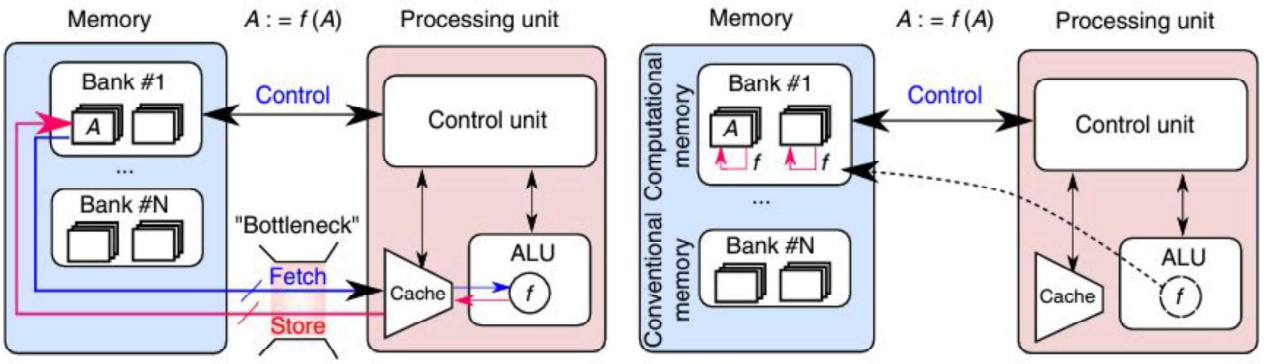
Design of Digital Compute-in-Memory Array

Supervised by:
Santosh Kumar Vishvakarma
Professor
Dept. of Electrical Engineering IIT Indore

Presented by:
Akash Sankhe
MS Research Scholar
Roll No.: 2304102003
Dept. of Electrical Engineering
IIT Indore

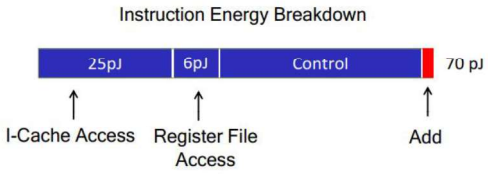
Why Compute-In-Memory ?

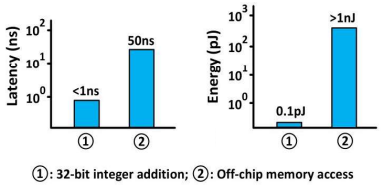


Integer	
Add	
8 bit	0.03pJ
32 bit	0.1pJ
Mult	
8 bit	0.2pJ
32 bit	3.1pJ

FP	
FAdd	
16 bit	0.4pJ
32 bit	0.9pJ
FMult	
16 bit	1.1pJ
32 bit	3.7pJ

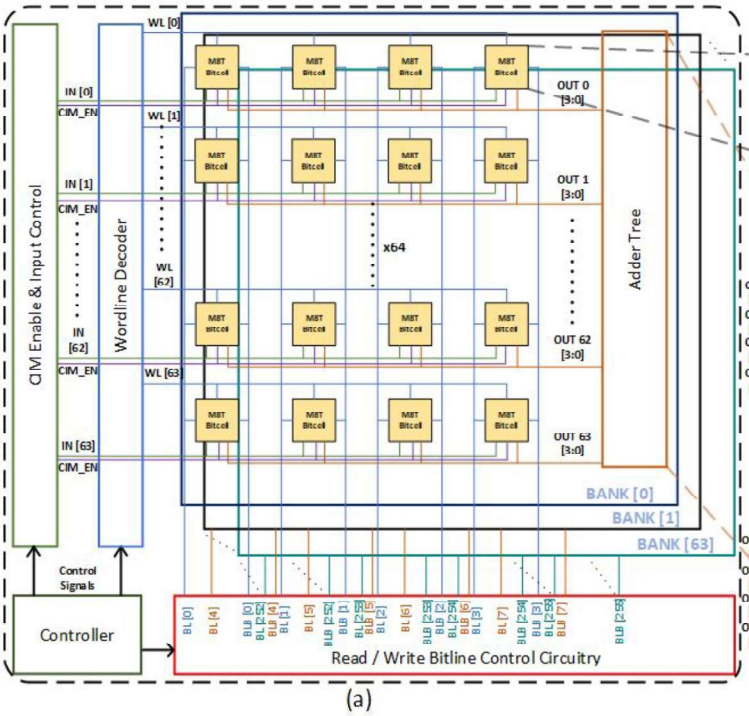
Memory	
Cache	(64bit)
8KB	10pJ
32KB	20pJ
1MB	100pJ
DRAM	1.3-2.6nJ





- Abu Sebastian,et.al. "Temporal correlation detection using computational phase-change memory." Nature Communications 8, no. 1 (2017): 1115.
- Yuzong Chen, et.al. "A reconfigurable 4T2R ReRAM computing in-memory macro for efficient edge applications." IEEE Open Journal of Circuits and Systems 2 (2021): 210-222.
- Mark Horowitz. "1.1 computing's energy problem (and what we can do about it)." In 2014 IEEE international solid-state circuits conference digest of technical papers (ISSCC), pp. 10-14. IEEE, 2014.

Work 1: 484 TOPS/W & 6.98 TOPS/mm^2 DCIM Macro (TSMC 65nm)



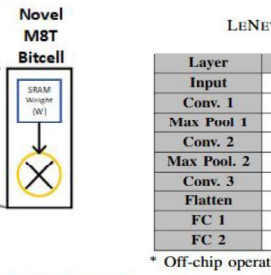
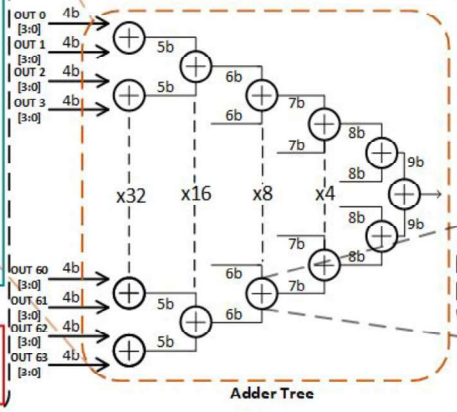
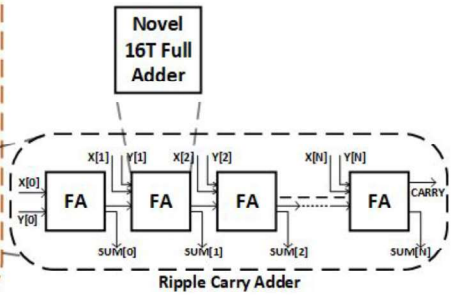


TABLE I
LENET-5 ARCHITECTURE SPECIFICATIONS(MNIST)

Layer	Kernal	Output	Banks Used	Op Cycles
Input	-	28 x 28 x 1	NA	-
Conv. 1	5 x 5 x 6	24 x 24 x 6	6	576
Max Pool 1	2 x 2	12 x 12 x 6	NA	-
Conv. 2	5 x 5 x 16	8 x 8 x 16	48	64
Max Pool. 2	2 x 2	4 x 4 x 16	NA	-
Conv. 3	1 x 1 x 120	4 x 4 x 120	60	16
Flatten	NA	1920 x 1	NA	-
FC 1	84	10	168	5760
FC 2	10	1	20	84

* Off-chip operations and post-accumulation cycles are not considered.



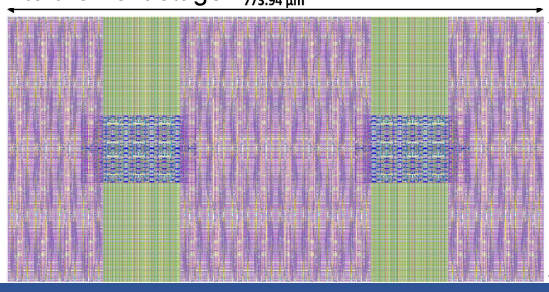


- Akash Sankhe, et.al. A 484 TOPS/W & 6.98 TOPS/mm^2 Novel Digital Compute-In-Memory Macro for Edge AI Applications (Manuscript submitted in IEEE Transactions on Very Large Scale Integration (VLSI) Systems)

Principle Contributions

Motivation

- ACIM are susceptible to PVT variations due to analog to digital and vice versa conversions of the signals. Also susceptible to bit-flipping issues when multiple wordlines are activated simultaneously.
- Current SOTA works are not addressing the threshold voltage loss issue when the multiplication of input activation and stored weight results in output logic '1'.
- Using techniques like inverter insertion, custom 12T adder with complement inputs and pass transistor logic based FAs which either require complementary input signals or have threshold voltage loss at the output which is passed on to the next stage.

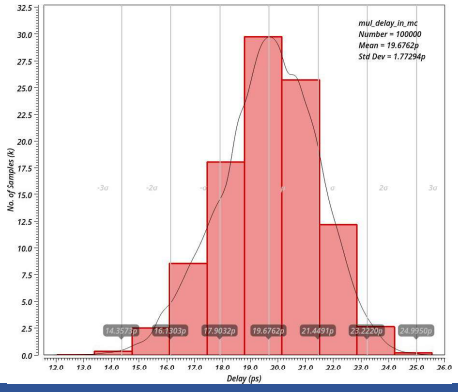


Layout of the proposed DCIM macro

Addressed

- Dual purpose M8T SRAM bitcell which in addition to its function as a storing element, also performs multiplication operation between stored weight and input activation providing full swing output when the result is logic '1'. While having a T_{mul} Std. Dev. of only 1.77ps.
- Area efficient adder tree design which uses RCAs having 16T FA which provide rail-to-rail swing at its output while reducing the transistor count by 42.86% and overall area of the adder tree by 65.71%.

Monte-Carlo simulation for T_{mul} (100k samples)

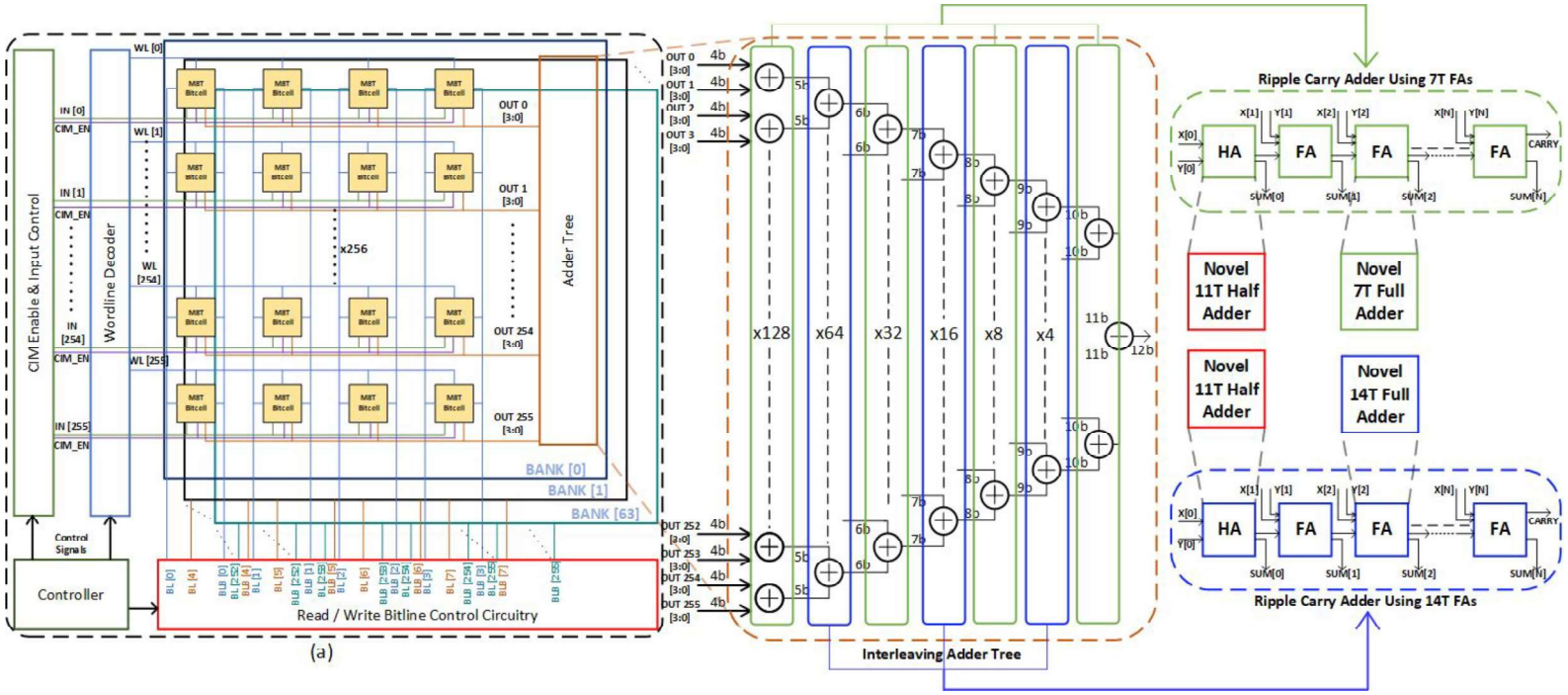


Comparison with State-Of-The-Art Works (16Kb)

	ISSCC'21 [6]	ESSCIRC'23 [11]	JSSC'24 [4]		ISCAS'23 [5]	TCAS'24 [7]	JSSC'19 [12]	JETCAS'22 [9]	JSSC'21 [3]	TNANO'23 [10]	This Work
			DIMCA2	DIMCA1							
Technology [nm]	22	28	28	28	40	55	65	65	65	65	65
MAC Operation	Digital	Digital	Digital	Digital	Digital	Digital	AMS	Analog	Digital	Digital	Digital
Cell Type	6T	6T+0.5T	8T	8T	PT-8T	8T	10T	AND8T	NA	10T	M8T
Supply Voltage [V]	0.72	0.6-1.1	0.45-1.1	0.45-1.1	0.8-1.0	1.2	0.8-1.0	1	0.6-0.8	1.2	1.2
Array Size	64Kb	16Kb	16Kb	16Kb	16Kb	64Kb	16Kb	16Kb	16Kb	16Kb	16Kb
Bitcell Area [μm^2]	0.379	0.379	NA	NA	NA	4.278	NA	2.778	10.53	4.504	4.21
Macro Area [mm^2]	0.202	0.0159	0.033	0.049	NA	2.8	NA	NA	0.2272	NA	0.2913
Activation Precision [bit]	1-8	1-8	1	1-4	1-8	4/8/12/16	6	4	1-16	4	1-4
Weight Precision [bit]	4/8/12/16	8	1	1	4/8	4/8/12/16	1	4	1-16	1-4	4
Model	NA	NA	VGG like	VGG like	NA	ResNet-10	LeNet-5	VGG-8 like	LeNet-5	CNN-Type	LeNet-5
Accuracy	NA	NA	86.96	90.41	NA	93.7	98.3	96.05	99.2	98.67	99.07
Operating Frequency [MHz]	100	30-360	280	250	100	200	5	100	138	25	250
Throughput [TOPS]	0.825	NA	9.175 @ 0.9V 20.032 @ 1.1V	2.035 @ 0.9V 4.804 @ 1.1V	0.82	NA	0.064	0.41	0.567	0.82	2.032
Energy Efficiency [TOPS/W]	22.25	22.4-60.4	1108 @ 0.9V 2219 @ 0.5V	154 @ 0.9V 248 @ 0.5V	94	66.3	40.3	180.14	156	273	484
Compute Density [TOPS/mm^2]	4.08	0.12-1.46	607	98	NA	0.288	NA	NA	2.5	NA	6.98

- The macro achieves inference accuracy of 98.7% and 98.77% for 1A4W and 99.07% and 97.81% for 4A4W for MNIST & A-Z alphabet datasets.
- The macro achieves improvement of 1.95x and 2.8x in energy efficiency and compute density respectively when compared to SOTA works.

Work 2: Interleaving Adder Tree Structure for 64Kb DCIM Macro (TSMC 65nm)



- Manuscript to be submitted in IEEE Transactions on Nanotechnology Journal.

Principle Contributions

Motivation

- ACIM are susceptible to PVT variations due to analog to digital and vice versa conversions of the signals. Also susceptible to bit-flipping issues when multiple wordlines are activated simultaneously.
- Current SOTA works are not addressing the threshold voltage loss issue when the multiplication of input activation and stored weight results in output logic '1'.
- Using techniques like inverter insertion, custom 12T adder with complement inputs and pass transistor logic based FAs which either require complementary input signals or have threshold voltage loss at the output which is passed on to the next stage.

Addressed

- Adder tree design which uses interleaved columns of RCAs made from 14T & 7T FAs and which also use 11T HA in the first stage of RCA. The 14T FA and 11T HA provides rail-to-rail swing but 7T FA does have threshold voltage drop at its output but its effect is mitigated by the interleaved structure of the adder tree.

THANK YOU