Aalto University
School of Electrical
Engineering

ELEC-L352001: Study on analog memories to exclude data conversion in neural networks

Miikka Tenhunen

miikka.tenhunen@aalto.fi

25.5.2022

Outline

- Background
- Paper 1: DARAM
- Paper 2: ARCHON
- Paper 3: eDRAM
- Assignment

Background: convolutional neural networks (CNN)

- CNNs classify image data based on detected patterns
    - Image and video analysis, autonomous vehicles, medical applications...
- Key operation is convolutional filtering
    - Filter matrix is slid over the image to extract patterns such as edges
- Filtering consists of multiply-accumulate (MAC) operations
    - Element-wise multiplication between filter and image $\Rightarrow$ multiplication results summed $\Rightarrow$ feature map for next layer
- MAC typically done using digital processing
- Lately analog MAC has gained attention due to its efficiency (low power, small area, high speed)

Background: analog MAC and memory

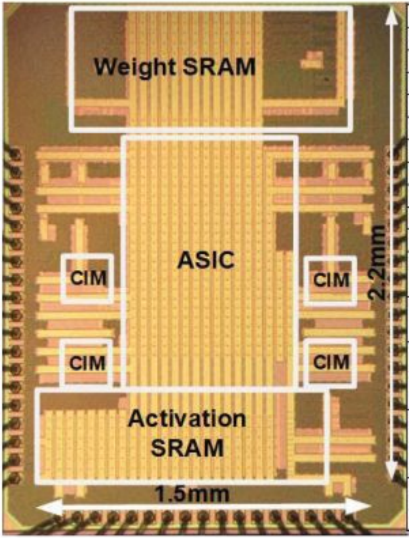
- Analog MAC requires at least one of its inputs to be analog
  - Filter weights or image data
  - Frequent use of DACs
- Analog MAC result must be stored somehow for further processing
  - Frequent use of ADCs to store to digital memory
- Data converters are slow and use a lot of power $\Rightarrow$ analog MAC potential wasted
- Converter usage may be reduced by using analog memory
- Analog memory = capacitor that stores analog voltage + a few transistors

A 65nm 3T Dynamic Analog RAM-Based Computing-in-Memory Macro and CNN Accelerator with Retention Enhancement, Adaptive Analog Sparsity and 44TOPS/W System Energy Efficiency

Zhengyu Chen, Xi Chen, Jie Gu
ISSCC 2021

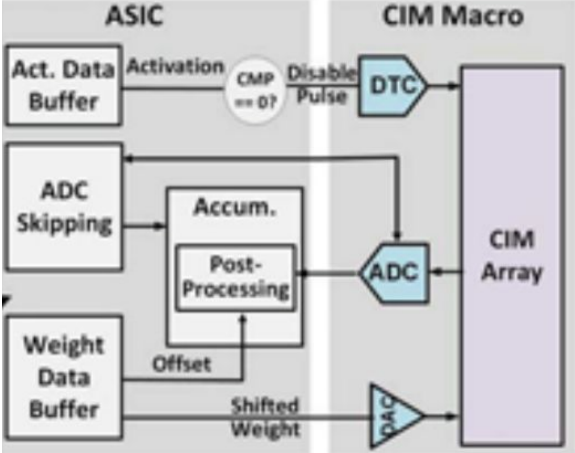
Design highlights

- Compute-in-memory (CIM) analog MAC
- 3T1C dynamic-analog-RAM (DARAM)
  - Density (4b per DARAM)
  - Retention
- 4b/4b or 8b/8b activation/weight data
  - Two DARAM cells combined for 8b/8b operation
- ADC skipping for power savings
  - Multiple MAC operations before A/D conversion



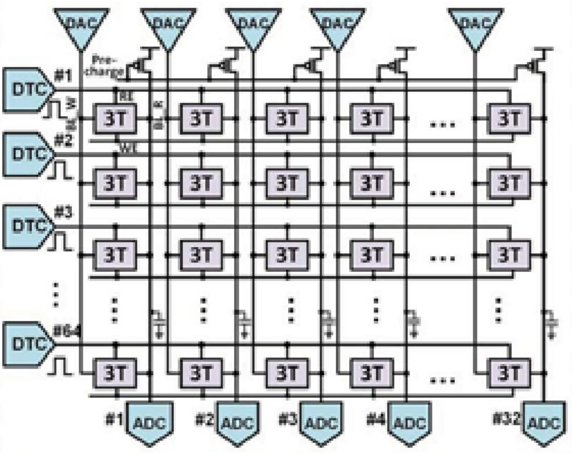
Block diagram

- Activation and weight data loaded from SRAM
- Activation data controls analog MAC through DTC
- Weight stored in DARAM of CIM array
- MAC result A/D converted for digital post-processing



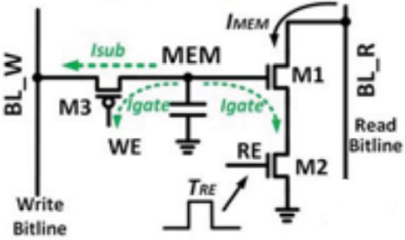
64x32 CIM array

- 64 rows, 32 columns (2048 DARAM)
- Each row has a 4b DTC
- Each column has 4b DAC, 5b ADC, MAC capacitor, precharge PMOS
- Each DARAM has WE, RE, BL_W, BL_R



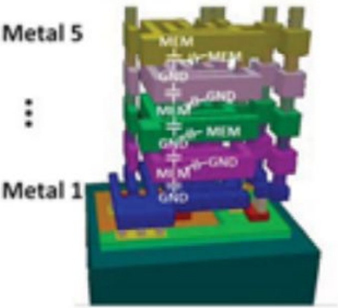
DARAM cell

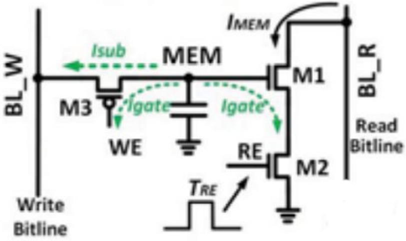
- 3T1C cell
  - Write access PMOS
  - Capacitor for storing 4-bit weight as analog voltage
  - NMOS buffer for reading data and performing analog MAC
- M1 W and L large to prevent mismatch related read errors

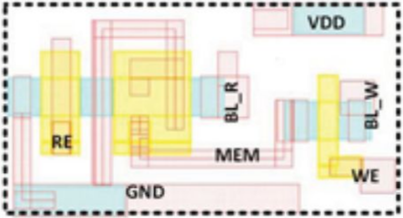


DARAM cell

- Capacitor on top of transistors to save area
- Special 3D capacitor interleaving GND and MEM nodes
  - 3x capacitance density



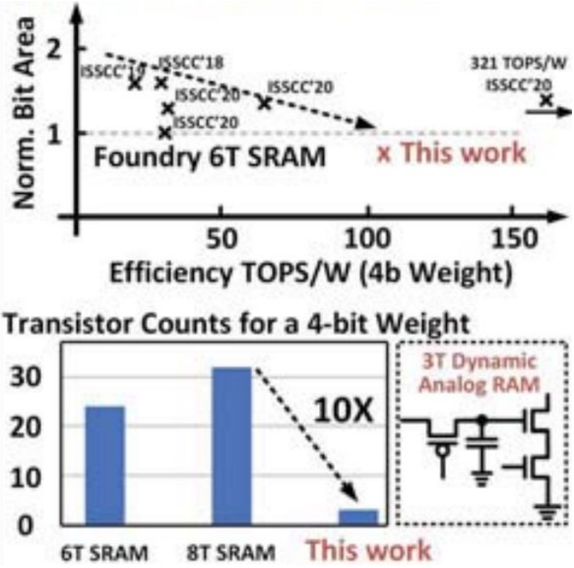




Layout of 3T Cell in This Work (only shown ME1, poly)

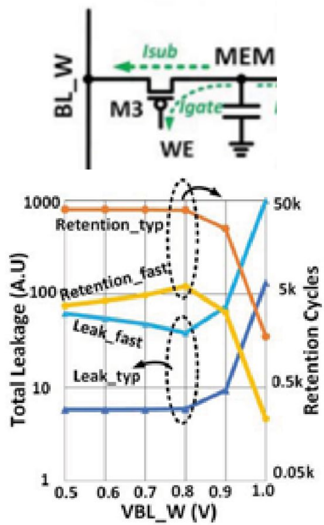
DARAM density

- 4-bit weight stored in one DARAM cell
- Effective 1-bit area is 75 % of foundry provided 6T SRAM area
- Transistor count greatly decreased from 6T-8T SRAM



Memory leakage

- Leakage Isub minimized by biasing BL_W with DAC when not writing
- 0.8 V results in 20x reduced leakage
- Retention time 5k-41k clock cycles
  - 5-40 images may be processed without refreshing memory
- Refresh every 5.5k-41k cycles ⇒ < 1.2 % throughput and < 0.4 % energy overhead
  - Refreshing takes 64 clock cycles (CIM array has 64 rows)

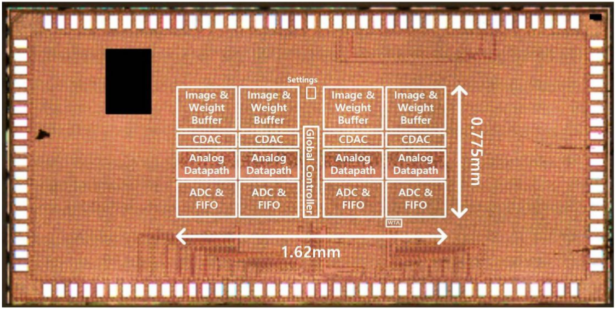


ARCHON: A 332.7TOPS/W 5b Variation-Tolerant Analog CNN Processor Featuring Analog Neuronal Computation Unit and Analog Memory

Jin-O Seo, Mingoo Seok, SeongHwan Cho
ISSCC 2022

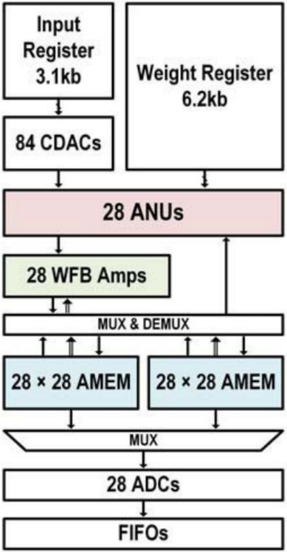
Design highlights

- Fully analog processing for efficiency
  - DACs, ADCs and input registers used only once per image ⇒ power savings
- Analog memory (AMEM) that stores computation results and passes them to next layer
- Tolerance to PVT variations
  - AMEM write with feedback



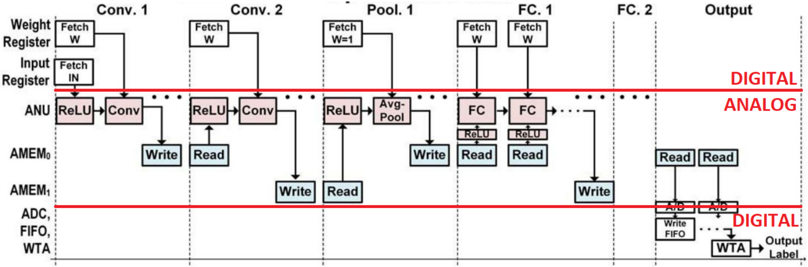
Block diagram

- Input data and weights stored in DFFs
- Input data (84 pixels) is converted to 84 voltages
- Analog neuronal computation unit (ANU) processes signal
  - Analog MAC, average pooling, ReLU, FC
- ANU result is written to AMEM
  - Two AMEMs, when one is writing the other is reading
- AMEM data is read back to ANU for next CNN layer
- Final result (classified output) is A/D converted



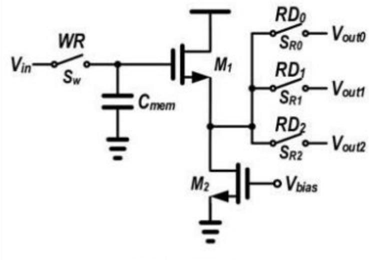
Analog operation

- All layers (convolution, pooling, fully connected) implemented in analog side
- Two AMEMs used in ping-pong manner

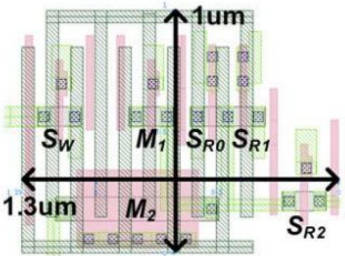


AMEM cell

- 6T1C analog memory cell
  - Write access transistor
  - 3 fF MOM capacitor storing ANU result (5b precision)
  - Source follower
  - Three read access transistors for 3x read speed
- Capacitor on top of transistors for increased density

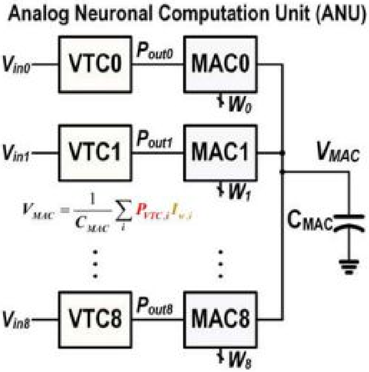


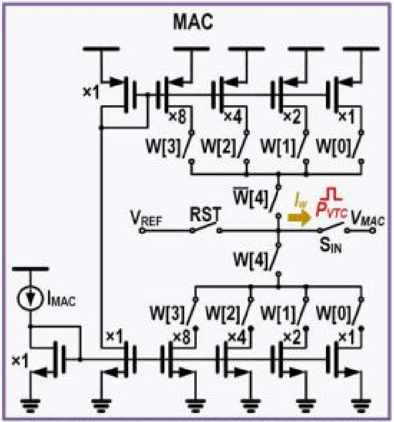
Schematic



Layout

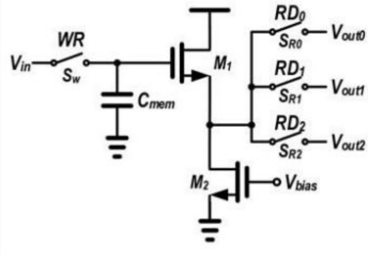
Analog MAC


Analog Neuronal Computation Unit (ANU)
V_{in0}
VTC0
P_{out0}
MAC0
W_0
V_{in1}
VTC1
P_{out1}
MAC1
W_1
V_{MAC}
$V_{MAC} = \frac{1}{C_{MAC}} \sum_i P_{VTC,i} I_{w,i}$
C_{MAC}
V_{in8}
VTC8
P_{out8}
MAC8
W_8


MAC
×1
×8
×4
×2
×1
W[3]
W[2]
W[1]
W[0]
$\overline{W[4]}$
I_w
P_{VTC}
V_{REF}
RST
V_{MAC}
S_{IN}
W[4]
I_{MAC}
W[3]
W[2]
W[1]
W[0]
×1
×8
×4
×2
×1
×1

AMEM potential problems

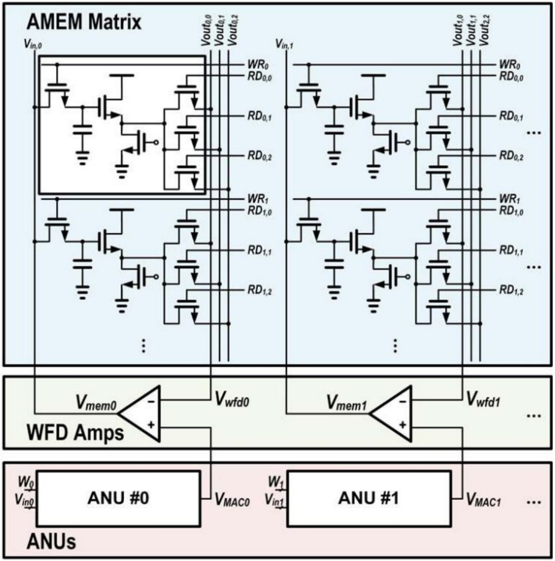
- Problem 1: leak current through write access transistor Sw
  - Changes capacitor voltage and degrades CNN performance
- Solution 1: this is not a problem
  - AMEM needs to hold the result only for a short time (computation of next layer) ⇒ low retention time requirement
- Problem 2: source follower sensitivity to PVT variations
  - AMEM output varies with threshold voltage of M1
- Solution 2: write with feedback (WFB)



Schematic

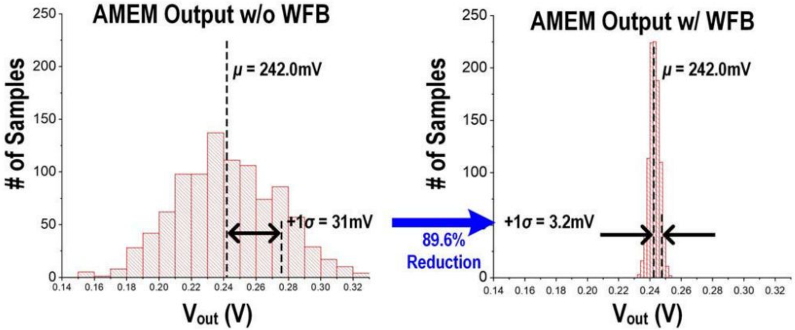
Write with feedback

- Write ANU result to AMEM using negative feedback
- WFB forces AMEM and ANU output voltages to be equal regardless of the source follower properties
- 28 amplifiers required



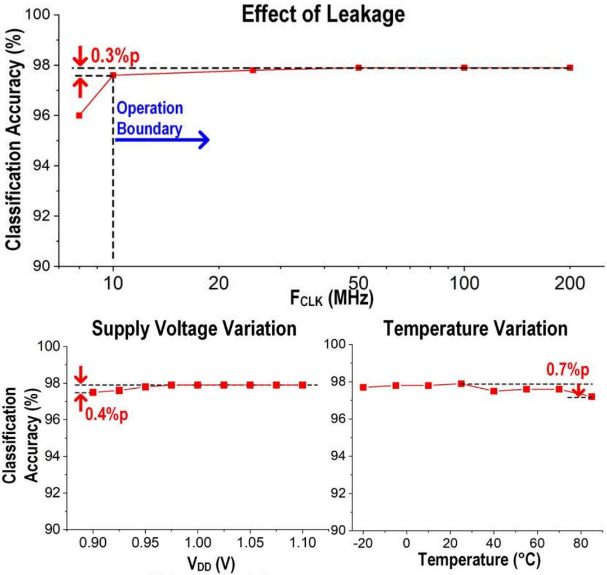
Write with feedback

- WFB reduces AMEM output variations by 90 %



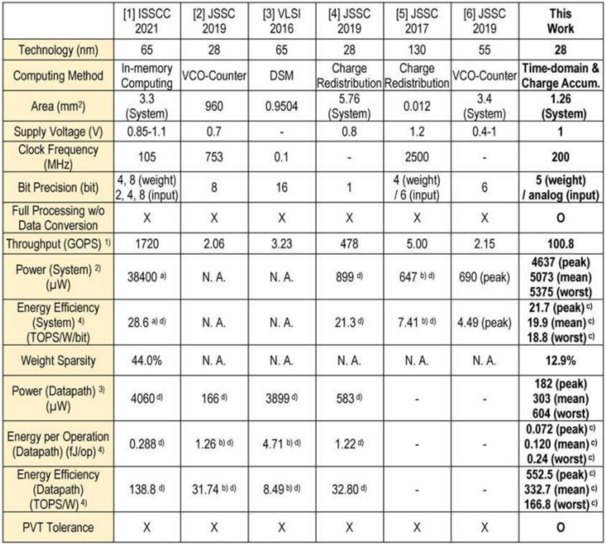
Leakage and PVT tolerance

- AMEM leakage severe under 10 MHz
- Excellent PVT tolerance



Performance table

- Column 1 shows paper 1 performance



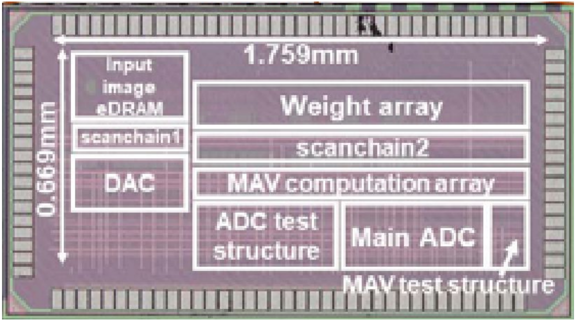
	[1] ISSCC 2021	[2] JSSC 2019	[3] VLSI 2016	[4] JSSC 2019	[5] JSSC 2017	[6] JSSC 2019	**This Work**
Technology (nm)	65	28	65	28	130	55	**28**
Computing Method	In-memory Computing	VCO-Counter	DSM	Charge Redistribution	Charge Redistribution	VCO-Counter	**Time-domain & Charge Accum.**
Area (mm^2)	3.3 (System)	960	0.9504	5.76 (System)	0.012	3.4 (System)	**1.26 (System)**
Supply Voltage (V)	0.85-1.1	0.7	-	0.8	1.2	0.4-1	**1**
Clock Frequency (MHz)	105	753	0.1	-	2500	-	**200**
Bit Precision (bit)	4, 8 (weight) 2, 4, 8 (input)	8	16	1	4 (weight) / 6 (input)	6	**5 (weight) / analog (input)**
Full Processing w/o Data Conversion	X	X	X	X	X	X	**O**
Throughput (GOPS) [1]	1720	2.06	3.23	478	5.00	2.15	**100.8**
Power (System) [2] (μW)	38400 [a]	N. A.	N. A.	899 [d]	647 [b][d]	690 (peak)	**4637 (peak) 5073 (mean) 5375 (worst)**
Energy Efficiency (System) [4] (TOPS/W/bit)	28.6 [a][d]	N. A.	N. A.	21.3 [d]	7.41 [b][d]	4.49 (peak)	**21.7 (peak) [c] 19.9 (mean) [c] 18.8 (worst) [c]**
Weight Sparsity	44.0%	N. A.	N. A.	N. A.	N. A.	N. A.	**12.9%**
Power (Datapath) [3] (μW)	4060 [d]	166 [d]	3899 [d]	583 [d]	-	-	**182 (peak) 303 (mean) 604 (worst)**
Energy per Operation (Datapath) (fJ/op) [4]	0.288 [d]	1.26 [b][d]	4.71 [b][d]	1.22 [d]	-	-	**0.072 (peak) [c] 0.120 (mean) [c] 0.24 (worst) [c]**
Energy Efficiency (Datapath) (TOPS/W) [4]	138.8 [d]	31.74 [b][d]	8.49 [b][d]	32.80 [d]	-	-	**552.5 (peak) [c] 332.7 (mean) [c] 166.8 (worst) [c]**
PVT Tolerance	X	X	X	X	X	X	**O**

eDRAM-CIM: Compute-In-Memory Design with Reconfigurable Embedded-Dynamic-Memory Array Realizing Adaptive Data Converters and Charge-Domain Computing

Shanshan Xie, Can Ni, Aseem Sayal, Pulkit Jain, Fatih Hamzaoglu, Jaydeep P. Kulkarni
ISSCC 2021

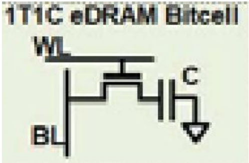
Design highlights

- Embedded dynamic random-access memory (eDRAM) cell that does everything
- CIM analog CNN operations



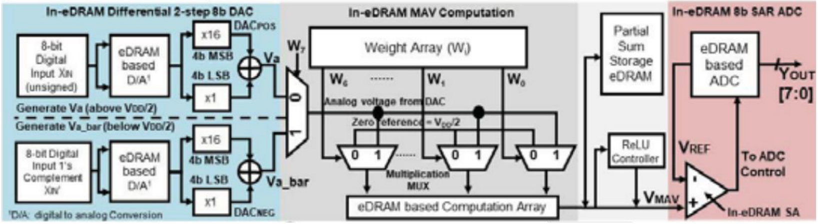
eDRAM cell

- 1T1C cell
  - Write/read access NMOS
  - 13 fF capacitor
- Used
  - as digital/analog memory that stores weight bits or computation results
  - to perform analog multiply-accumulate-average and other CNN operations
  - as a part of DAC
  - as a part of ADC
  - ...

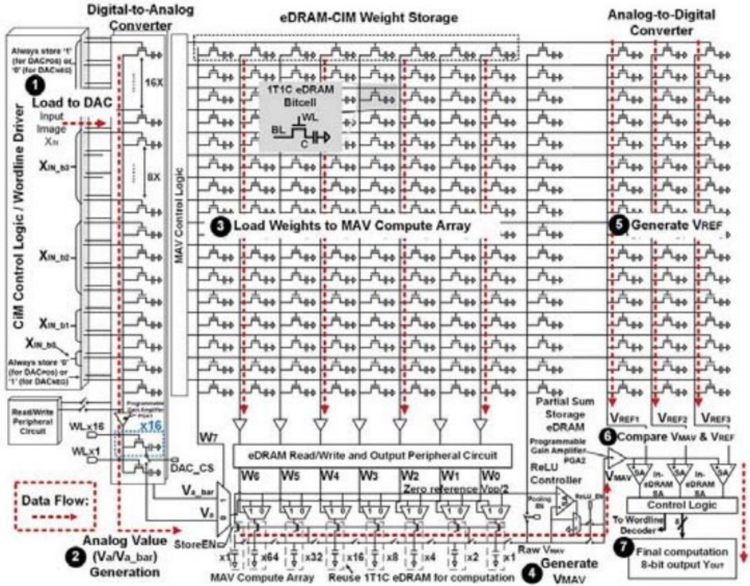


Block diagram

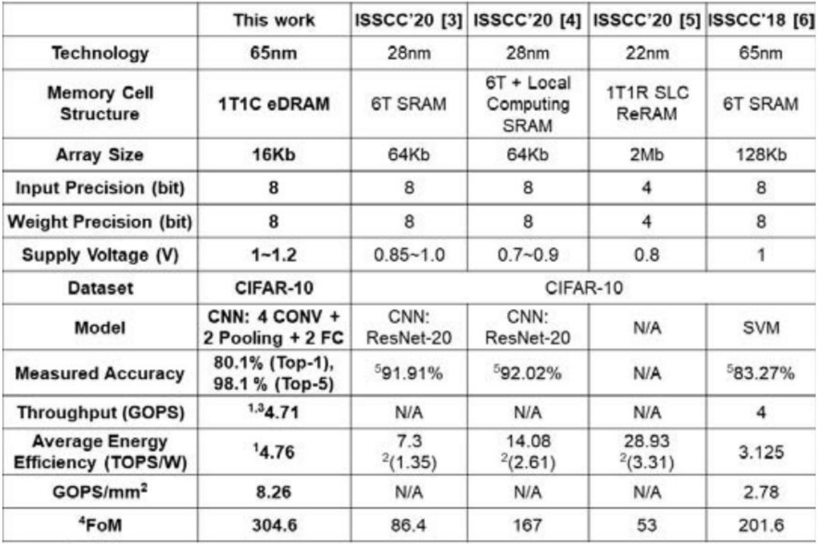
- Diff. eDRAM DAC transforms 8-bit image data to voltage
  - Two cycles, 4-bit per cycle, voltages stored in eDRAM
  - Charge redistribution between eDRAMs to get final result
- DAC output multiplied with 8b weight using 2:1 analog muxes
  - Weight bit selects between 0 and DAC voltage ⇒ multiplication
- MUX outputs saved to binary weighted eDRAM
  - Charge redistribution ⇒ accumulate-average
- Other CNN operations with eDRAM
- Final result converted to 8 bits by eDRAM SAR ADC



eDRAM array


Digital-to-Analog Converter
eDRAM-CIM Weight Storage
Analog-to-Digital Converter
Always store '1' (for DACPOS) or '0' (for DACNEG)
1 Load to DAC
Input Image XIN
CiM Control Logic / Wordline Driver
16X
8X
XIN_b3
XIN_b2
XIN_b1
XIN_b0
Always store '0' (for DACPOS) or '1' (for DACNEG)
MAV Control Logic
1T1C eDRAM Bitcell
WL
BL
C
3 Load Weights to MAV Compute Array
5 Generate VREF
Read/Write Peripheral Circuit
Programmable Gain Amplifier PGA1
WL x16
x16
WL x1
DAC_CS
W7
eDRAM Read/Write and Output Peripheral Circuit
W6
W5
W4
W3
W2
W1
W0
Zero reference VDD/2
Va_bar
Va
StoreEN
x1
x64
x32
x16
x8
x4
x2
x1
Data Flow:
2 Analog Value (Va/Va_bar) Generation
MAV Compute Array
Reuse 1T1C eDRAM for computation
Partial Sum Storage eDRAM
Programmable Gain Amplifier PGA2
ReLU Controller
Pooling EN
ReLU_EN
Raw VMAV
4 Generate VMAV
VREF1
VREF2
VREF3
6 Compare VMAV & VREF
VMAV
SA
In-eDRAM SA
Control Logic
To Wordline Decoder
8
7 Final computation 8-bit output YOUT

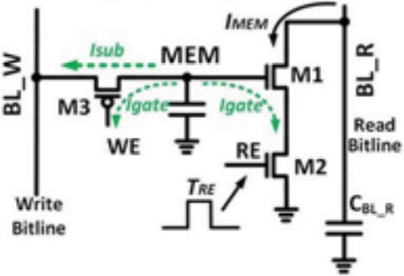
Performance



	This work	ISSCC'20 [3]	ISSCC'20 [4]	ISSCC'20 [5]	ISSCC'18 [6]
Technology	65nm	28nm	28nm	22nm	65nm
Memory Cell Structure	1T1C eDRAM	6T SRAM	6T + Local Computing SRAM	1T1R SLC ReRAM	6T SRAM
Array Size	16Kb	64Kb	64Kb	2Mb	128Kb
Input Precision (bit)	8	8	8	4	8
Weight Precision (bit)	8	8	8	4	8
Supply Voltage (V)	1~1.2	0.85~1.0	0.7~0.9	0.8	1
Dataset	CIFAR-10	CIFAR-10			
Model	CNN: 4 CONV + 2 Pooling + 2 FC	CNN: ResNet-20	CNN: ResNet-20	N/A	SVM
Measured Accuracy	80.1% (Top-1), 98.1 % (Top-5)	[5]91.91%	[5]92.02%	N/A	[5]83.27%
Throughput (GOPS)	[1,3]4.71	N/A	N/A	N/A	4
Average Energy Efficiency (TOPS/W)	[1]4.76	7.3 [2](1.35)	14.08 [2](2.61)	28.93 [2](3.31)	3.125
GOPS/mm^2	8.26	N/A	N/A	N/A	2.78
[4]FoM	304.6	86.4	167	53	201.6

Assignment

1. Compare analog and digital MAC. Why is analog better?
2. Derive the output voltage (voltage over C_{BL_R}) after MAC operation. The output capacitor is initally charged to VDD and it is shared by three DARAM circuits. Each DARAM has different weight stored in it (output currents I_{MEM1}, I_{MEM2}, I_{MEM3}) and each gets a different read enable pulse (T_{RE1}, T_{RE2}, T_{RE3}).


I_MEM
Isub
MEM
BL_W
M3
Igate
Igate
M1
BL_R
Read
Bitline
WE
RE
M2
T_RE
Write
Bitline
C_BL_R