DDPMnet: All-Digital Pulse Density-Based DNN Architecture with 228 Gate Equivalents/MAC Unit, 28-TOPS/W and 1.5-TOPS/mm2 in 40nm

DDPMnet: All-Digital Pulse Density-Based DNN Architecture with 228 Gate Equivalents/MAC Unit, 28-TOPS/W and 1.5-TOPS/mm² in 40nm

Animesh Gupta, Viveka Konandur*, Thoithoi Salam*, Saurabh Jain*, Orazio Aiello*, Paolo Crovetti**, Massimo Alioto***

*ECE, National University of Singapore, Singapore,

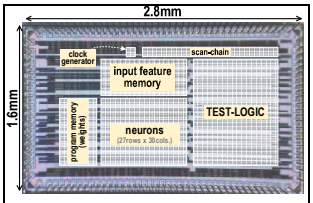
** DET, Politecnico di Torino, Italy

Relentless advances in DNN accelerator energy and area efficiency are demanded in low-cost edge devices [1]-[8]. Both directly benefit from the reduction in the complexity of MAC units (neurons), thanks to the reduction in area and energy of computations and the interconnect fabric. Unfortunately, such area and energy cost per neuron further increases in practical cases where flexibility is needed (e.g., precision scaling), ultimately limiting cost and power reductions. In this work, the all-digital DDPMnet architecture for DNN acceleration based on a pulse density data representation is introduced to reduce the gate count/MAC unit from the thousand range to few hundreds (Fig. 1). The proposed architecture removes any arithmetic block from MAC units (e.g., multipliers), while retaining the advantages of standard cell based design.

In this work, the dyadic digital pulse modulation [9] (DDPM) is exploited to perform MAC computations in the pulse density domain (Fig. 2). Each N-bit input feature X_i is represented by the pulse density $X_i/2^N$ over 2^N cycles, according to its DDPM modulation that translates X=(X[N-1], X[N-2]… X[0])$_2$ into the superposition of N pulse trains having binary-scaled density. For example, if the MSB X[N-1]=1, $2^N/2$ 1-pulses are generated in positions 0, 2, 4…2^N-2; if the next bit X[N-2]=1, $2^N/4$ 1-pulses are generated in positions 1, 5, 9…2^N-3, and so on. Weights are instead represented by digital pulse duration encoding (Fig. 2), with the generic value W_i being represented with a sequence of W_i/W_{LSB} subsequent 1-pulses (W_{LSB}= weight LSB) in sign-magnitude representation. The resulting pulse count $count_i$ of the DDPM-modulated input during the weight pulse duration is clearly proportional to the pulse density in the X_i stream (i.e., X_i itself) and the pulse duration of W_i (i.e., W_i itself), and hence represents their product $W_i \cdot X_i$. Extending the bitstreams to subsequent values of X_i and W_i, the final count adds the contributions $W_i \cdot X_i$ and thus inherently performs accumulation. This simplifies the MAC operation into an up/down counter to accumulate positive (up) or negative terms (Fig. 2). Weights are concatenated in time so that each kernel is computed in a 2^R-cycle window, and are normalized so that the maximum sum of kernel weights (i.e., duration of union of weight pulses) uses all 2^R cycles (Fig. 2, scaling involves also biases and outputs). ~100% utilization in non-max kernels is preserved via weight reordering/interleaving via the software scheduler in Fig. 3.

From Fig. 3, input features (weights) are distributed and shared by row (column) across the 27x30 MAC unit array, and stored in the input (program) memory. At any time, two of the three input banks are active, and the third stores the next patch of the input to sustain the acquisition of the input bitstream via three-way time interleaving. The scheduler pre-computes and stores 1) the resulting 6-bit ΔX and 4-bit ΔY displacement in the input memory, to cover the input video frame with the intended stride and kernel size, followed by the 27x2 DDPM modulator array, 2) the MAC unit 4-bit control instructions in the program memory, determined by the weight pulse duration and their pulse sequences (Fig. 3). Instructions are executed by the MAC units, which properly select incoming input features with the same order as weights via a MUX, count the resulting pulses, execute ReLU, and store the MAC output in a register (memory-mapped for output readout). The MUX enables the scheduler to reorder/interleave weights (and input features accordingly) for maximum MAC unit utilization. The software framework also handles the optimization of R, retraining and weight pulse normalization.

In DDPMnet, the MAC accuracy gracefully decreases at lower R (Fig. 4), while doubling throughput, area, and energy efficiency for every 1-bit reduction, due to the halved 2^R-cycle window duration. The approximation error is determined by the quantization in weight pulse normalization (Fig. 2), and the discrete nature of the density-based multiplication. The error is nearly image- and layer-independent, and is reduced via 1) retraining with DDPM computation in the loop, 2) bias subtraction of the mean residual error (evaluated across the training dataset, Fig. 4). Over a pre-trained network, this aggressively reduces R from 16 down to 10-12 at near-zero accuracy drop (<0.2%), improving throughput and energy efficiency by 25X (35X at 2% accuracy drop).



Die micrograph.

The average energy/area efficiency and runtime performance were evaluated using three datasets: MNIST (using LeNet-5), CIFAR-10 (using AlexNet) and ImageNet (using AlexNet), which achieve an average efficiency of 22.39 (1.22), 12.48 (0.84) and 9.77 TOPS/W (0.52 TOPS/mm²) at 0.5V and 35MHz (1.2V, 530MHz). The optimal R depends on the neural network, and its across-layer average is 6.81 for LeNet-5 on MNIST, 10.47 for AlexNet on CIFAR-10 and 11.4 on ImageNet. At 1.2V, 530MHz, DDPMnet takes an average execution time of 18.8ms (10.6ms) for Conv+FC (Conv only) layers on AlexNet and can provide an inference throughput of 41K inferences/s on LeNet-5, and 53.1 inferences/s on AlexNet.

From Fig. 5, the optimization of R in Fig. 4 brings the energy efficiency from 0.65TOPS/W (pre-trained weights) to 21.4TOPS/W (15.53 TOPS/W) under ~2% (<0.2%) accuracy loss. Also, on average the software scheduler improves the MAC unit array (memory capacity) utilization by 25.6% (29%), resulting in 20.2% better energy efficiency. As expected, more aggressive R reductions are allowed in layers with larger number of parameters/kernel (Fig 4). Similar considerations hold for fully-connected layers, whose vector multiplication nature expectedly reduces the energy efficiency by 25X compared to the best convolutional layer. Being the accuracy most sensitive to the first (Conv1) and last (FC3) layer, their R is set higher to allow aggressive scaling in the intermediate layers at minor overall accuracy degradation.

Fig. 6 shows other measurement results for the DDPMnet testchip in 40nm CMOS technology. The area of 4.48mm² includes a 14.12-KB latch memory, and the clock frequency ranges from 6MHz to 530MHz at a supply voltage from 0.4V to 1.2V. In terms of peak values, DDPMnet achieves a peak energy efficiency of 28.06 TOPS/W at 0.5V and 35MHz at no accuracy loss, and a peak throughput of 0.6 TOPS at 1.2V and 530MHz. The measured peak area efficiency is 1.55 TOPS/mm², and is 0.44 TOPS/mm² including all on-chip peripherals within the testchip (better amortized across the core array under larger array sizes).

Comparison with prior art (Fig. 6) shows that DDPMnet has the uncommon ability to scale the MAC precision without any additional reconfiguration logic in MAC units. This preserves the intended low gate count even under accuracy/complexity adjustment via R. In absolute terms, the peak energy efficiency under AlexNet is comparable to [1], [2], [5] in sub-10nm technologies and [4] in 12nm, and 1.9-27X better than prior time/pulse-domain accelerators [6]-[8]. Compared to the latter, the DDPMnet peak area efficiency is 152-6,285X better, which is still comparable to [4], [5] in 7-12nm, in view of its very compact MAC unit and efficient architecture. DDPMnet is expectedly disadvantaged by 6-11.9X compared to 5-7nm accelerators [1], [2]. Given the very broad range of CMOS technologies (8 generations), the technology-normalized comparison in Fig. 6 needs to be considered for fairness (see conservative scaling factors in Fig. 6). When the benefits attributed to the different technology are evened out, DDPMnet expectedly outperforms energy (area) efficiency of prior art by 1.7-20.6X (1.9-3,384X), offering a favorable tradeoff in terms of power and cost.

References:

[1] A. Agrawal et al., ISSCC 2021

[2] J. -S. Park et al., ISSCC 2021

[3] Z. Tan et al., CICC 2021

[4] K. Goetschalckx et al., VLSI Symposium 2021

[5] C.-H. Lin et al., ISSCC 2020

[6] A. Sayal et al., JSSC, July 2021

[7] A. Sayal et al. ISSCC 2019

[8] Y. Toyama et al., JSSC, Oct. 2019

[9] P. S. Crovetti, TCAS-I, Mar. 2017

978-1-6654-0756-4/22/$31.00 ©2022 IEEE

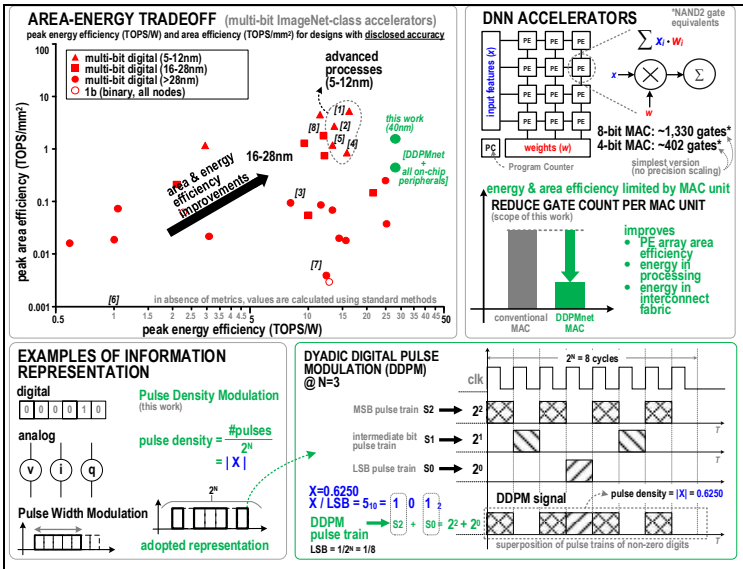


Fig. 1. Prior DNN implementations are limited by complex MAC units, which are simplified into simple counters in the proposed DDPMnet.

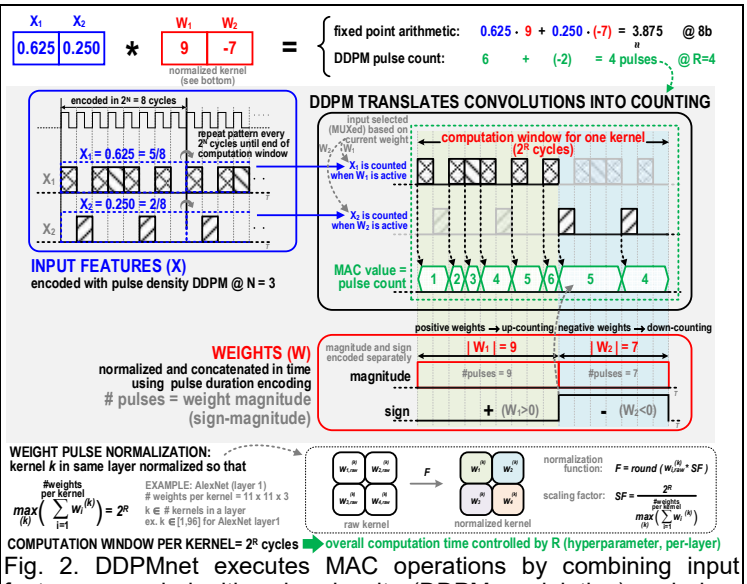


Fig. 2. DDPMnet executes MAC operations by combining input features encoded with pulse density (DDPM modulation) and sign-magnitude weights with pulse duration encoding.

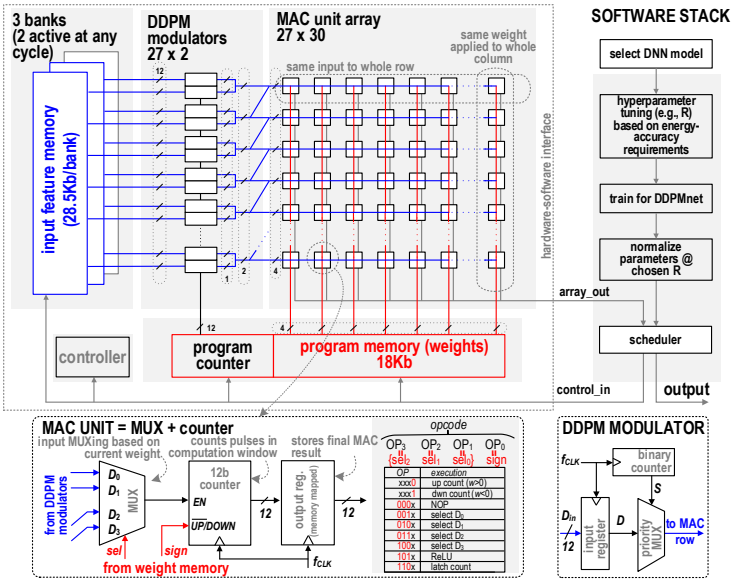


Fig. 3. DDPMnet architecture and underlying software stack.

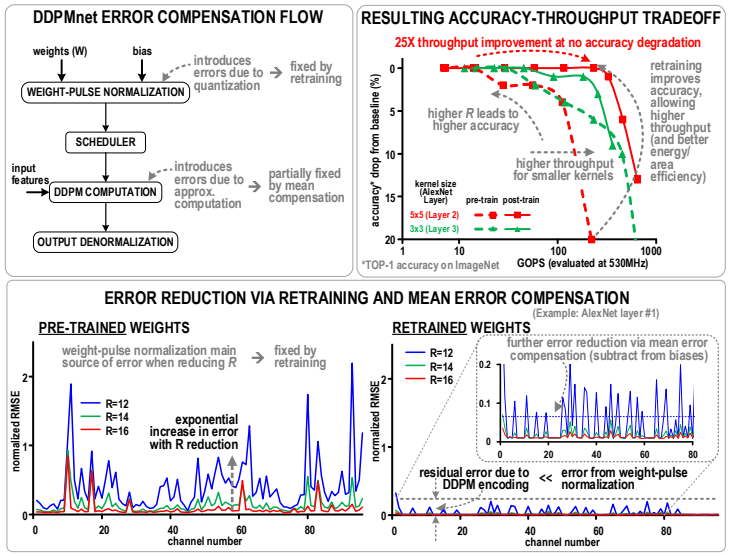
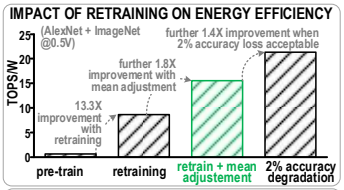
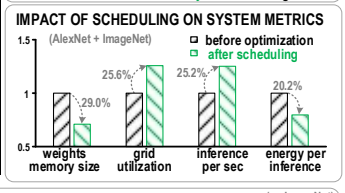


Fig. 4. Error compensation and accuracy recovery using retraining and accuracy-throughput tradeoff.

DDPMnet PERFORMANCE ACROSS DATASETS

dataset	MNIST	CIFAR10	ImageNet
network	LeNet-5	AlexNet*	AlexNet
layer type	CNN+FC	CNN+FC	CNN+FC
total operations	4.57MOP	1.44GOP	1.45GOP
avg. R across layers	6.81	10.47	11.43
accuracy loss	0	<0.1%	<0.2%
cycles/inference	12.7K	7.6M	9.9M
peak TOPS/W @0.5V, 35MHz	28.06	18.21	15.53
avg. TOPS/W @0.5V, 35MHz	22.39	12.48	9.77
peak TOPS/mm² @1.2V, 530MHz	1.55	1.08	0.84
avg. TOPS/mm² @1.2V, 530MHz	1.22	0.84	0.52
inferences/second @0.5V, 35MHz	2.7K	4.6	3.5
inferences/second @1.2V, 530MHz	41K	69.7	53.1





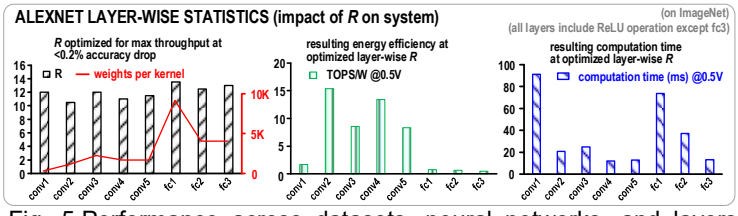


Fig. 5.Performance across datasets, neural networks, and layers along with the benefits of retraining and scheduling.

	This work	ISSCC'21 [1]	ISSCC'21 [2]	CICC'21 [3]	VLSI'21 [4]	ISSCC'20 [5]	JSSC'21 [6]	ISSCC'19 [7]	JSSC'19 [8]
	← recent DNN accelerators →						time/phase-domain accelerators		
technology	40nm	7nm	5nm	55nm	12nm	7nm	65	65	28
area (mm²)	1.36	19.6	5.46	13.65	4.5	3.04	1.74	0.124	0.004
MAC precision	unbounded (set by R)	FP32,16,8, INT 4,2	8b, 16b	4b	8b	8b	8b	1b	8b
precision scaling w/o extra MAC logic	YES	NO	NO	NO	NO	NO	NO	NO	YES
on-chip mem. (KB)	14.12	8192	3072	343	1780	2176	67	-	-
supply voltage (V)	0.4-1.2	0.55-0.75	0.55 -0.9	0.75-1	0.5-0.8	0.57-0.82	0.72-1.1	0.37-1.1	0.7
peak throughput (TOPS)	0.6	102.4	29.4	1.4	3.8	3.604	0.00013	0.00036	-
peak energy efficiency (TOPS/W)	15.53 [1] 28.06 [2]	16.25	13.6	8.1	15.8	13.32	1.044	12.8 (MNIST)	12.4 (MNIST)
peak area efficiency (TOPS/mm²)	1.55 [3] 0.44 [4]	5.22	2.69	0.102 [5]	0.84	1.185	0.00007	0.0029	1.57
tech-normalized (40nm) peak energy efficiency [6]	**15.53 [1] 28.06 [2]**	4.38	2.82	9.28	6.31	3.59	1.36	16.64 (MNIST)	9.53 (MNIST)
tech-normalized (40nm) peak area efficiency [6]	**1.55 [3] 0.44 [4]**	0.20	0.05	0.14	0.09	0.05	0.00013	0.0057	0.818

One addition + one multiplication = two operations. **1.** AlexNet + ImageNet @0.5V, 35MHz. **2.** LeNet-5 + MNIST @0.5V, 35MHz **3.** DDPMnet @1.2V, 530MHz. **4.** DDPMnet + all on-chip peripherals @1.2V, 530MHz. **5.** extrapolated from throughput, area. **6.** normalized to 40nm (scaling factors/gen conservatively set at: energy 0.75X/gen, area 0.625X/gen, throughput 1.2X/gen)

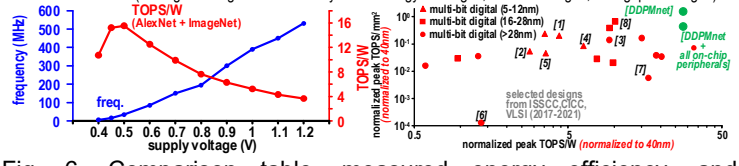


Fig. 6. Comparison table, measured energy efficiency, and technology-normalized TOPS/mm² vs TOPS/W.