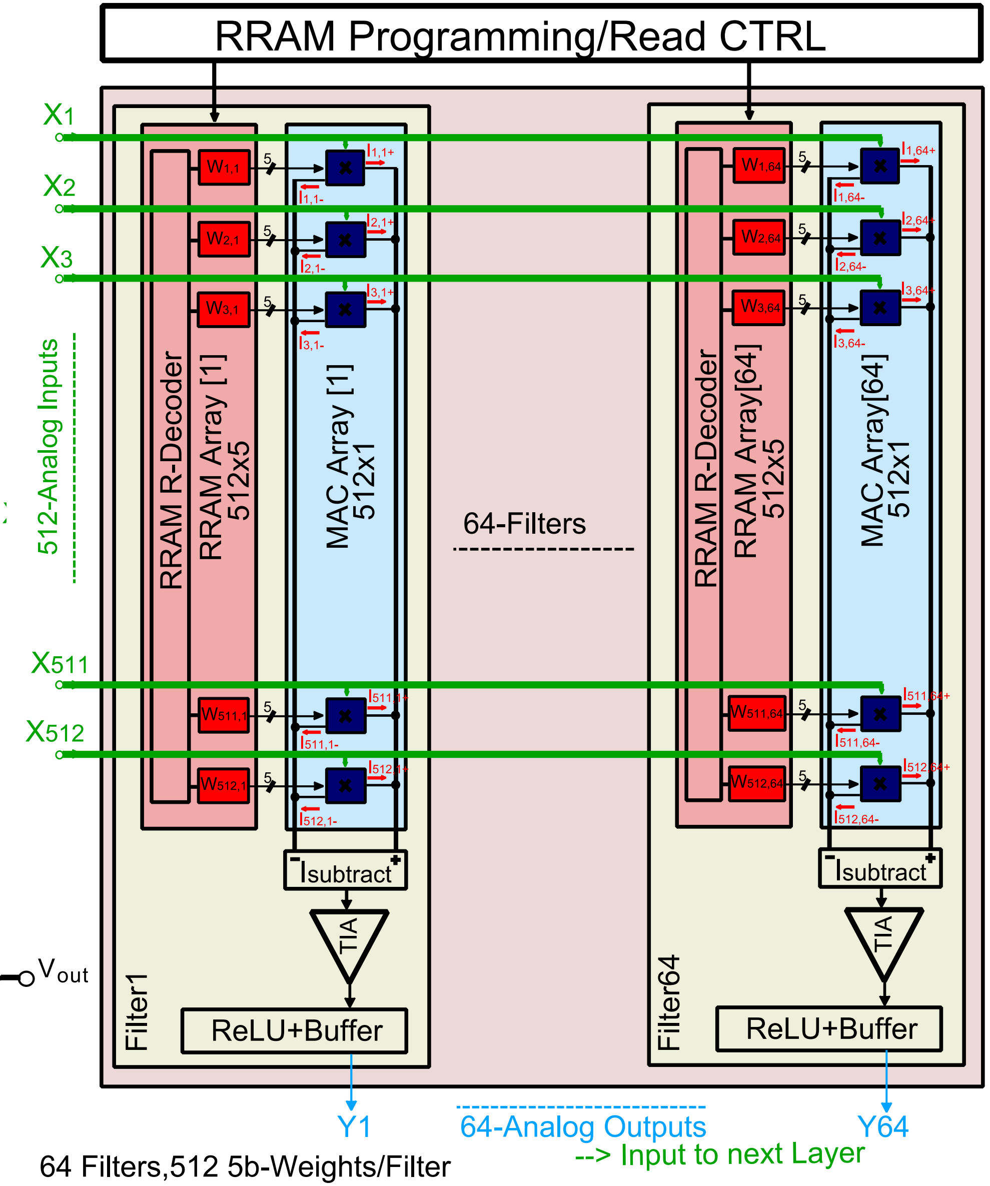
Fully-Connected Layer Implementation

- **All outputs are connected to all inputs**
  - ➢ N Filters to generate N outputs.
  - ➢ Each filter has an RRAM array to store filter weights.



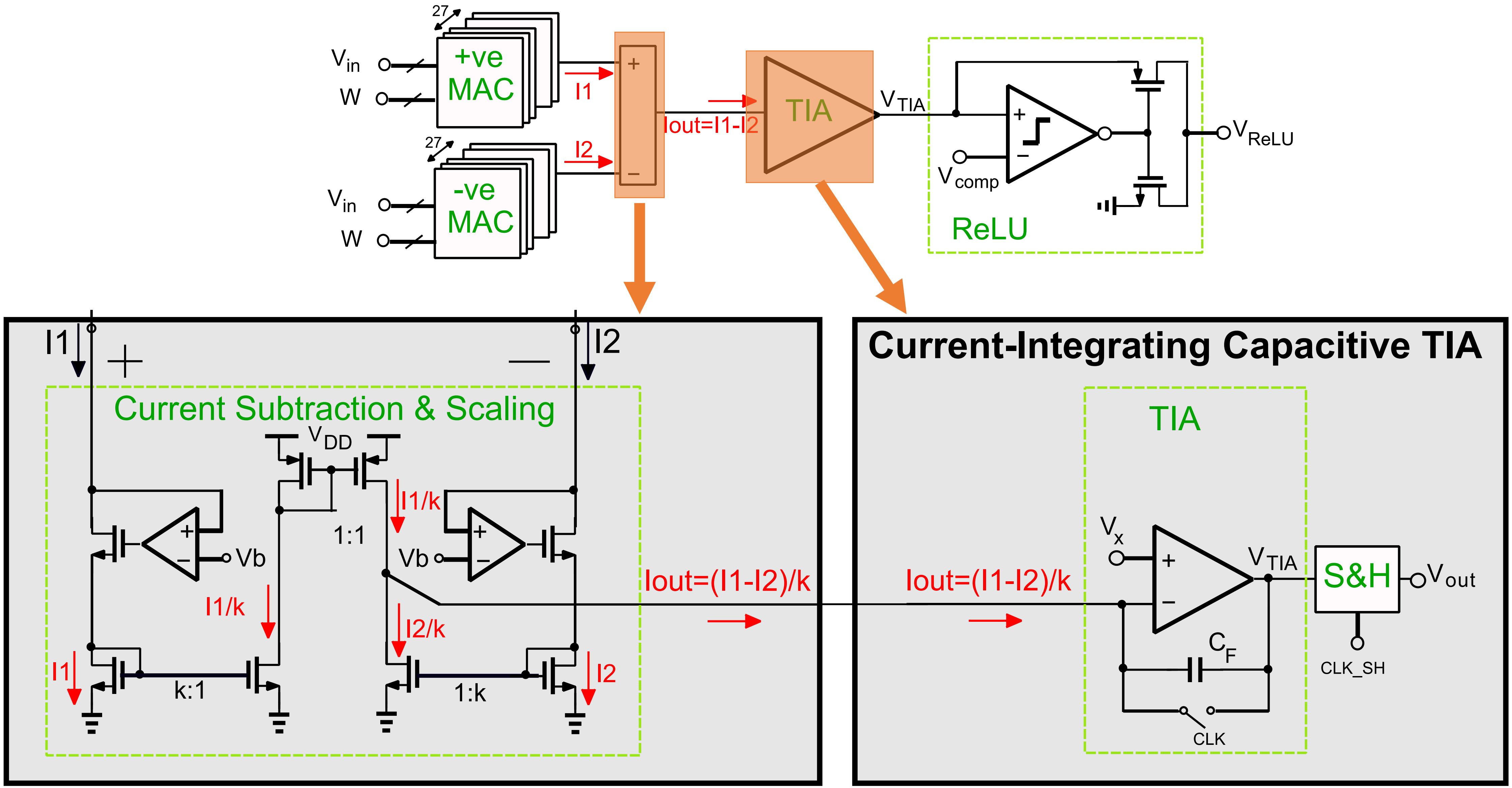
- **Circuit implementation is similar to CONV. layer.**



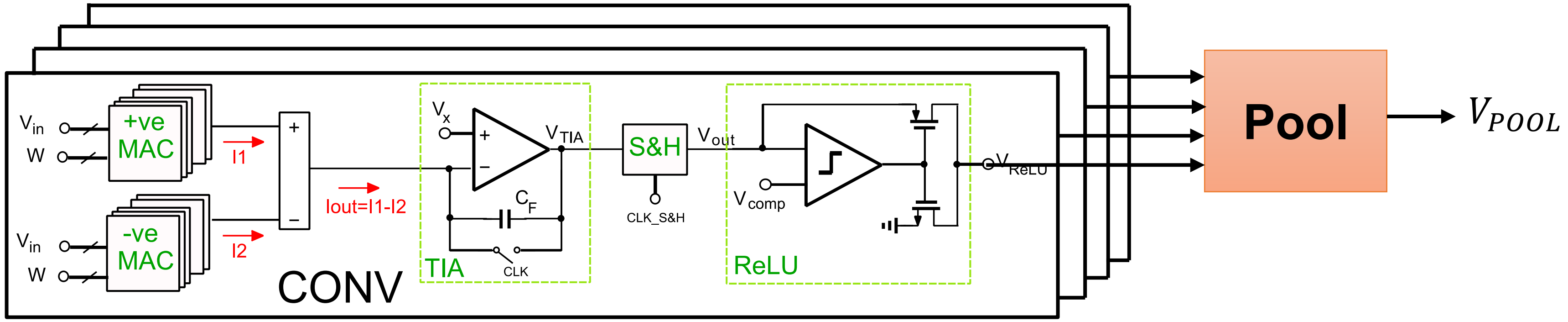


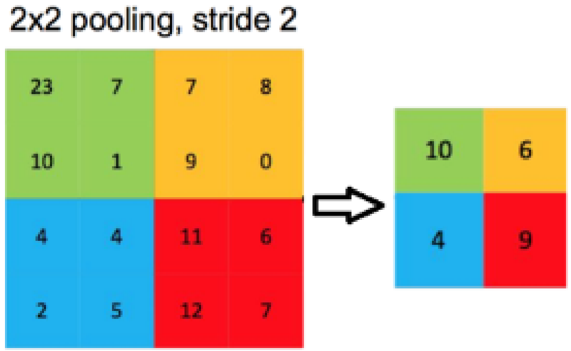
64 Filters,512 5b-Weights/Filter
160 kbit RRAM cells, 512x64 MAC cells

Low-Power Analog Circuits (I-Sense + TIA)



Low-Power Analog Circuits (Pool)




2x2 pooling, stride 2

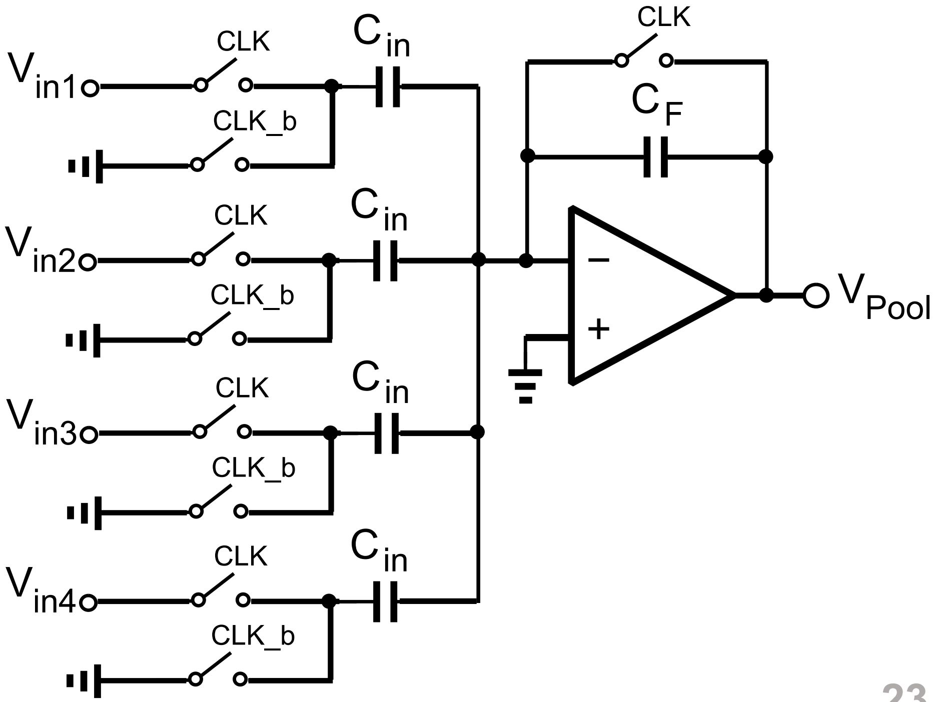
23	7	7	8
10	1	9	0
4	4	11	6
2	5	12	7

10	6
4	9

- **2x2 Average POOL**
  - ➢ Summing amplifier configuration.

$$V_{POOL} = -\frac{C_{in}}{C_f} * \sum_{n=1}^{4} V_{in_n}$$

  - ➢ Works as a buffer to drive the following layer.



Fabricated Die Photo

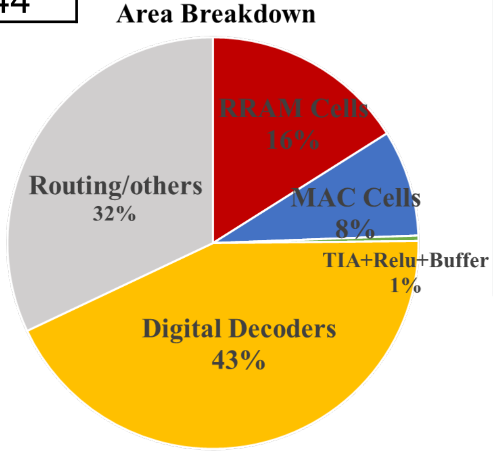
- **SkyWater 130-nm Technology**
  - Monolithic 3D (Si + RRAM)
  - First design + tape-out of US Foundry RRAM

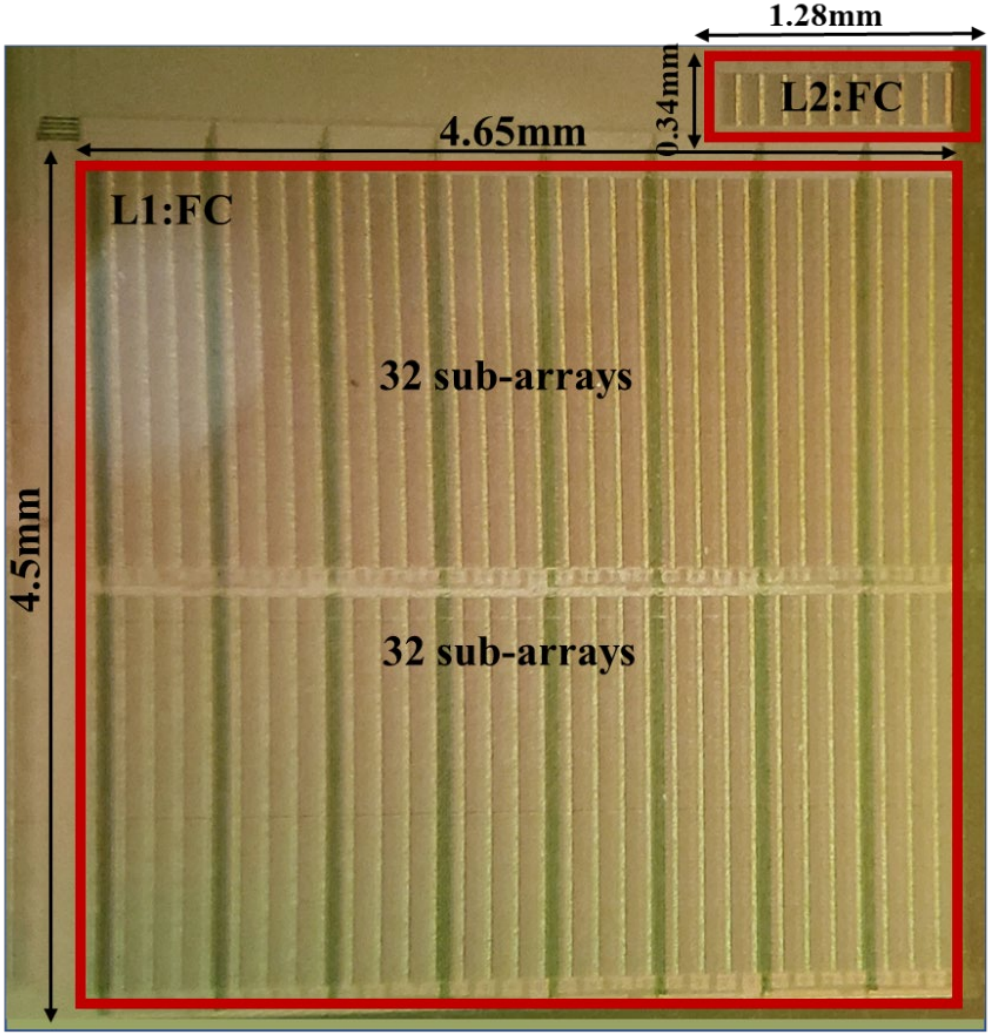
- **Two Successive FC layers**
  - 164 kbit RRAM Cells + 33 k MAC Cells

L1: FC	
No of sub-arrays	64
Subarray RRAM	513 x5
Subarray MACs	513
GOPS	1223
Area (mm^2)	20.9

L2:FC	
No of sub-arrays	10
Subarray RRAM	65 x5
Subarray MACs	65
GOPS	24.21
Area (mm^2)	0.44

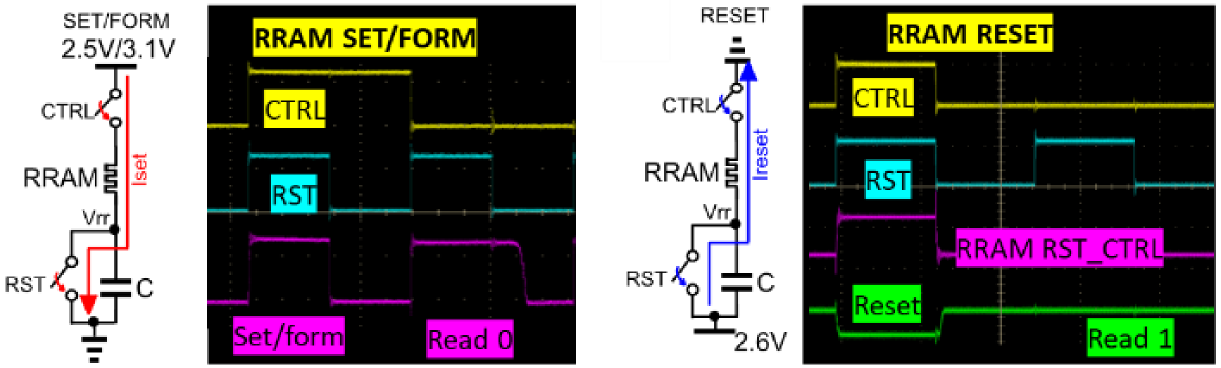
  - RRAMs + MACs area ~ 24% of whole system area.
  - RRAM + MAC density efficiency = 240 GOPS/ mm^2



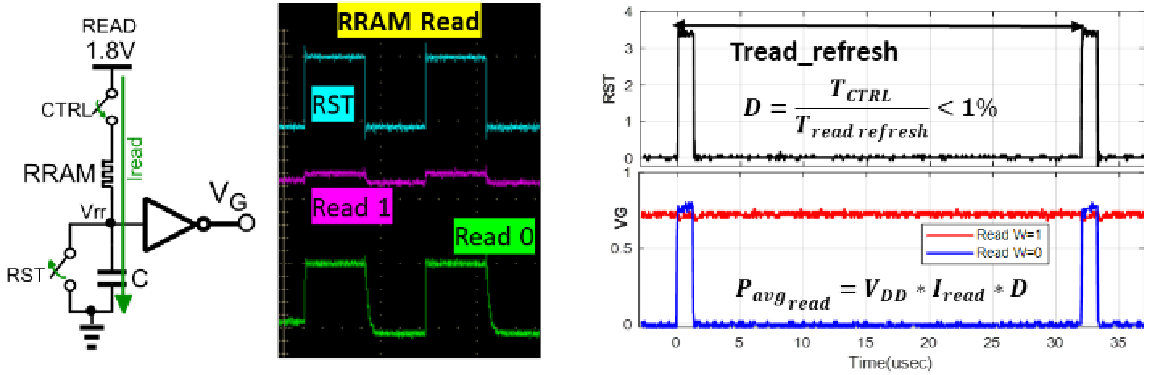


Experimental Measurements Results

1- RRAM Programming



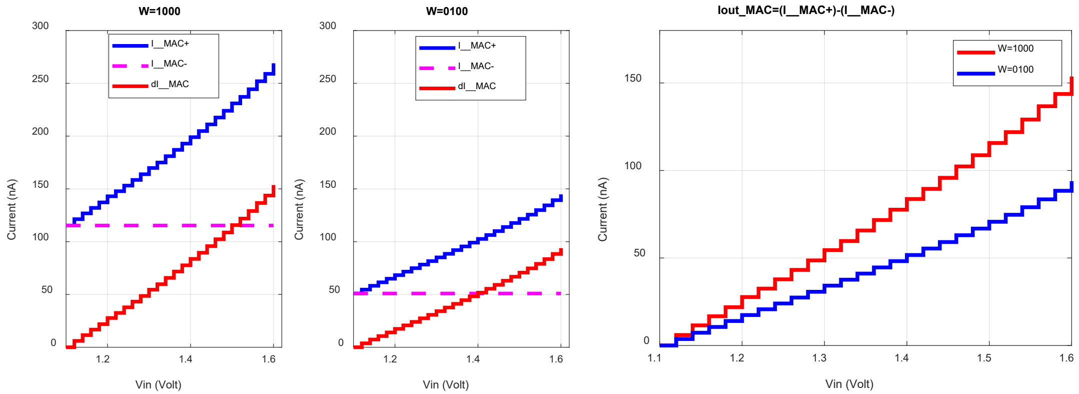
2- RRAM Duty-Cycled Read



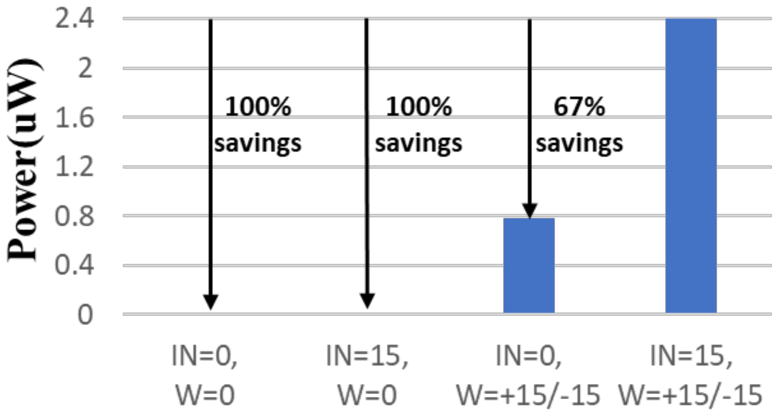
$$D = \frac{T_{CTRL}}{T_{read\ refresh}} < 1\%$$

$$P_{avg_{read}} = V_{DD} * I_{read} * D$$

3- MAC Cell I-V Characterization

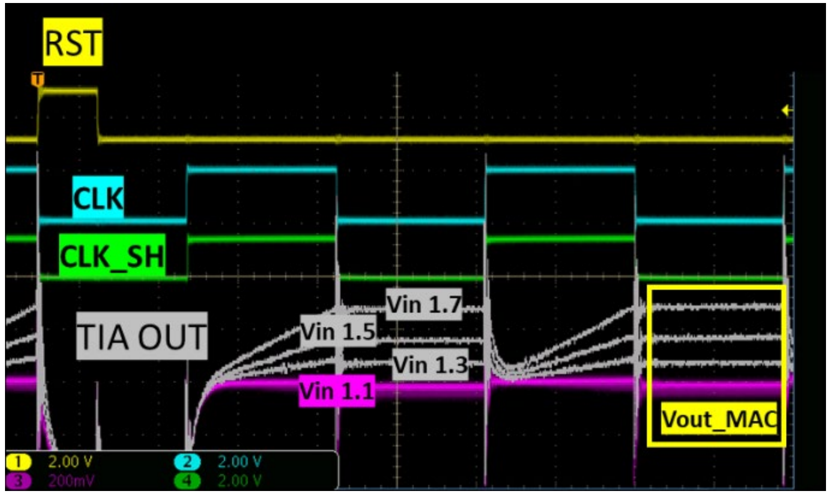


4- MAC Cell Power

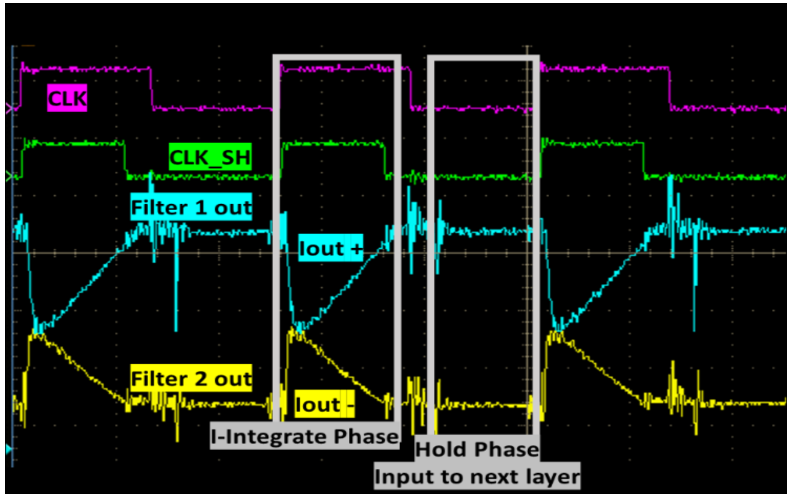


Experimental Measurements Results

5- MAC Cell + Analog Computing (1MAC + TIA+ Isub+ ReLU+ Buffer)

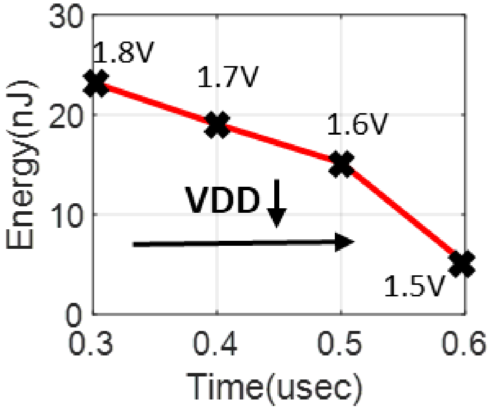


6- Fully-Connected Filters (1 filter = 64MACs+ 320 RRAMs)



7- Power, Energy, and Speed

- Verify the power/speed measured numbers with circuit simulations and models.
- Voltage scaling for higher energy-efficiency.



Measured Performance Summary

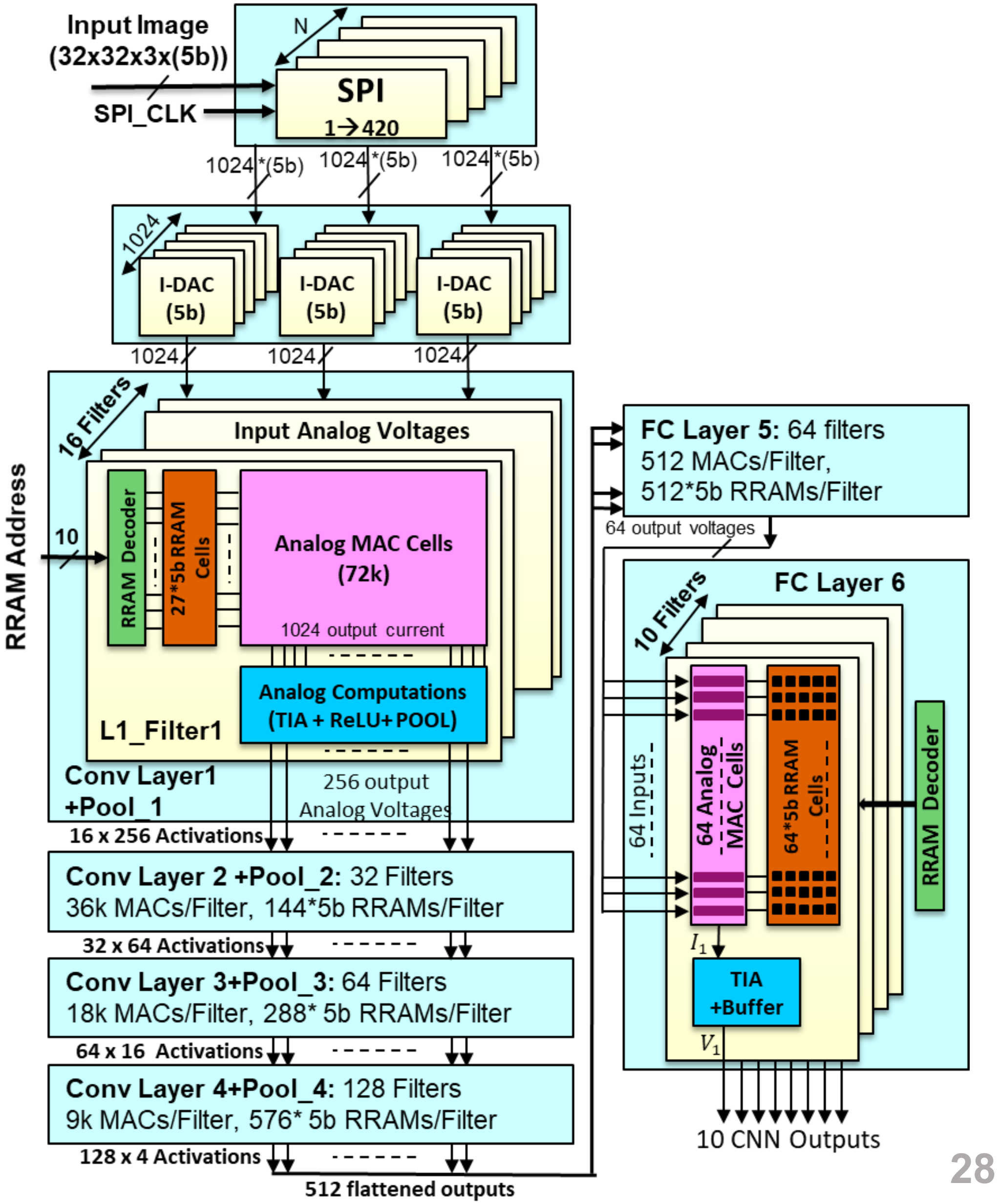
	This Work	ISSCC' 19 [1]	VLSIC'22 [2]	ISSCC'22 [3]	ISSCC'22 [4]
Technology	130-nm	55-nm	40-nm	40-nm	40-nm
Voltage	1.53V		0.9V	0.9V	0.8V-1.1V
nbits/RRAM Cell	1	3	1	1	1
On-chip RRAM Capacity	164kb L1(FC): 513x64x5b L2(FC): 65x10x5b	1Mb 8 Subarrays =8x256x512	64kb L1: 256x256x1b L2: 256x256x1b	2.25MB 288 modules =288x8KB	64kb
Input/output Precision	Analog (4b accuracy)	Digital 2b/3b	Analog	Digital 1-8b/ 32b	Digital 1b/4b
Weight Precision	5b (4b+1b sign)	3b	2b	1-8b	1b
Column Sensing Scheme	Integrating TIA	1b SA	PWM Conversion	ADC	Flash 4b-ADC
Frequency	**1.7 MHz**	-	100MHz	192MHz	64MHz
Throughput (GOPS) [1]	207.9 (Analog 4b/5b)	8361(2b/3b)	13.93 (Analog 8b /2b)	3880.96 (4b/4b)	
Bitwise Throughput[2]	2.07 TOPS	24.5TOPS	222.88 GOPS	7.96 TOPS	5.44 GOPS
Bitwise Throughput/ RRAM Capacity	**25.4 TOPS/Mb**	24.5 TOPS/Mb	3.48 TOPS/Mb	0.44 TOPS/Mb	0.085 TOPS/Mb
Bitwise Density Efficiency (GOPS/mm2)	101.4	-	360	650	201.48
Power (mW)	9		0.504	131.26	0.2
Bitwise Energy Efficiency (TOPS/W)	**230**	131.4	431.52	60.64	26.56

Energy Efficient Analog Design at 130-nm Node

Scaling to a Fully-Parallel Architecture

- **CNN for CIFAR10 image classification**
  - 4 CONV. Layers + 2 FC. Layers
  - 638 kbit RRAM cells for CNN weights.
  - 7.6M MACs, 31k ReLU, and 7.5k Pool.

- **Parallel Computing**
  - Just **1 clock cycle** to generate outputs from each layer.
  - No SRAM to store activation and intermediate data.
  - Much faster classification.
  - Get the classification in 6 clock cycles.



CNN Estimated Performance Summary

Metric	This work	JSSC’18 [6]	IBM TrueNorth [7]	BinarEye [8]	FINN [9]	JSSC’21 [10]
Tech. (nm)	130	28	28	28	28	
Voltage (V)	1.8	0.6V/0.8V	1V	0.7V	-	0.8V
ML. Algorithm	CNN (6-layer)	CNN (9-layer)	CNN	CNN (9-layer)	CNN (9-layer)	CNN
ML. dataset	CIFAR10	CIFAR10	CIFAR10	CIFAR10	CIFAR10	AlexNet
Computation mode	Analog, near memory	Mixed-signal	Neuromorphic	Digital	Digital	Digital
On-chip memory	80kB RRAM	328kB SRAM	-	-	-	3MB RRAM 32Kb SRAM3
# bits/weight	5b (4b+1b sign)	1b	1.6b	1b	1b	1.5b
# bits/activation	analog	1b	1b	1b	1b	
Clock Frequency	1 MHz	10 MHz		10.6 MHz	-	120MHz
Throughput (TOPS)1	2.5	0.48	-	0.48	2.5	0.123
Speed (Frames/sec)	**143k**	237	1.25k	945	**21.9k**	-
Classification Power	400mW	0.9mW	204.4mW	3.7mW	3.6W	127.9mW
Classification accuracy	84%	86%	83.41%	82%	80%	-
Classification Energy	**2.4μJ**	3.8 μJ	164 μJ	**3.8 μJ**	164 μJ	-
Energy Efficiency (TOPS/W)	6.3	533	-	132	0.7	0.96
Active area	621mm^2	6mm^2	-	2.3mm^2	FPGA ZC706	10.8mm^2
Energy-delay product2	**1.4e-11**	1.6e-8	1.3e-7	**4e-9**	1.16e-8	-

>1.6X better Energy, and >350X better EDP

Key Message

- **Analog Near-RRAM Compute**
  - Linear MAC operations
  - Low-power RRAM read
  - Robust against RRAM variations

- **Analog Computations**
  - No ADCs/DACs between CNN layers
  - Energy efficient computing

- **Architectural Parallelism**
  - Tackle bottleneck of data transfer to/from memory.
  - Faster operations
  - 350x better energy-delay product

Acknowledgements

ANALOG DEVICES

skywater

DARPA

DARPA 3DSoC

Thank You!

Questions?

References

[1] S. D. Spetalnick et al., "A 40nm 64kb 26.56TOPS/W 2.37Mb/mm2RRAM Binary/Compute-in-Memory Macro with 4.23x Improvement in Density and >75% Use of Sensing Dynamic Range," 2022 IEEE International Solid-State Circuits Conference (ISSCC), 2022, pp. 1-3, doi: 10.1109/ISSCC42614.2022.9731725.

[2] Cheng-Xin Xue, et al. "16.1 a 22nm 4mb 8b-precision reram computing-in-memory macro with 11.91 to 195.7 tops/w for tiny ai edge devices." 2021 IEEE International Solid-State Circuits Conference (ISSCC). Vol. 64. IEEE, 2021.

[3] Hongwu Jiang, et al. "A 40nm Analog-Input ADC-Free Compute-in-Memory RRAM Macro with Pulse-Width Modulation between Sub-arrays." 2022 IEEE Symposium on VLSI Technology and Circuits (VLSI Technology and Circuits). IEEE, 2022.

[4] Muya Chang, et al. "A 40nm 60.64 TOPS/W ECC-Capable Compute-in-Memory/Digital 2.25 MB/768KB RRAM/SRAM System with Embedded Cortex M3 Microprocessor for Edge Recommendation Systems." 2022 IEEE International Solid-State Circuits Conference (ISSCC). Vol. 65. IEEE, 2022.

[5] C. -X. Xue et al., "24.1 A 1Mb Multibit ReRAM Computing-In-Memory Macro with 14.6ns Parallel MAC Computing Time for CNN Based AI Edge Processors," 2019 IEEE International Solid- State Circuits Conference - (ISSCC), 2019, pp. 388-390, doi: 10.1109/ISSCC.2019.8662395.

[6] D. Bankman, L. Yang, B. Moons, M. Verhelst and B. Murmann, "An Always-On 3.8 μ J/86% CIFAR-10 Mixed-Signal Binary CNN Processor With All Memory on Chip in 28-nm CMOS," in IEEE Journal of Solid-State Circuits, vol. 54, no. 1, pp. 158-172, Jan. 2019, doi: 10.1109/JSSC.2018.2869150.

References

[7] J. Sawada et al., "TrueNorth Ecosystem for Brain-Inspired Computing: Scalable Systems, Software, and Applications," SC '16: Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis, 2016, pp. 130-141, doi: 10.1109/SC.2016.11.

[8] B. Moons, D. Bankman, L. Yang, B. Murmann, and M. Verhelst, “BinarEye: An always-on energy-accuracy-scalable binary CNN processor with all memory on chip in 28nm CMOS,” in Proc. Custom Integr. Circuits Conf. (CICC), Apr. 2018, pp. 1–4.

[9] Umuroglu, Yaman, et al. "Finn: A framework for fast, scalable binarized neural network inference." Proceedings of the 2017 ACM/SIGDA international symposium on field-programmable gate arrays. 2017.

[10] Z. Li et al., "RRAM-DNN: An RRAM and Model-Compression Empowered All-Weights-On-Chip DNN Accelerator," in IEEE Journal of Solid-State Circuits, vol. 56, no. 4, pp. 1105-1115, April 2021, doi: 10.1109/JSSC.2020.3045369.

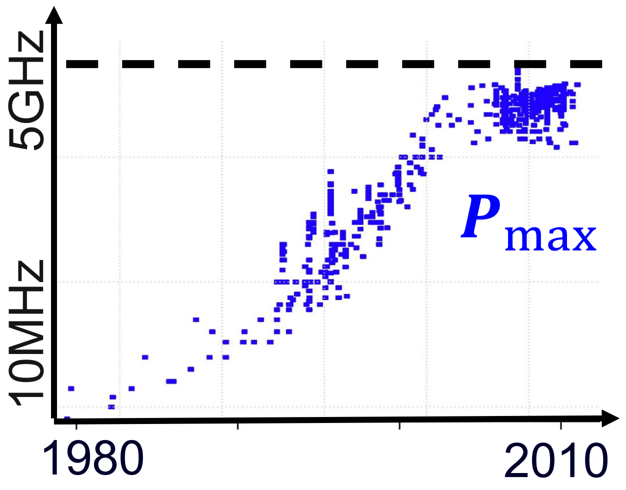
Thank You!

Questions?

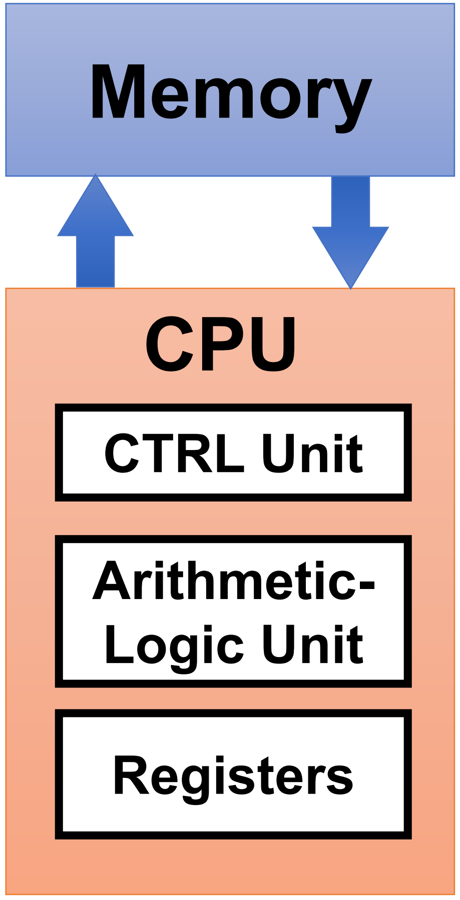
Back-up Slides

Motivation: Computing Limitations

Power-Wall



Limited Power Budget



Memory

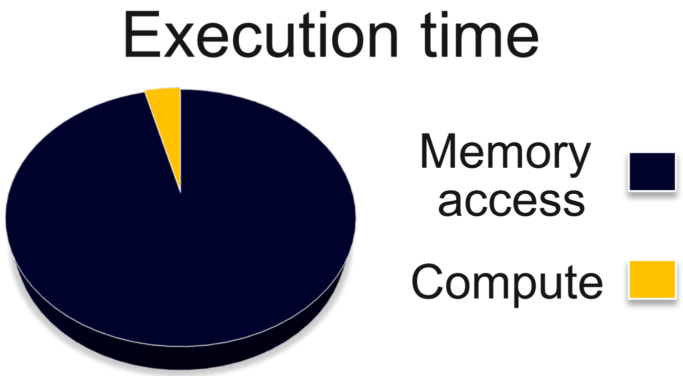
CPU

CTRL Unit

Arithmetic-Logic Unit

Registers

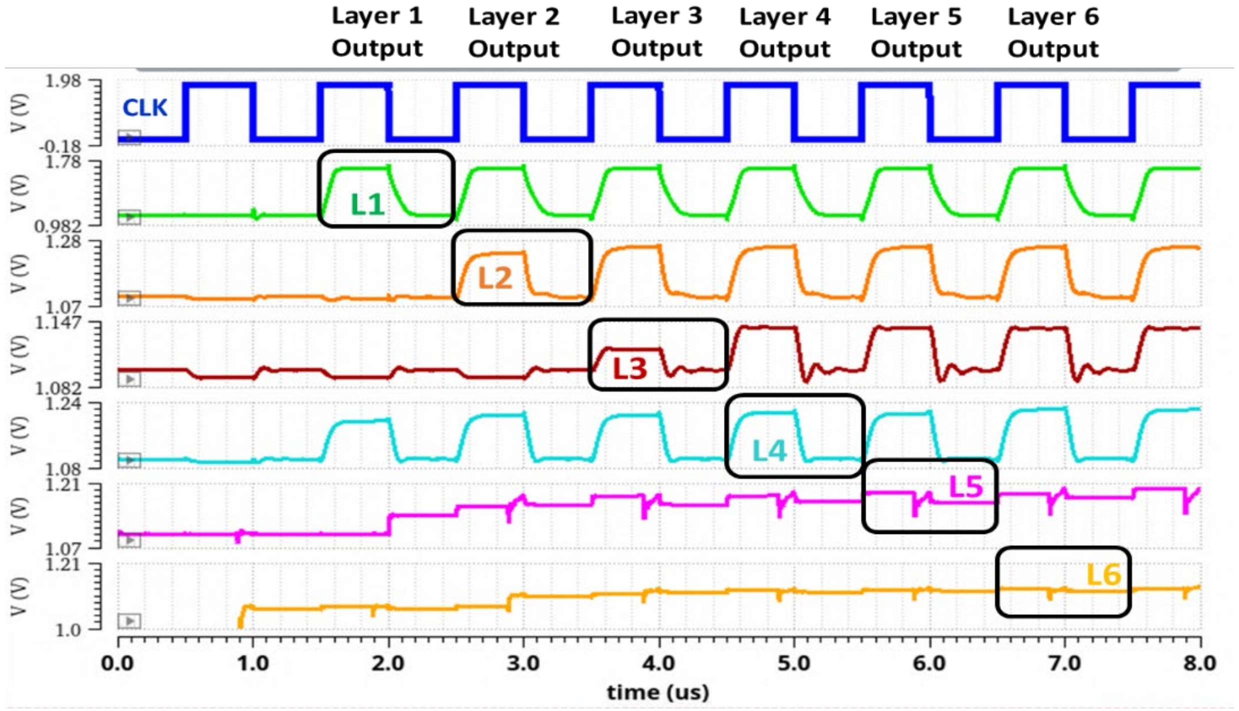
Memory-Wall



Limited Bandwidth

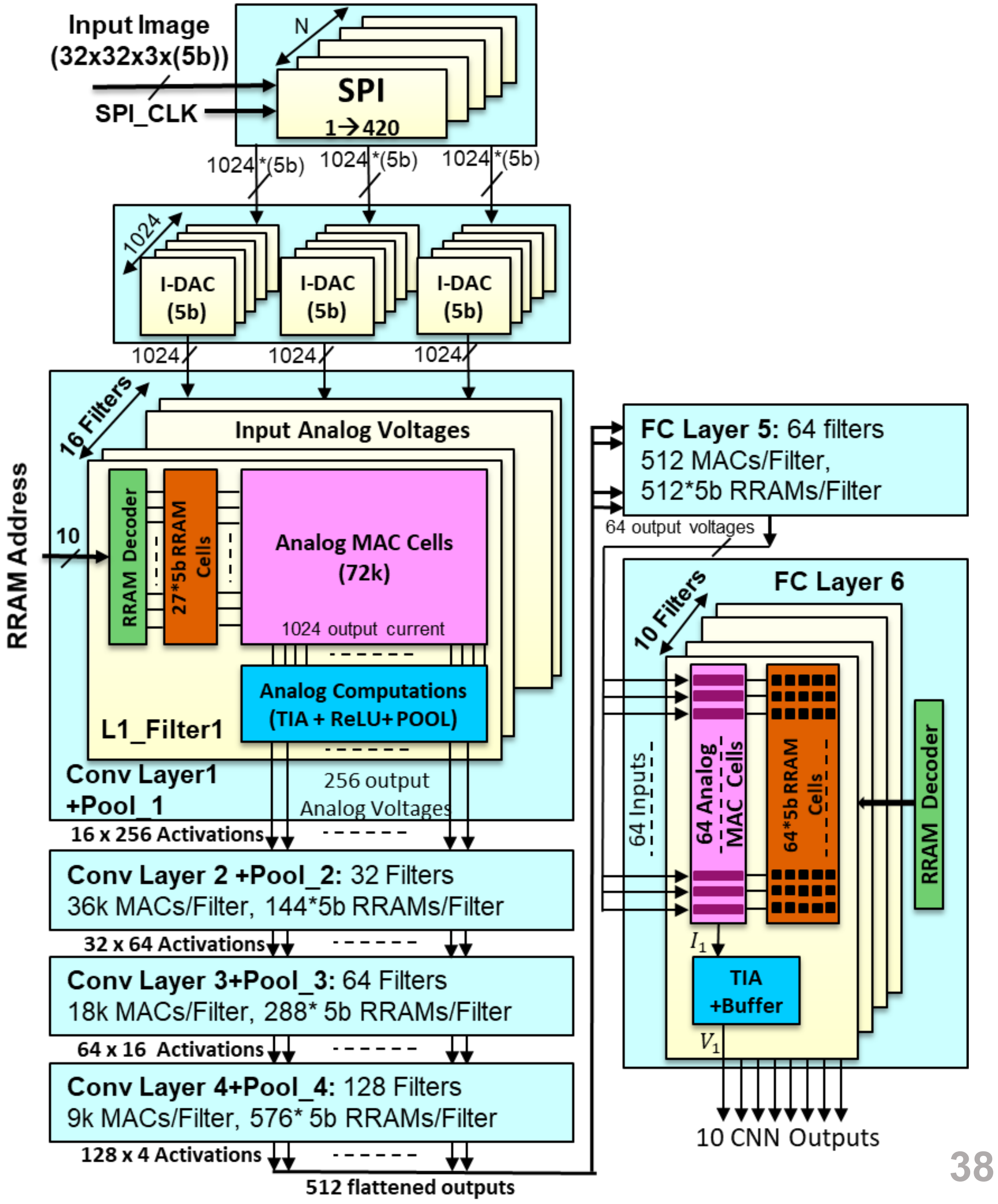
Scaling to a Fully-Parallel Architecture

- **Transient Simulation**



Whole CNN latency ≈ No of CNN layers * T_{CLK}

Only 6 clock cycles to generate output after loading input image



CIFAR 10: Proposed Model (Accuracy = 83%)

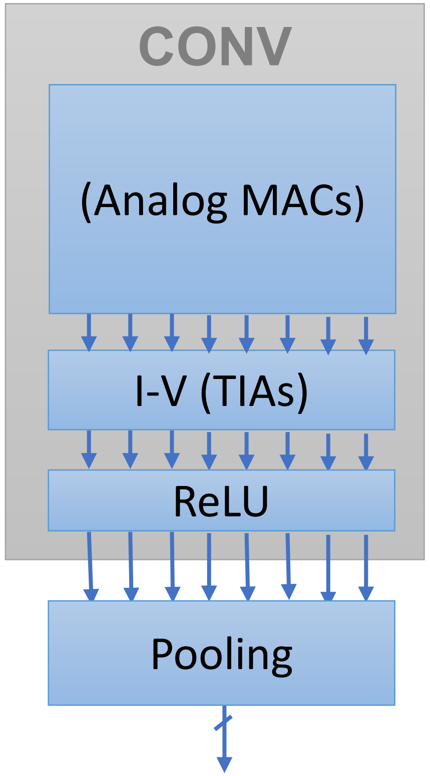
Layer	Input Activations	Output Activations	# Weights	# MACs
Conv1 : 16 (3x3)	32x32x3	32x32x16=16k	16x3x3x3=432	432k
Average Pooling		16x16x16		
Conv2 : 32 (3x3)	16x16x16	16x16x32=8k	32x3x3x16=4.5k	1.125M
Average Pooling		8x8x32		
Conv3: 64 (3x3)	8x8x32	8x8x64=4k	64x3x3x32=18k	1.125M
Average Pooling		4x4x64		
Conv4: 128 (3x3)	4x4x64	4x4x128=2k	128x3x3x64=72k	1.125M
Average Pooling		2x2x128=512		
FC1: 512--64	512	64	512x64=33k	33k
FC2: 64--10	64	10	640	640
Total		30k	128.5k * 5b= 642.5kb	3.8M

Analog Circuits Noise

- Represent mean & sigma of analog computation noise as a percentage of the whole output voltage dynamic range.
- Inject noise as a percentage random variation of output activations in the inference model.

Noise Extraction from simulations

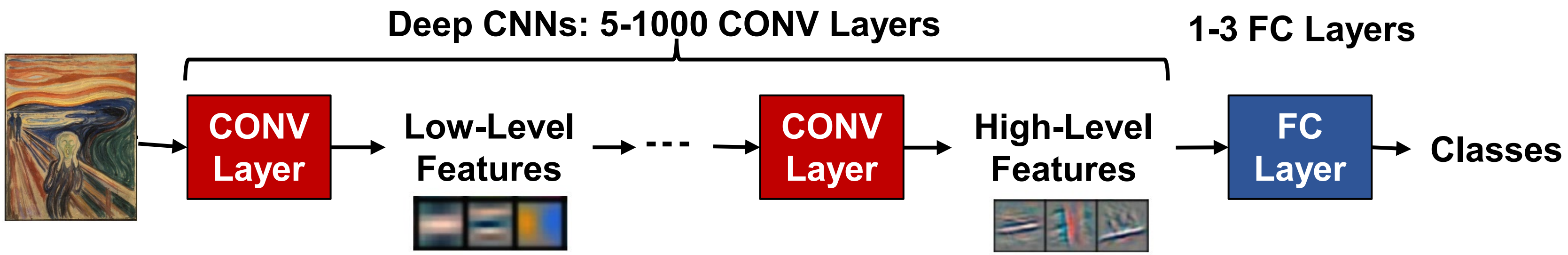
Layer	Mean	RMS Noise
CONV1	0.17%	2.3%
POOL1	0.017%	0.16%
CONV2	0.02%	2.3%
POOL2	0.16%	0.11%
CONV3	0.17%	2.9%
POOL3	0.15%	0.1%
CONV4	-0.12%	4.1%
POOL4	0.15%	0.1%
FC1	0.33%	6.2%
FC2	0.08%	4.7%



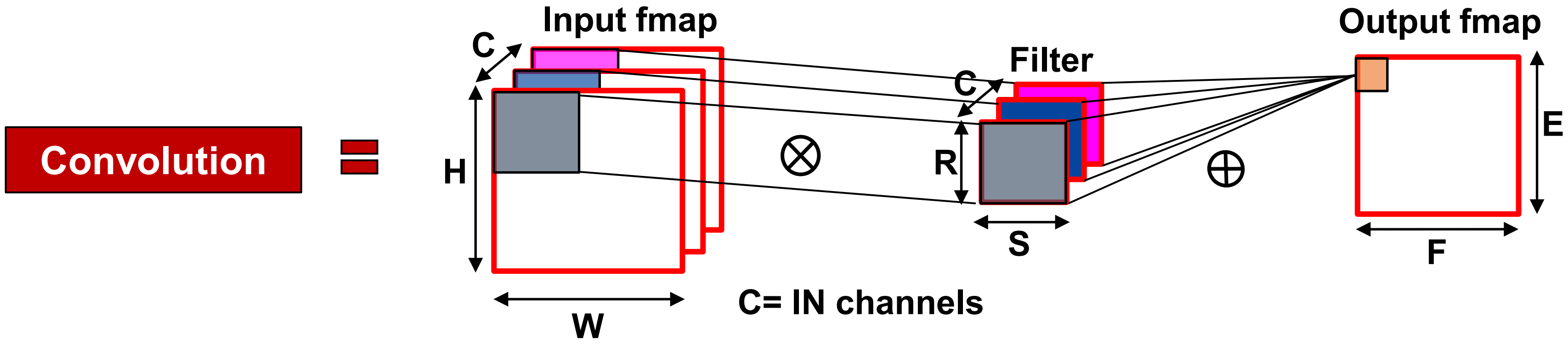
Accuracy loss < 1.5%

Convolutional Neural Networks: CNNs

- CNNs provide state-of-the-art accuracy for wide range of tasks.



- CNNs basic operation is dot product or multiply and accumulate (MAC).



$$Y_{(E,F)} = \sum_{c=1}^{C}\sum_{s=1}^{S}\sum_{r=1}^{R} IN(r+E, s+F, c) * W(r,s,c)$$

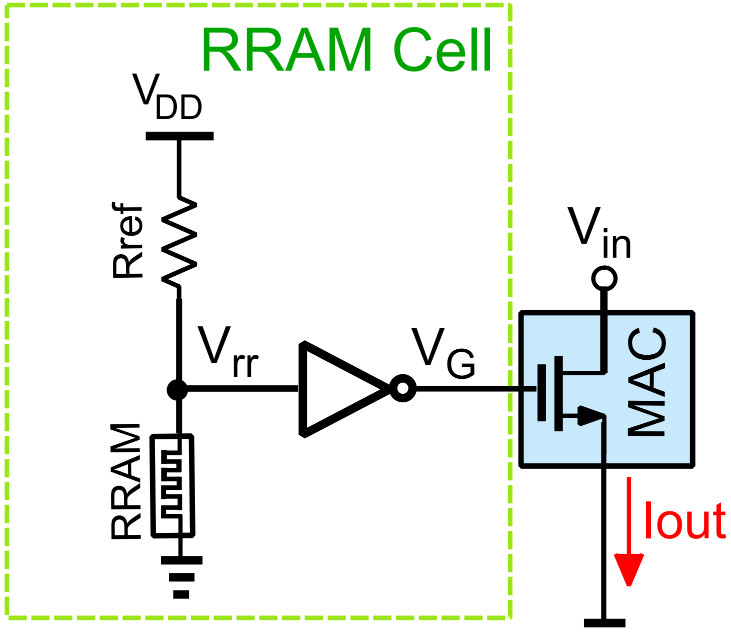
Output Layer

Input Layer

Filter weights

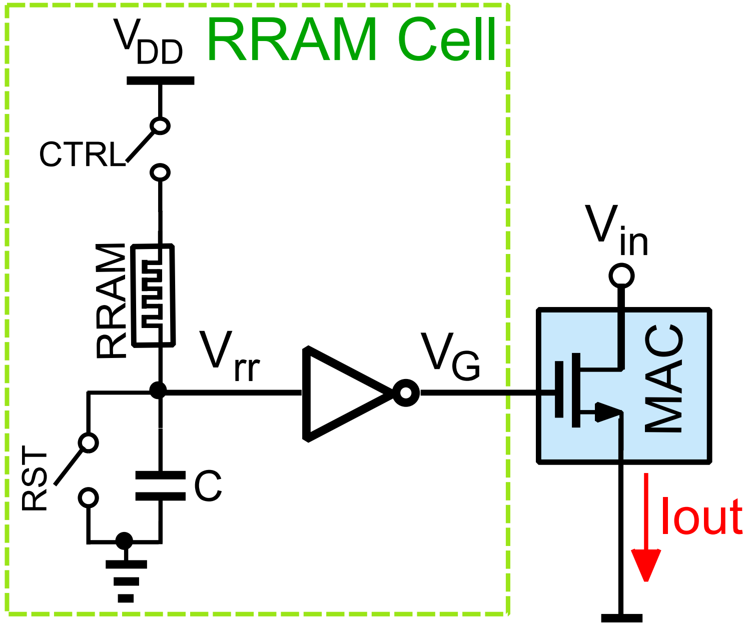
Proposed RRAM Cell

- **RRAM cell turns on/off MAC transistor.**
  - ➢RRAM Cell Output = VDD/0 depending on the RRAM State (HRS/ LRS)



Resistive Divider + Inverter

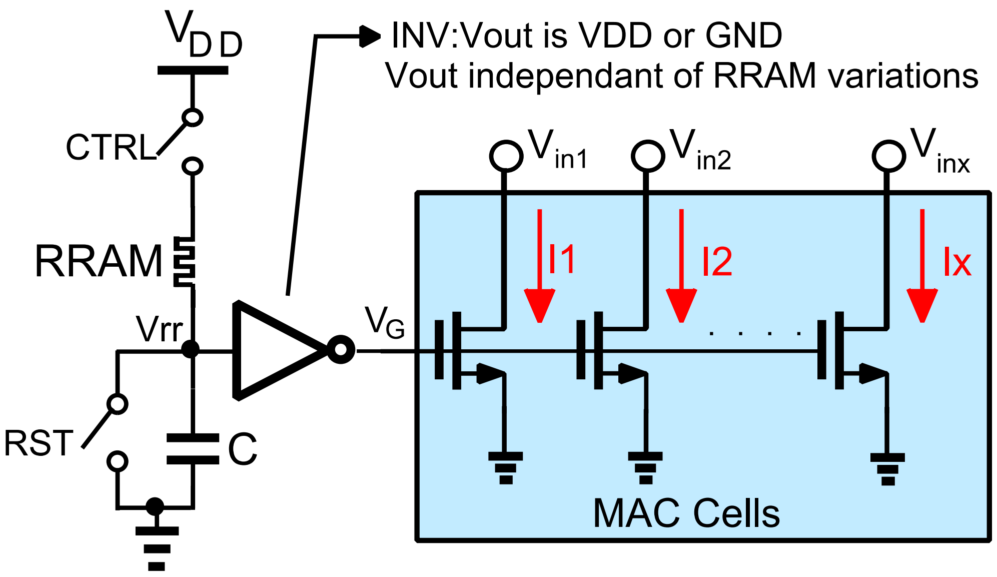
Static Power Consumption



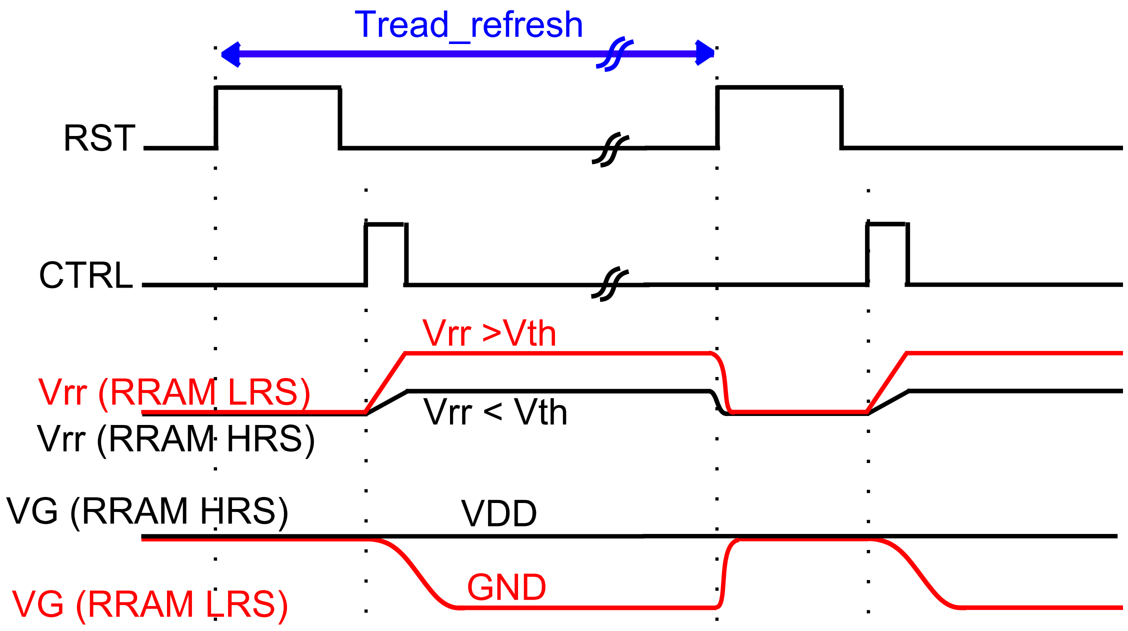
Switches + Capacitor + Inverter

Low Power Consumption

Proposed Pulsed RRAM Read



INV:Vout is VDD or GND
Vout independant of RRAM variations



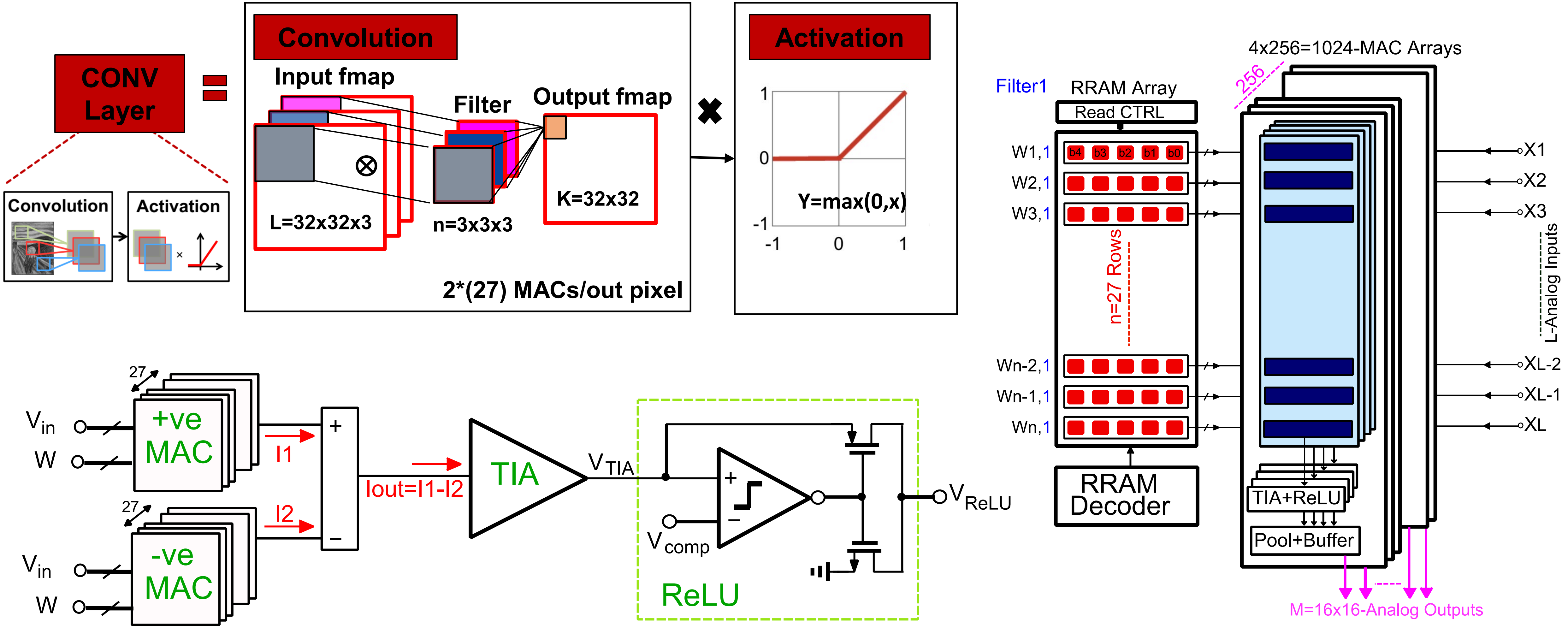
- **Read once at start of the operation**

  No static power and leakage power.

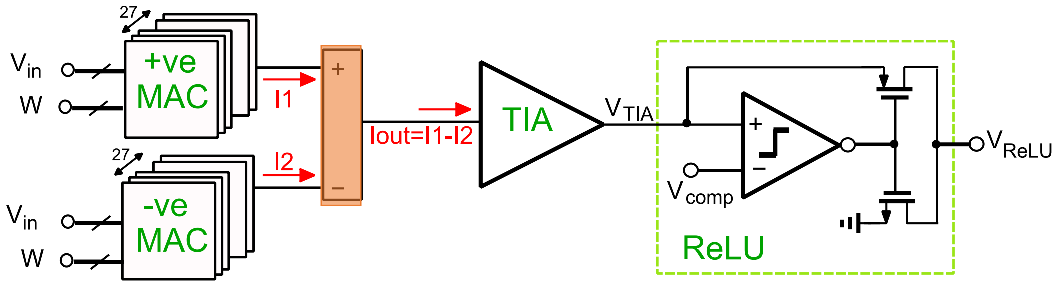
- **RRAM Cell output can be shared with many MAC cells.**
  - ➢ Reduce number of RRAM cells on chips.

- **RRAM Cell output independent of RRAM variations**
  - ➢ Works with 50% RRAM variations and Ron/Roff down to 5

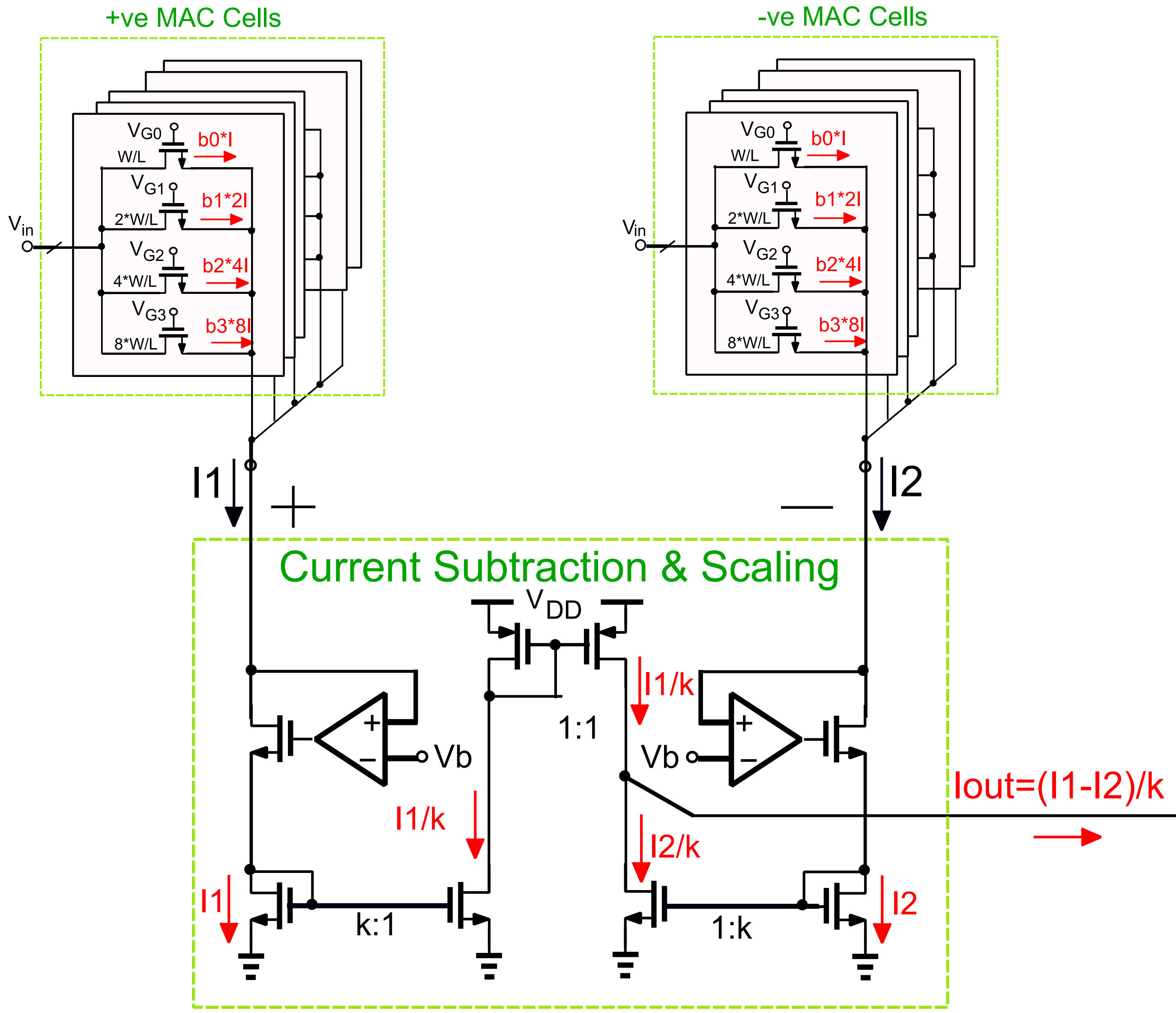
Convolution Layer Implementation



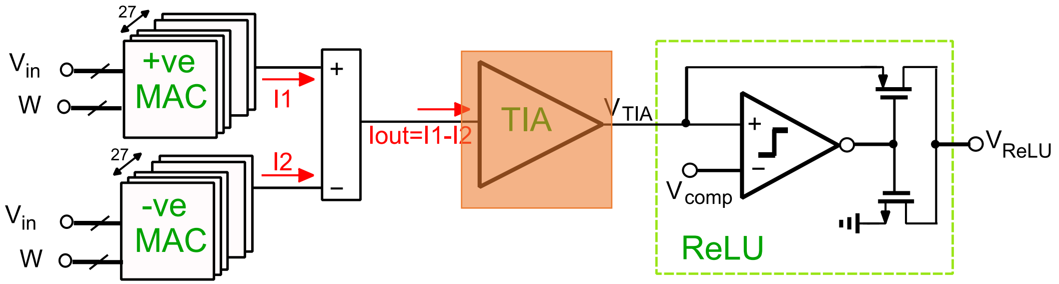
Low-Power Analog Circuits (I-Sense)



- **Differential Current Sensing**
  - ➢ $\mathrm{I_{out}}\ \alpha \sum W * V_{in}$
  - ➢ Opamps to regulate the MAC transistors' source terminals.
  - ➢ Mirror and scale currents down.
    - ➢ Low Power
    - ➢ Relax TIA requirements.



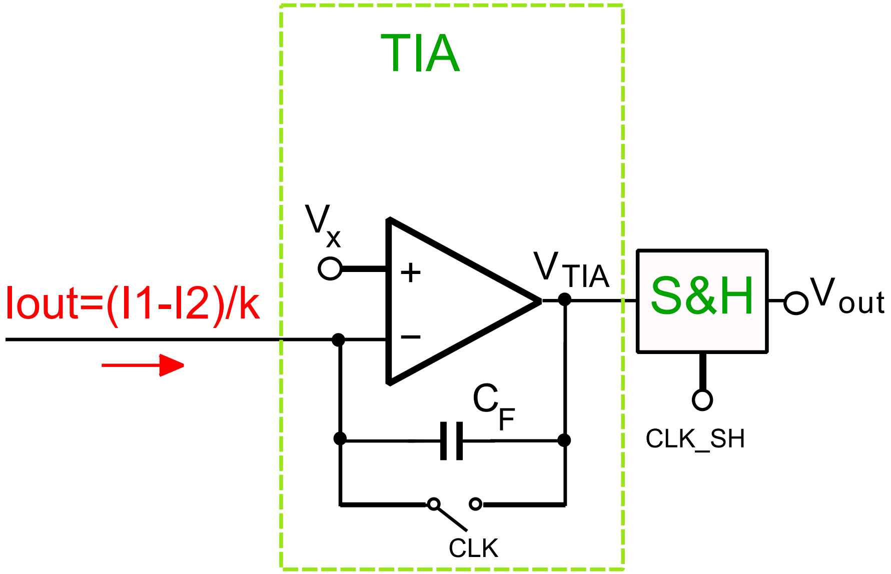
Low-Power Analog Circuits (TIA)

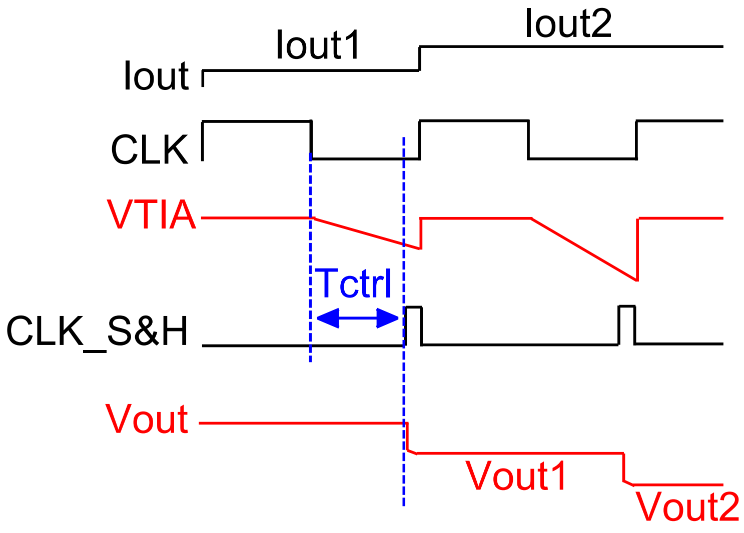


- **Current-Integrating Capacitive TIA**
  - ➢ Lower noise and power vs resistive TIAs
  - ➢ Small area
  - ➢ Controllable and high gain

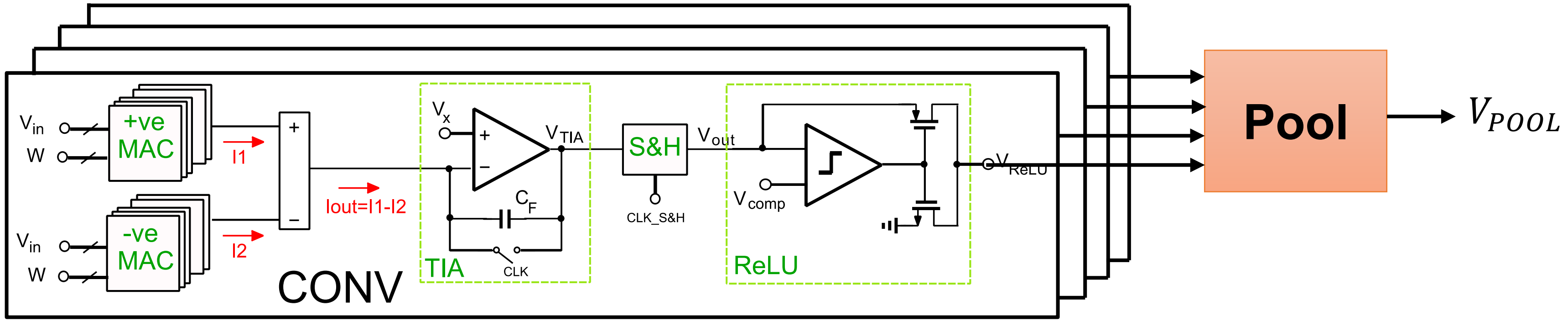
$$V_{out} = \frac{I_{out}}{C_f} * T_{ctrl} = I_{out} * G_{TIA}$$

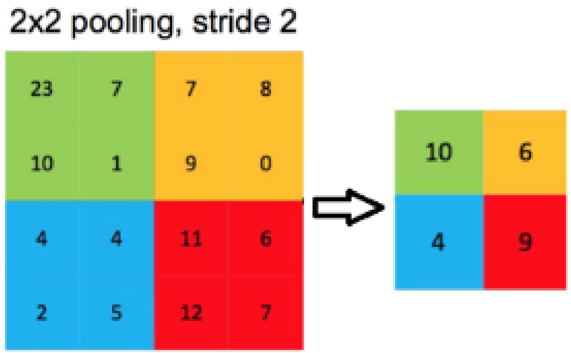
  - ➢ Design to support maximum input MAC currents.
  - ➢ GBW << $1/T_{ctrl}$





Low-Power Analog Circuits (Pool)





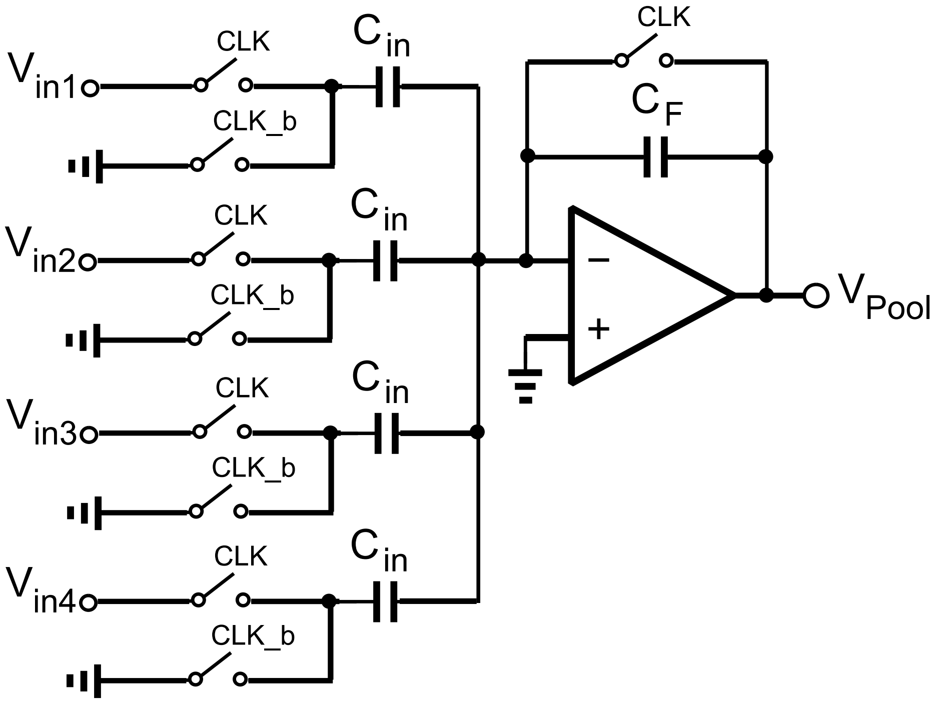
23	7	7	8
10	1	9	0
4	4	11	6
2	5	12	7

10	6
4	9

- **2x2 Average POOL**
  - ➢ Summing amplifier configuration.

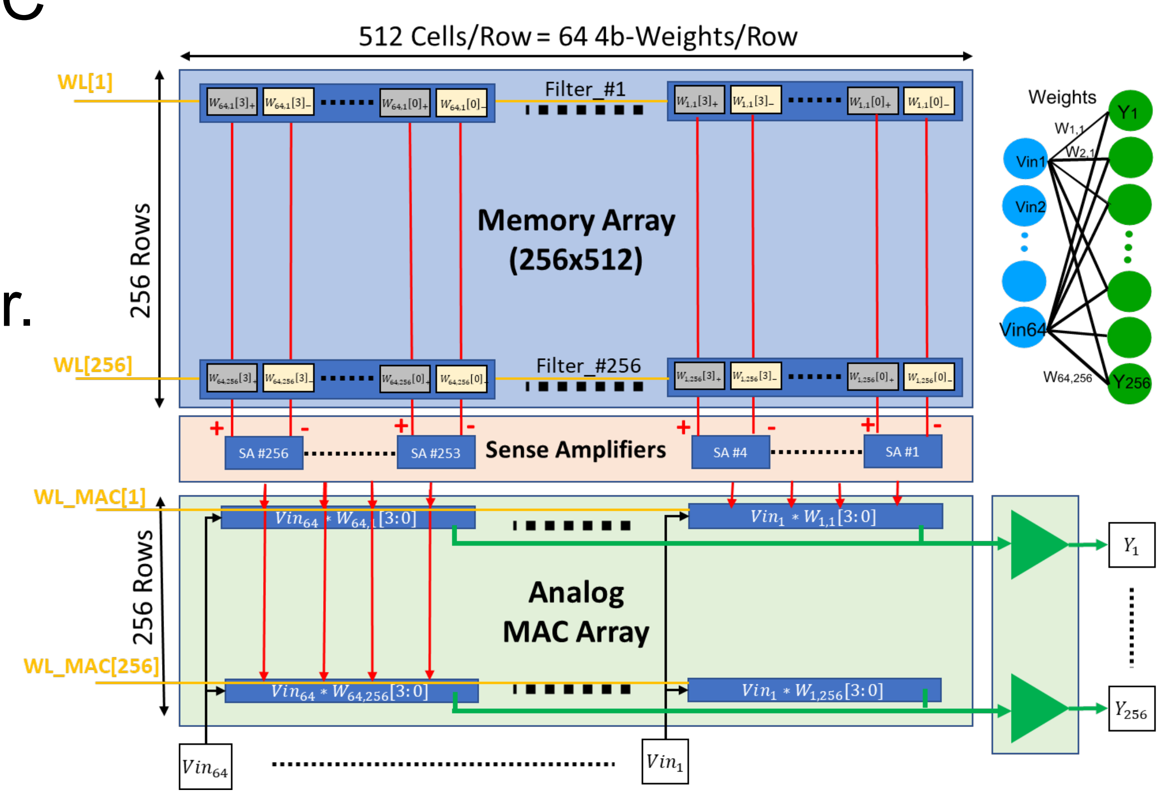
$$V_{POOL} = -\frac{C_{in}}{C_f} * \sum_{n=1}^{4} V_{in_n}$$

  - ➢ Works as a buffer to drive the following layer.
  - ➢ Design for fast and stable settling.
  - ➢ C_f stores the analog output.



Future Directions: Analog NMC Processing Arrays

- **Analog processing near Memory Array**
  - ➢ Integrate memory arrays with analog MAC arrays.
- **Investigate different NV memory devices: MRAM, PCM, FRAM**
  - ➢ 3D integrate memory cells with processor.
- **Improve MAC design**
  - ➢ Non-linear quantization.
  - ➢ Support higher IN/W precision.
- **Investigate energy-efficient techniques**
  - ➢ Pipelining the analog processing arrays.
  - ➢ Power-gating analog MAC arrays.



512 Cells/Row = 64 4b-Weights/Row
WL[1]
WL[256]
256 Rows
$W_{64,1}[3]_+$
$W_{64,1}[3]_-$
$W_{64,1}[0]_+$
$W_{64,1}[0]_-$
Filter_#1
$W_{1,1}[3]_+$
$W_{1,1}[3]_-$
$W_{1,1}[0]_+$
$W_{1,1}[0]_-$
Memory Array
(256x512)
$W_{64,256}[3]_+$
$W_{64,256}[3]_-$
$W_{64,256}[0]_+$
$W_{64,256}[0]_-$
Filter_#256
$W_{1,256}[3]_+$
$W_{1,256}[3]_-$
$W_{1,256}[0]_+$
$W_{1,256}[0]_-$
Weights
$W_{1,1}$
$W_{2,1}$
Vin1
Vin2
Vin64
$W_{64,256}$
Y1
Y256
+
-
SA #256
SA #253
Sense Amplifiers
SA #4
SA #1
WL_MAC[1]
WL_MAC[256]
256 Rows
$Vin_{64} * W_{64,1}[3:0]$
$Vin_1 * W_{1,1}[3:0]$
Analog
MAC Array
$Vin_{64} * W_{64,256}[3:0]$
$Vin_1 * W_{1,256}[3:0]$
Y_1
Y_{256}
Vin_{64}
Vin_1