Supplementary Materials for "Benchmarking Delay and Energy of Neural Inference Circuits"

Dmitri E. Nikonov and Ian A. Young
Components Research, Intel Corp., Hillsboro, Oregon 97007, USA
dmitri.e.nikonov@intel.com

1. Fundamentals and Concepts of Neuromorphic Computing

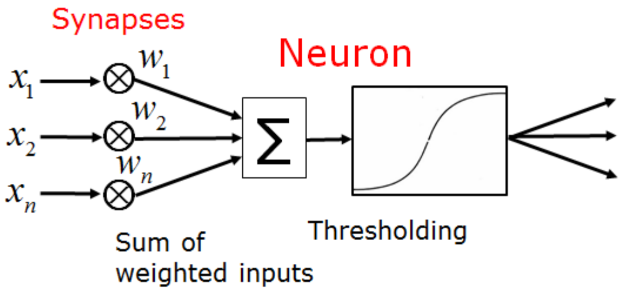


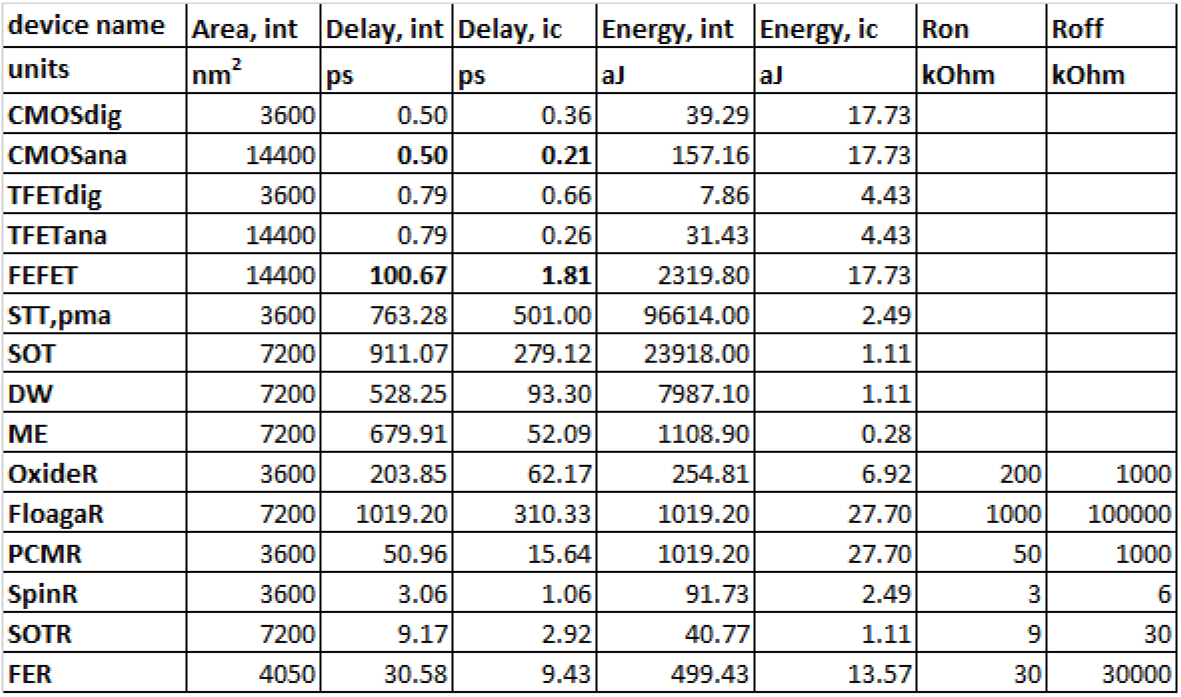
Figure 1. Scheme of a neural gate, perceptron.

2. Types of Neuromorphic Devices

Digital CMOS.

The first kind of digital NN is based on SRAM synapses that only provide a weight, while the multiplication and summation (MAC) operations are performed consecutively in the neuron [1]. The circuit considered here follows that in [2]: a synapse consists of n-bits of a SRAM register and state element; a neuron consists of two n-bit registers, an n-bit adder, *n* NAND gates, *n* inverters, and three *n*-state elements. Therefore area of the synapse and the neuron are the sums of the areas of the above constituent circuits. The delay and energy are mostly expended in the neuron, but some of the contributions are proportional to the number of synapses. Therefore such contributions are inserted to the equations for synapses below.

Table 1. PARAMETERS FOR DEVICES COMPRISING SYNAPSES AND NEURONS.



device name	Area, int	Delay, int	Delay, ic	Energy, int	Energy, ic	Ron	Roff
units	nm^2	ps	ps	aJ	aJ	kOhm	kOhm
CMOSdig	3600	0.50	0.36	39.29	17.73		
CMOSana	14400	**0.50**	**0.21**	157.16	17.73		
TFETdig	3600	0.79	0.66	7.86	4.43		
TFETana	14400	0.79	0.26	31.43	4.43		
FEFET	14400	**100.67**	**1.81**	2319.80	17.73		
STT,pma	3600	763.28	501.00	96614.00	2.49		
SOT	7200	911.07	279.12	23918.00	1.11		
DW	7200	528.25	93.30	7987.10	1.11		
ME	7200	679.91	52.09	1108.90	0.28		
OxideR	3600	203.85	62.17	254.81	6.92	200	1000
FloagaR	7200	1019.20	310.33	1019.20	27.70	1000	100000
PCMR	3600	50.96	15.64	1019.20	27.70	50	1000
SpinR	3600	3.06	1.06	91.73	2.49	3	6
SOTR	7200	9.17	2.92	40.77	1.11	9	30
FER	4050	30.58	9.43	499.43	13.57	30	30000

$$a_{syn} = n_b a_{reg} \qquad (1)$$

$$\tau_{syn} = 3\tau_{reg} + 4\tau_{se} + \tau_{nan} + \tau_{inv} + n_b \tau_1 \qquad (2)$$

$$E_{syn} = n_b \left(3E_{reg} + 4E_{se} + E_{nan} + E_{inv} + E_1 \right) \qquad (3)$$

$$a_{neu} = n_b \left(2a_{reg} + a_{inv} + a_{nan} + a_1 + a_{se} \right) \qquad (4)$$

$$\tau_{neu} = 2\tau_{reg} + 3\tau_{se} + \tau_{nan} + \tau_{inv} + n_b \tau_1 \qquad (5)$$

$$E_{neu} = n_b \left(2E_{reg} + 3E_{se} + E_{nan} + E_{inv} + E_1 \right) \qquad (6)$$

The performance estimate and parameters (**Error! Reference source not found.**) for a sense amplifier follow [3]. It is used as a part of a reading circuit for SRAM memories. The quantities per bit below are added to the corresponding neuron estimates. The transconductance and load capacitance of

$$a_{sa} = a_{inv1}\left(w_n + w_p + w_{iso} + w_n\right)/\, w_{dt} \qquad (7)$$

$$g_{msa} = g_{mdt}\left(w_p + w_n\right)/\, w_{dt} \qquad (8)$$

$$C_{lsa} = c_{tran}\left(w_p + w_n\right) \qquad (9)$$

$$\tau_{sa} = \log\left(V_{cc}/V_{sa}\right)c_{lsa}/\, g_{msa} + n_b\tau_1 \qquad (10)$$

$$E_{sa} = C_{lsa}V_{cc}^2 \qquad (11)$$

where the second term in the delay corresponds to the time to enable the sense amp and is proportional to the clock time.

The performance estimate and parameters (**Error! Reference source not found.**) for a voltage sense amplifier follow [3]. It is used as a part of a reading circuit for digital resistive memories. The quantities per bit below are added to the corresponding neuron estimates. It comprises 3 of n-type and 3 of p-type minimum width transistors.

$$a_{vsa} = 6a_{inv1} \qquad (12)$$

the pre-charge resistance, the sense input capacitance, and the bit line capacitance

$$R_{pch} = R_{ondt} \qquad (13)$$

$$C_{si} = 2c_{tran}w_{dt} \qquad (14)$$

$$C_{li} = s_{neu}c_{ic}l_{ic} \qquad (15)$$

$$\tau_{vsa} = 2.3R_{pch}C_{si} + V_{vsa}\left(C_{si} + C_{li}\right)/\left(V_{rvsa}/R_{on} - V_{rvsa}/R_{off}\right) + 2n_b\tau_1 \qquad (16)$$

$$E_{vsa} = C_{si}V_{cc}^2 \qquad (17)$$

Digital MAC.

Another kind of digital NN contains a multiplier and an adder in every synapse, so that the MAC operation is performed in the synapse [39]. The role of neurons is summation of partial results and application of the activation function.

$$a_{syn} = (n_b + 1) a_{add} + a_{se} \tag{18}$$

$$\tau_{syn} = \tau_{add} + \tau_{se} \tag{19}$$

$$E_{syn} = (n_b + 1) E_{add} / 2 + E_{se} \tag{20}$$

$$a_{neu} = a_{add} + 2a_{se} + n_b a_{ram} \tag{21}$$

$$\tau_{neu} = \tau_{add} + 2\tau_{se} + \tau_{ram} \tag{22}$$

$$E_{neu} = E_{add} + 2E_{se} + n_b E_{ram} \tag{23}$$

The factor n_b in energy and delay would correspond to simple ripple carry adders and multipliers based on them. More efficient designs based adders and multipliers (e.g. carry-save adders) are accounted by an additional factor of 1/2.

Analog CMOS.

We assume a cell similar to that in [4], where a neuron consists of an opamp, a current source, and a threshold function circuit; a synapse consist of 2 operational transconductance amplifiers (OTA), see also [5]. Transistors of various width are used, **Error! Reference source not found.**. The effective capacitance of the cell is dominated by the capacitance of the two OTAs

$$C_f = 4c_{tran} w_{out} \tag{24}$$

The subthreshold swing of a transistor is

$$SS = \frac{V_{sat}}{\log_{10}\left(i_{on} / i_{off}\right)}$$

The bias current is approximated as the geometric average of the on- and off-states:

$$I_b = \sqrt{i_{on} i_{off}}\, w_{in} \tag{25}$$

The transconductance of an OTA is

$$g_{mOTA} = \frac{I_b \ln 10}{SS} \frac{w_{out}}{w_{up}} \tag{26}$$

The output conductance of two OTAs is determined by

$$G_m = 2g_{mOTA} / w_{\max} \tag{27}$$

The effective resistance of the cell (with a factor of 2x for the nonlinearity of OTA and 2x to ensure output stability).

$$R_f = 4 / G_m \tag{28}$$

Then the opamp driving current is

$$I_{opamp} = V_{cc} / R_f \tag{29}$$

and the OTA current is

$$I_{OTA} = 2 I_b \frac{2 w_{sum}}{w_{\max}} \left(1 + \frac{w_{out}}{w_{up}} \right) \tag{30}$$

Thus benchmarks for the synapse and the neuron are

$$a_{syn} = 2 a_{inv4} (w_{in} + w_{in} + w_{in}) / w_{idt} \tag{31}$$

$$\tau_{syn} = 8.4 R_f C_f \tag{32}$$

$$P_{syn} = V_{cc} I_{OTA} \tag{33}$$

$$E_{syn} = P_{syn} \tau_{syn} \tag{34}$$

$$a_{syn} = 3 a_{inv4} (w_{in} + w_{in} + w_{in}) / w_{idt} \tag{35}$$

$$\tau_{neu} = \tau_{syn} \tag{36}$$

$$P_{neu} = V_{cc} \left(I_{opamp} + I_{OTA} \right) \tag{37}$$

$$E_{neu} = P_{neu} \tau_{neu} \tag{38}$$

where the area of standard cells in circuit is approximated as fan-out-4 inverters.

The performance estimate and parameters (**Error! Reference source not found.**) for an analog read circuit follow [3]. It is used as a part of a reading circuit for analog valued resistive memories. The quantities per analog cell below are added to the corresponding neuron estimates. It comprises several circuits equal in area to 32 standard inverter cells.

$$a_{adr} = 32 a_{inv1} \tag{39}$$

the column voltage is

$$V_{col} = V_{row} - V_{rvsa} \tag{40}$$

$$\tau_{adr} = \tau_{repu} + 2 n_b \tau_1 \tag{41}$$

$$P_{adr} = 25 V_{col}^2 / R_{ondt} \tag{42}$$

$$E_{adr} = P_{adr} \tau_{adr} \tag{43}$$

Analog spintronic and ferroelectric devices.

Both synapses and neurons consist of just one intrinsic device. Spintronic synapses and neurons have been proposed in [6], as well as ones based on magnetic tunnel junctions [7] or magnetoelectric switching [8]; see overview [9]. We assume the supply voltage to be 0.1V for all spintronic devices. Ferroelectric synapses were explored in [10,11].

These analog neurons and synapses have greater size, delay, and energy proportionally to the number of analog levels:

$$a_{syn} = n_l a_{dev} \quad (44)$$

$$\tau_{syn} = \tau_{dev} \quad (45)$$

$$E_{syn} = E_{dev} \quad (46)$$

$$a_{neu} = n_l a_{dev} \quad (47)$$

$$\tau_{neu} = n_l \tau_{dev} / 4 \quad (48)$$

$$E_{neu} = n_l E_{dev} \quad (49)$$

Resistive memories. We will use this term synonymously with ‘memristor’. Resistive elements are used here as analog memory with multiple levels of resistance in a single cell. Various types of resistive elements, such as oxide memristors [12,13], floating gate transistors (“flash”) [14,15], spintronic devices [16,17], have been proposed for neural networks.

In inference, the weights are not modified, therefore the characteristics of switching resistive memories are not relevant, but only their on and off resistances are. We assume characteristic on- and off-resistances in various resistive memory cells, **Error! Reference source not found.**. The parameters contributed from the memory cell per se are

$$I_{on} = V_{cc} / R_{on} \quad (50)$$

$$I_{off} = V_{cc} / R_{off} \quad (51)$$

$$a_{syn} = a_{dev} \quad (52)$$

The intrinsic capacitance of the synapse is of the order of that in a minimum interconnect, and the delay of synapses is determined by the upper bound of synapse resistance set at

$$R_{eff} = R_{on} \sqrt{n_l} \quad (53)$$

so

$$\tau_{syn} = 2.3 R_{eff} C_{ic} \quad (54)$$

$$E_{syn} = I_{on} V_{cc} \tau_{syn} \quad (55)$$

Another contribution comes from interconnects in the core and is described in Section 4.

3. Types of Neural Networks

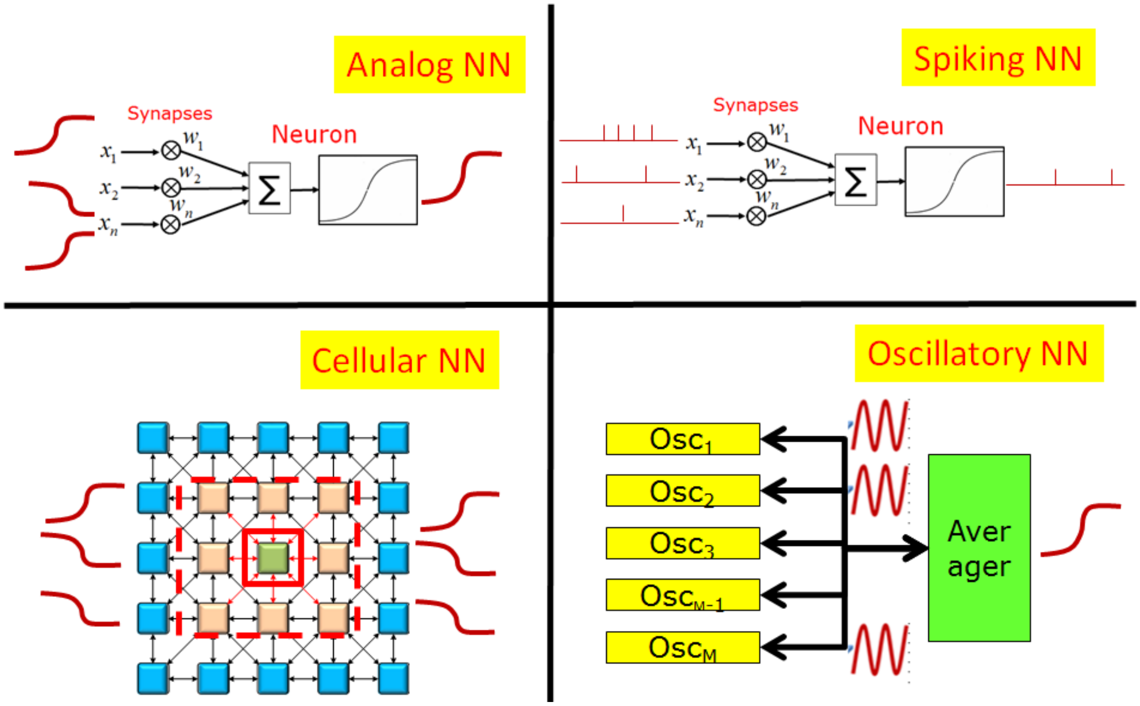


Figure 2. Schemes of the four types of neural networks considered in this paper.

Table 2. LABELS FOR DEVICES/ARCHITECTURE COMBINATIONS

Neuron	Synapse	A, C, S + …	ONN
Digital CMOS	Digital CMOS 6T SRAM	CSd	
Digital CMOS	Digital CMOS MAC	CMd	
Digital CMOS	Oxide memristor digital	COd	
Digital TFET	Digital TFET MAC	TMd	
Digital CMOS	FEFET digital	CFd	
Digital CMOS	Spin-transfer torque digital	CJd	OSTT
Digital CMOS	Spin-orbit digital	CHd	OSOT
Analog CMOS	Analog CMOS	CCa	OCr
Analog TFET	Analog TFET	TTa	OTr
Analog CMOS	Ferroelectric FET	CFa	OPz
Analog CMOS	Oxide memristor	COa	OOx
Analog CMOS	Floating gate	CGa	
Analog CMOS	PCM	CPa	
Ferroelectric FET	Ferroelectric FET	FFa	
Domain wall	Domain wall	WWa	
Spin-orbit torque	Spin-orbit analog	HHa	
Magnetoelectric	Magnetoelectric	EEa	OME

ANN.

This is the default case, we directly use the estimates for the synapses and neurons obtained in the previous section.

CeNN.

We follow the treatment of cellular neural networks in [4]. Application of CeNN to CoNN was considered in [18]. Due to both feedback and feedforward connections in a CeNN and due to more connections than just nearest neighbors, the number of synapses is doubled. Also it takes a longer time for CoNN networks to settle to the steady state due to a larger number of connections [4]. This delay depends on the input patterns; we take estimated average values. Therefore

$$a_{syn} = M_{syncnn} a_{syn,ann} \tag{56}$$

$$\tau_{syn} = M_{stepcnn} M_{syncnn} \tau_{syn,ann} \tag{57}$$

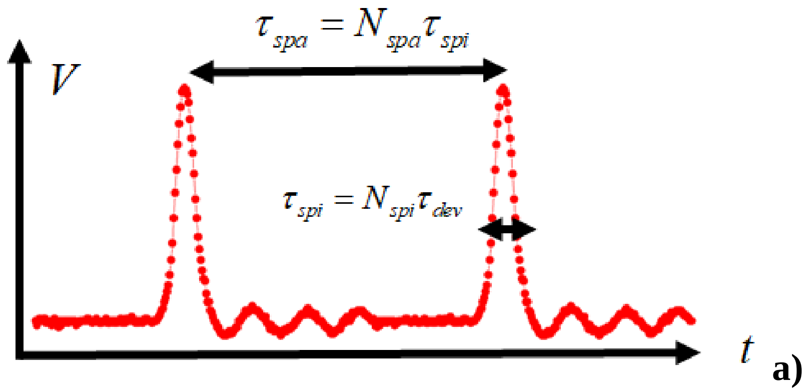
$$E_{syn} = M_{stepcnn} M_{syncnn} E_{syn,ann} \tag{58}$$

$$a_{neu} = a_{neu,ann} \tag{59}$$

$$\tau_{neu} = M_{stepcnn} \tau_{neu,ann} \tag{60}$$

$$E_{neu} = M_{stepcnn} E_{neu,ann} \tag{61}$$

Neural network parameters related to Hebbian learning are based on the synaptic weight information: the maximum weight value obtained from the training weights, and the average summation of the weights per cellular cell [4].


$\tau_{spa} = N_{spa}\tau_{spi}$
V
$\tau_{spi} = N_{spi}\tau_{dev}$
t
a)

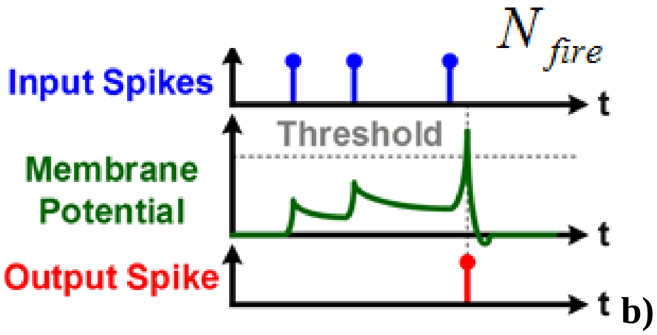


Figure 3. Approximate wave forms in a spiking neural network. a) The spike separation is longer than the spike duration; b) Multiple synapse spikes are required for a neuron to fire [37].

SNN.

We introduce a factors (**Error! Reference source not found.**) relating the spike duration to the device delay and relating the time spacing between spikes to the spike duration, Figure 3.

With these factors the estimates for SNN become

$$\tau_{syn} = \tau_{syn,ann} N_{spi} N_{spa} \quad (62)$$

$$E_{syn} = E_{syn,ann} N_{spi} \quad (63)$$

$$\tau_{neu} = \tau_{neu,ann} N_{spi} N_{spa} N_{fire} \quad (64)$$

$$E_{neu} = E_{neu,ann} N_{spi} N_{fire} \quad \text{(rate coded)} \quad (65)$$

$$E_{neu} = E_{neu,ann} N_{spi} \quad \text{(temporal coded)} \quad (66)$$

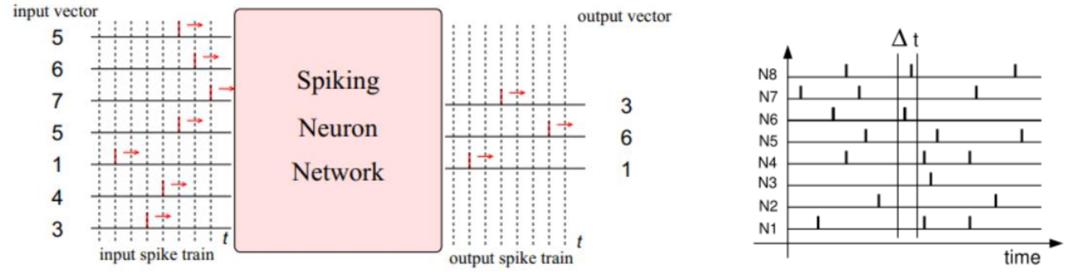


Figure 4. Two types of spiking NN: rate coded and temporal coded.

Note that it takes a different number of spikes arriving at a neuron from synapses to make it fire for the cases of rate coding or temporal coding of the signal, Figure 4. We also account for the spiking activity, i.e., the probability of a synapse producing a spike in a given spiking interval. We incorporate an empirical trend that the spiking activity decreases in the later stages where spike activity in an SNN decreases by $r_a = 1/n_{stage}$ with stage number in a DNN or CoNN [19].

ONN.

For the oscillator neural networks (ONN) we consider the frequency-shift keying (FSK) approach [20]. The way convolution operations are implemented in ONN follows the approach [21] of coupling oscillators to a common node, 'averager', and identifying the envelope of the signal there as the measure of the convolution. In this approach, an oscillator plays the role of a synapse and the averager with the envelope detector – the role of a neuron.

The area of oscillators is typically larger because they contain multiple instances of simple gates, e.g. several inverters in the CMOS ring oscillator. The area of the averager and the peak detector is bound to be even larger.

$$a_{syn} = 10a_{syn,ann} \tag{67}$$

$$a_{neu} = 30a_{neu,ann} \tag{68}$$

The frequency of transistor-based ring oscillators is determined by the product of the number of inverters (chosen here to be 5) and an average delay in an inverter. The period of oscillation in this case is equal to 10 stage delays. The average power is proportional to that of a logic device.

$$f_{osc} = 0.1/\tau_{inv4} \quad \text{(for transistor oscillators)} \tag{69}$$

$$P_{osc} = 3E_{\text{int}} / \tau_{\text{int}} \quad \text{(for transistor oscillators)} \tag{70}$$

The frequency of spintronic oscillators empirically proves to be several times faster than the inverse switching time of a nanomagnet. By the comparison of typical values from micromagnetic simulations or experimental reports, we arrive at the following proportionality constants:

$$f_{osc} = 6 / \tau_{neu,ann} \quad \text{(for spintronic oscillators)} \tag{71}$$

$$P_{osc} = 6E_{neu,ann} / \tau_{neu,ann} \quad \text{(for spintronic oscillators)} \tag{72}$$

$$f_{osc} = 1/\tau_{neu,ann} \quad \text{(for piezo oscillators)} \tag{73}$$

$$P_{osc} = 3E_{neu,ann} / \tau_{neu,ann} \quad \text{(for piezo oscillators)} \tag{74}$$

The operation of the ONN synapse is limited by the synchronization time of the oscillators which takes several periods of oscillations**Error! Reference source not found.**. Thus the ONN benchmarks are

$$\tau_{syn} = N_{synch} / f_{osc} \tag{75}$$

$$E_{syn} = P_{osc}\tau_{syn} \tag{76}$$

$$\tau_{neu} = \tau_{syn} \tag{77}$$

$$E_{neu} = E_{syn} \tag{78}$$

4. Treatment of interconnects

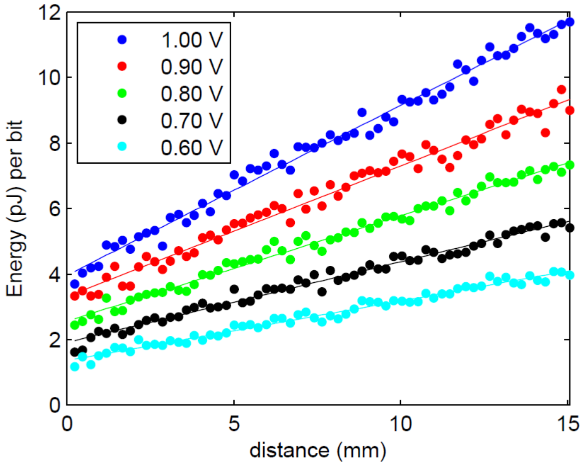


Figure 5. Energy per bit vs. distance in TrueNorth [22].

A geometry calculation with a low-k interlayer dielectric results in $c_{ic} = 10^{-10} F/m$ for 20nm wire width. We use this value for shorter interconnects shorter than 0.1mm. Energy and capacitance vs. distance for an actual NN chip, TrueNorth [22], is shown in Figure 5. With voltage of 1V, the energy of a spike is 8pJ for 15mm of interconnect length, which implies that $c_{ic} = 5 \cdot 10^{-10} F/m$. This energy dissipation in the interconnect incorporates routers, drivers, and repeaters. Therefore the energy to transmit a bit over an interconnect in neural networks is less efficient by the factor of 5 than the energy of the ideal case, i.e., just charging the interconnect capacitance. This empirical factor of 5 is incorporated into estimates for interconnect longer than 0.1mm.

The delay in a core-wide interconnect is dominated by the RC-delay in wires connecting synapses and neurons:

$$\tau_{cic} = \left(0.38 R_{ic} C_{ic} + R_{eff} C_{ic} + R_{ic} C_{load}\right) l / l_{ic} \,. \qquad (79)$$

The delay of charging a global, chip-wide interconnect

$$\tau_{gic} = \frac{c_{ic} l V}{I_{neu}} = \frac{E_{ic}}{I_{neu} V} \,. \qquad (80)$$

The delay and energy of a core-wide interconnect are added to those of a synapse. The energy and delay of a chip-wide interconnect are added to those of a neuron.

5. Neuromorphic computing workloads vs. hardware

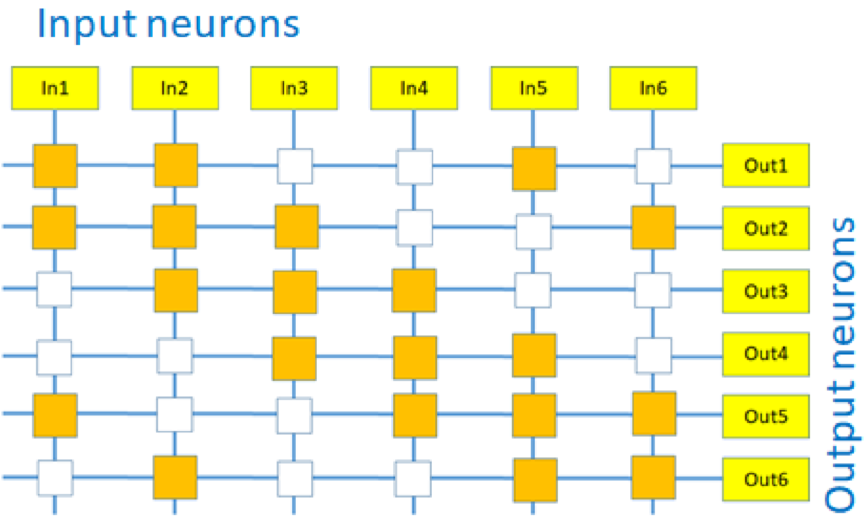


Figure 6. Cross-connect topology for the neural network. Input ('In') and output ('Out') neurons are shown in yellow. Active synapses are shown in orange, and unused synapses in white.

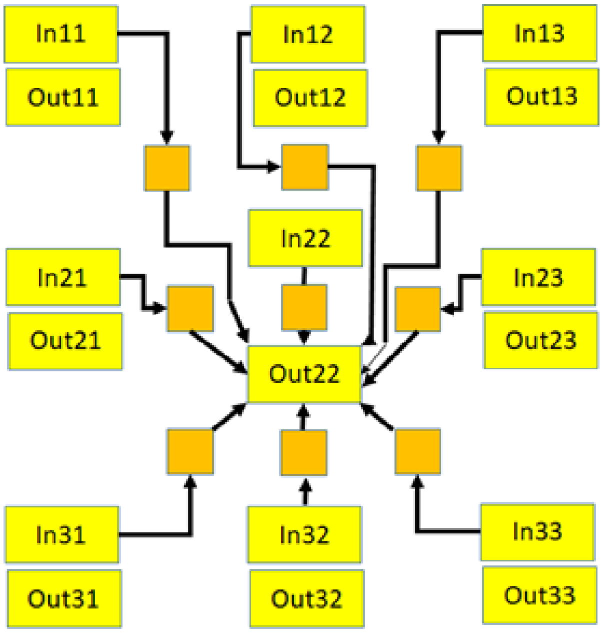


Figure 7. Convolution topology for the neural network. Input ('In') and output ('Out') neurons are shown in yellow. Active synapses are shown in orange.

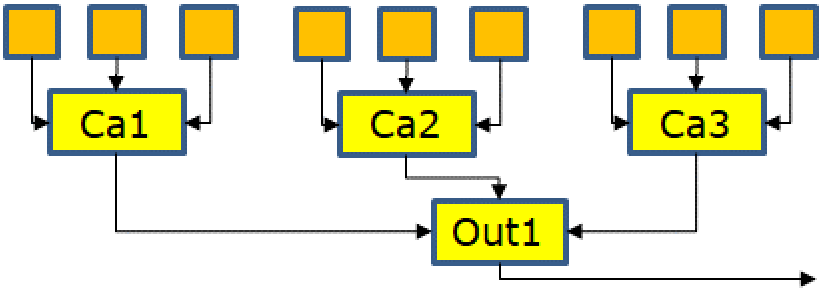


Figure 8. Synapses connecting to the output neuron via cascaded neurons in case of a limited fan-in.

6. Prototype neuromorphic chips

Table 3. PARAMETERS FOR NEUROMORPHIC CHIPS

Chip Name	Main Affiliation	Year	# cores	Neurons per core	Synapses per neuron	Area, mm^2	Power, mW	Syn Throughput, MSOPS	Energy syn event, pJ	Syn fire rate, s^{-1}	Activity	Process, nm	Voltage, V	References
Notation			c_{ch}	n_{cor}	s_{neu}	a_{ch}	P_{ch}	T_{syn}	E_{spi}	f_{syn}	r_a			
HICANN	Heidelberg	2010	1	512	224	50	1150*	11,500	100	100k	1	180	1.8	[23]
HICANN-X	Heidelberg	2018	1	512	256	32	2100*	2600	800	20k	1	65	1.2	[24]
SyNAPSE	HRL	2013	1	576	128	42	130	15	8700	203*	1	90	1.4	[25]
SpiNNaker	Manchester	2013	16	1024	1024	102	1000	64	16k*	10	0.4*	130	1.2	[26][27]
SpiNNaker 2	Manchester	2017	64	2048	1024	?	110	250	440	10	0.2*	28	1.0	[28]
True North	IBM	2014	4096	256	256	430	72	3000	26	20	0.5	28	0.78	[22][29]
Neurogrid	Stanford	2014	1	65536	1024	168	59*	62.5	941	10	0.09*	180	1.8	[30]
IFAT	UCSD	2014	32	2048	1024	16	1.57	73	22	10	0.11*	90	1.2	[31]
ROLLS	ETH	2015	1	256	512	51.4	4	4	1000*	30	1	180	1.8	[32][33]
DYNAP-SEL	ETH	2016	4	256	64	43.8	?	?	50	30	?	28	1.0	[34]
Loihi	Intel	2018	128	1024	128	60	450	30,000	15*	1800*	1	14	0.75	[35][36]
SBNN	Intel	2018	64	64	256	1.72	209	25,200	8.3	50k	0.5*	10	0.53	[37]

* derived value

We compared our performance estimates with experimentally measured [38] for the speech recognition workload on the Loihi and Mydiad2 (Movidius) chips. This table is comparing our estimates with experimental results obtained in particular chips. Our estimates relate to the minimal circuit needed to perform the computing in a certain neural network. We have no visibility into how the algorithm was compiled to utilize circuits of a particular chip or what the overheads of such implementations were. The purpose of the comparison was to see the general trends for delay and energy. We note that the theoretical estimates are much more optimistic than experimental. The reasons for the discrepancy could be the circuit overhead required in an actual chip such as stand-by power, need to fetch the data, slower clock frequency, etc.

Table 4. COMPARISON OF BENCHMARKS WITH MEASURED PERFORMANCE

	Loihi [38]	Loihi this work	Movidius [38]	Movidius this work
Speed, inference/s	89.8	55k	300	167k
Energy, μJ/inference	770	6	1500	5.5

7. Digital Neural Accelerators

Table 5. PARAMETERS FOR DIGITAL NEURAL ACCELERATORS

Chip Name	**Main Affiliation**	**Year**	**# cores**	**Neurons per core**	**Synapses per neuron**	**Memory Bytes**	**Area, mm²**	**Power, W**	**Performance, GMAC/s**	**Synapse energy, pJ**	**Clock frequency, MHz**	**Process, nm**	**References**
Notation			c_{ch}	n_{cor}	s_{neu}	m_{ch}	a_{ch}	P_{ch}	T_{syn}	E_{syn}	f_{cl}		
Diannao	CAofS	2014	1	16	16	2k	3.02	0.485	452	1.1*	980	65	[39]
Dadiannao	CAofS	2014	16	16	16	32M	67.73	15.97	5585	2.9*	606	28	[40]
Pudiannao	CAofS	2015	1	16	16	32k	3.51	0.596	1056	0.56*	1000	65	[41]
Shidiannao	CAofS	2015	1	16	16	36k	4.86	0.32	194	1.7*	1000	65	[42]
Eyeriss	MIT	2016	1	1	168	192k	12.25	0.278	33.6	8.3*	200	65	[43]
EIE	Stanford	2016	1	64	8	10.3M	40.8	0.579	51.2	11.3*	800	45	[44]
Origami	ETH	2016	1	4	49	43k	3.09	0.654	98	6.7*	500	65	[45][46]
Envision	Leuven	2017	1	16	16	128k	1.87	0.044	51	0.86*	200	28	[47]
TPU	Google	2017	1	256	256	28M	300	40	11400	3.5*	700	28	[48]
Tesla	Nvidia	2017	80	32	32	6M	815	300	14900	20*	1300	12	[49]
DPU	Wave	2018	16384	1	1	24M	400	200	3900	51*	6700	16	[49]
Q4MobilEye	Intel	2018	1	32	32	1M	?	3	1078	2.8*	1000	28	[49]
Parker	Nvidia	2016	1	256	256	4M	?	5	375	13.3*	3000	16	[49]
S32V234	NXP	2017	1	64	64	4M	?	5	512	9.8*	1000	28	[49]
Myriad 2	Intel	2017	12	4	16	2M	27	1.5	58	26*	800	28	[50]

* derived value; ** 'CAofS' designates the Chinese Academy of Sciences.

8. Supplementary Plots

Remaining benchmarking plots are collected here in order to keep the main text concise.

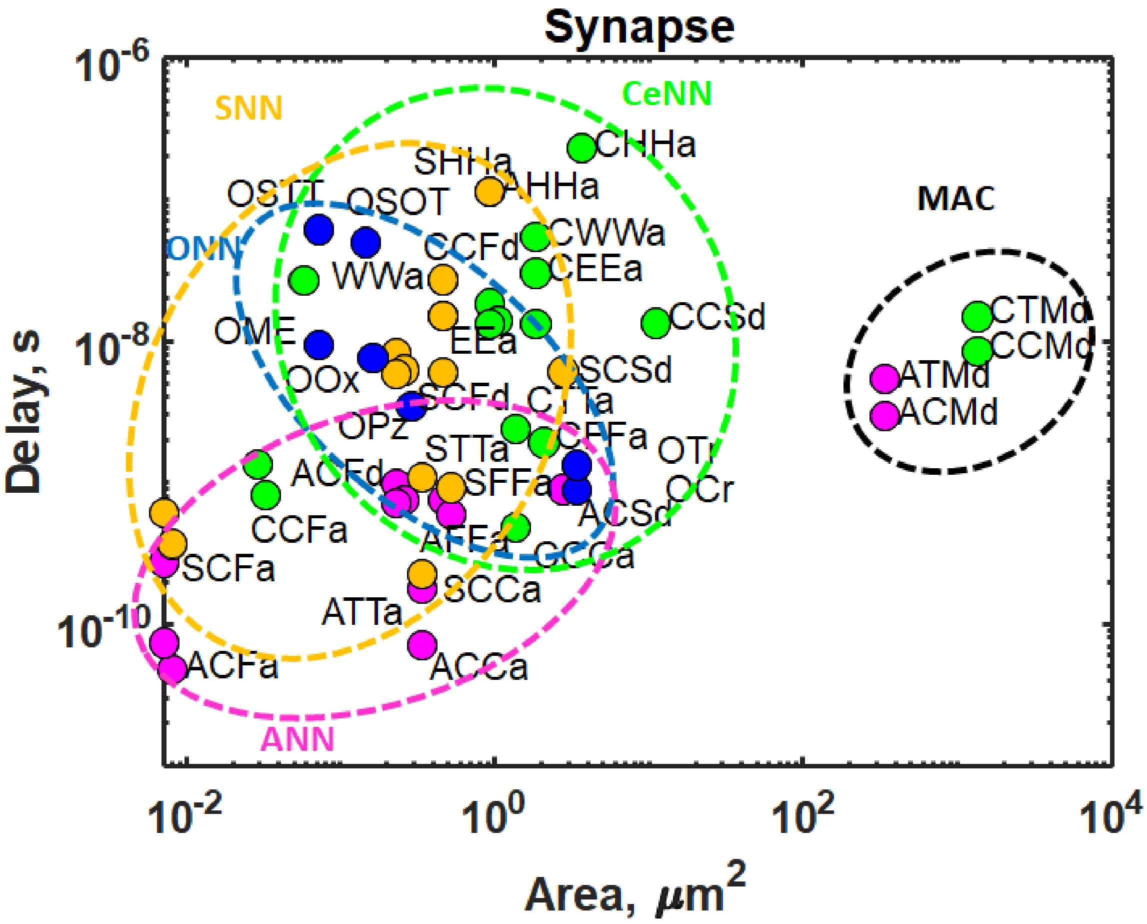


Figure 9. Delay vs. area for synapses.

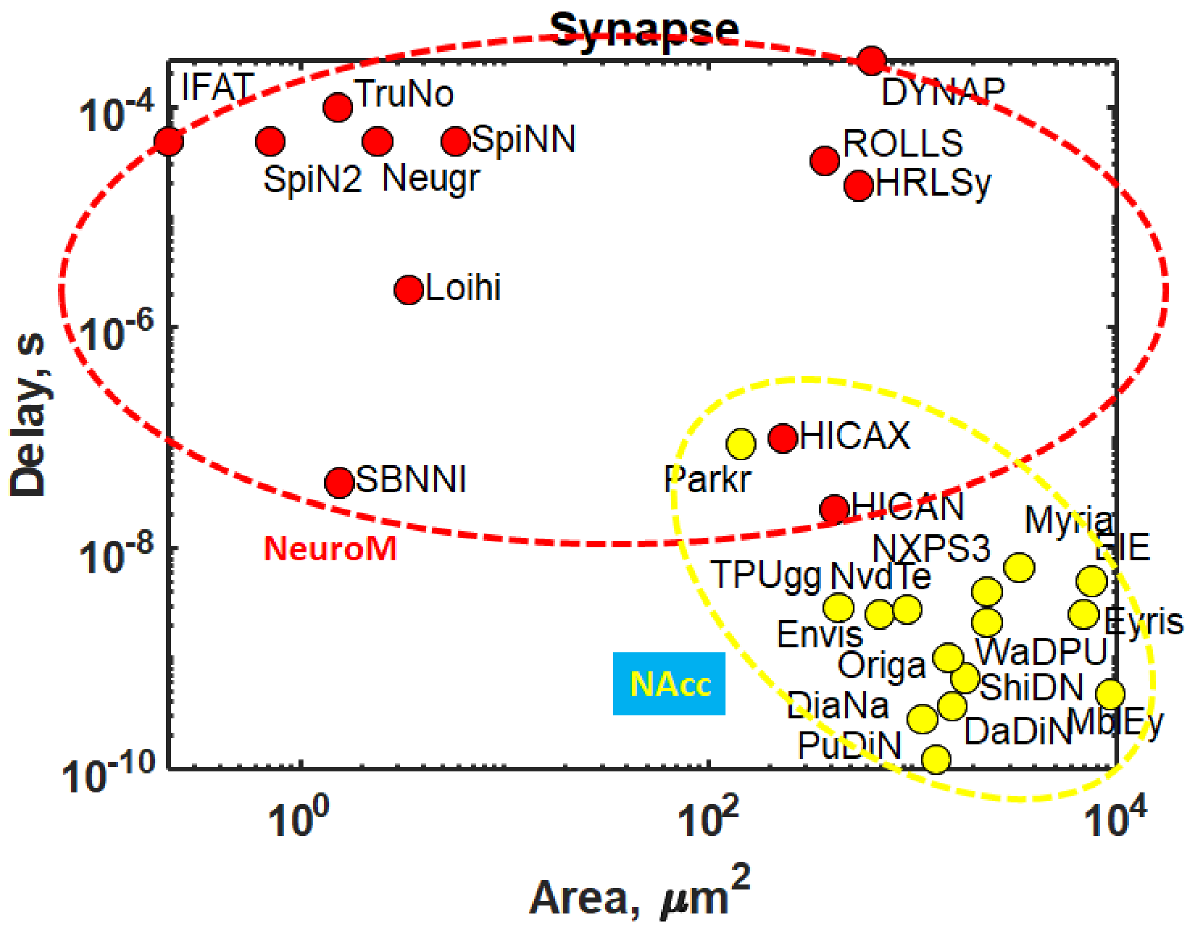


Figure 10. Delay vs. area for synapses.

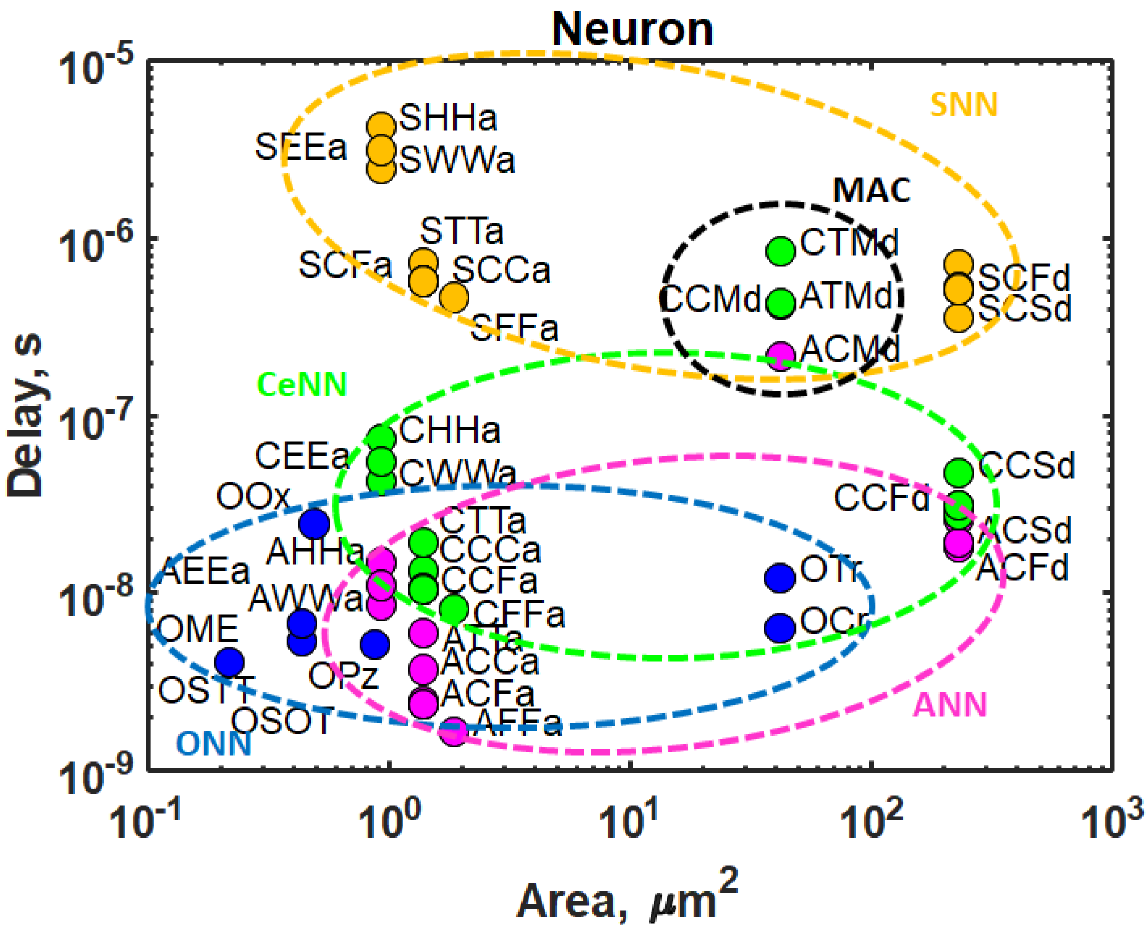


Figure 11. Delay vs. area for neurons.

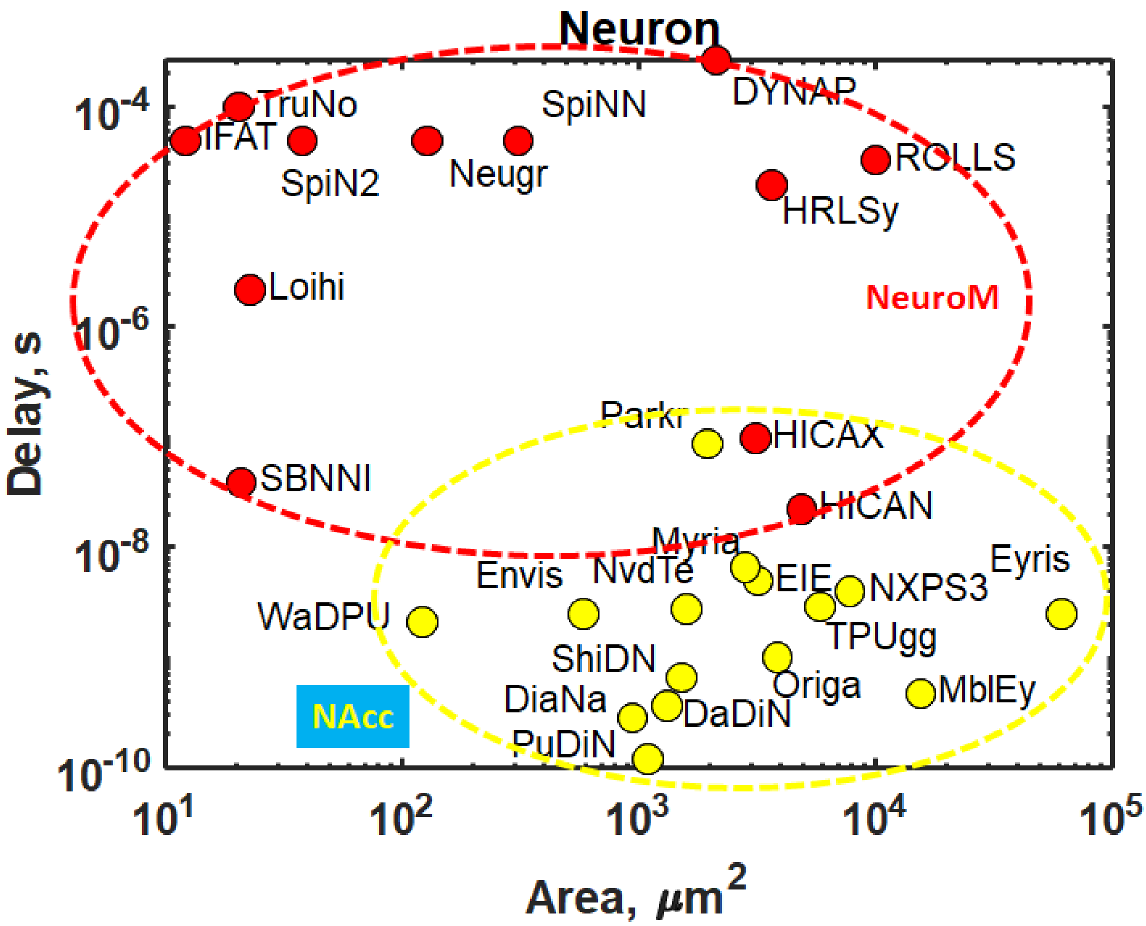


Figure 12. Delay vs. area for neurons.

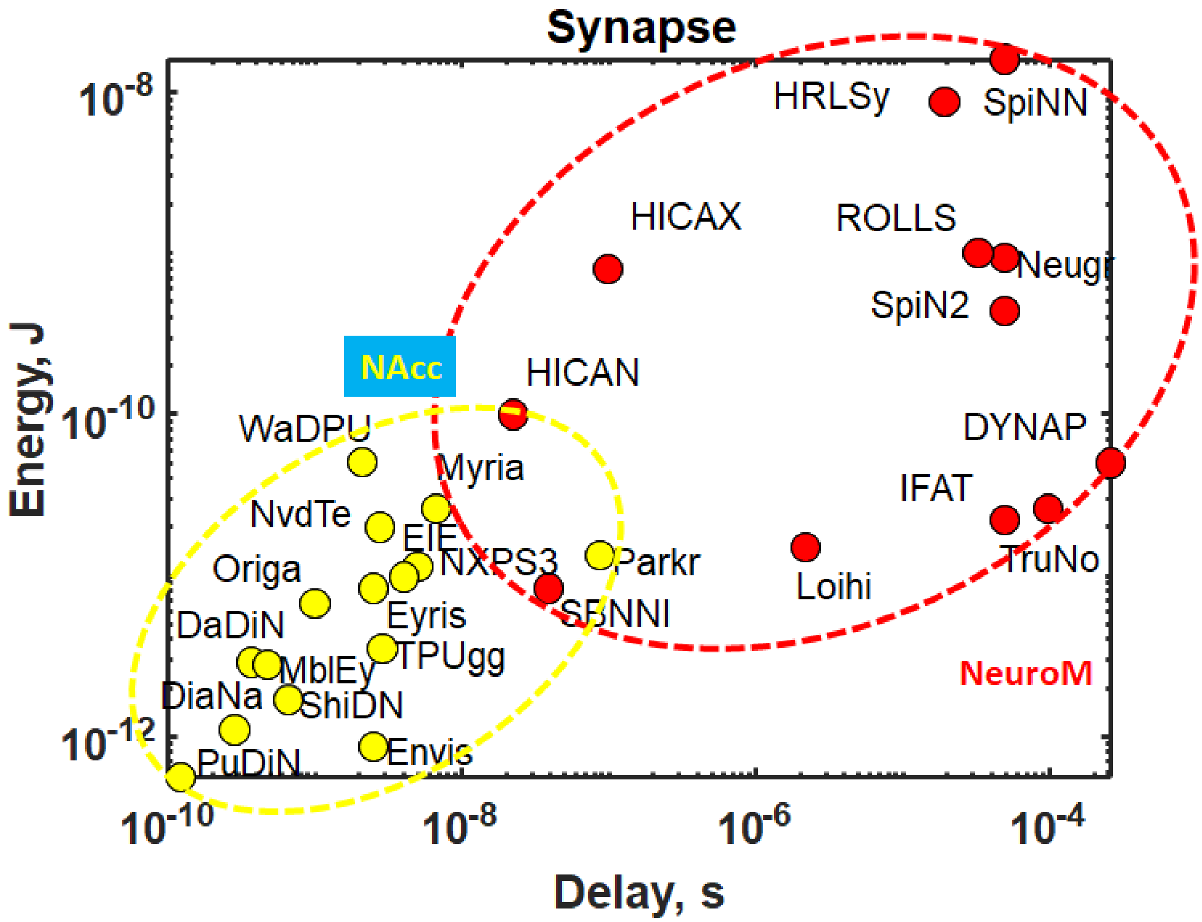


Figure 13. Energy vs. delay for synapses.

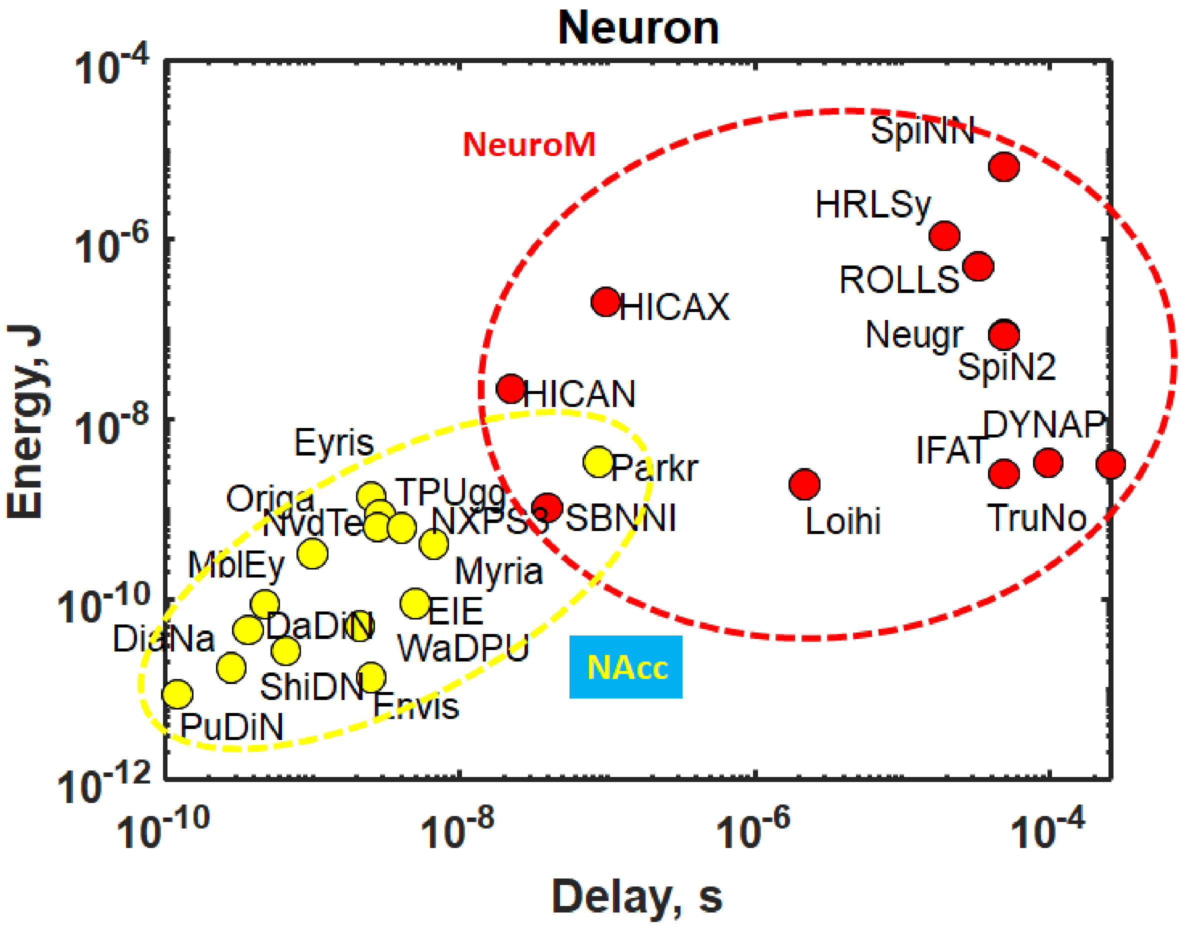


Figure 14. Energy vs. delay for neurons.

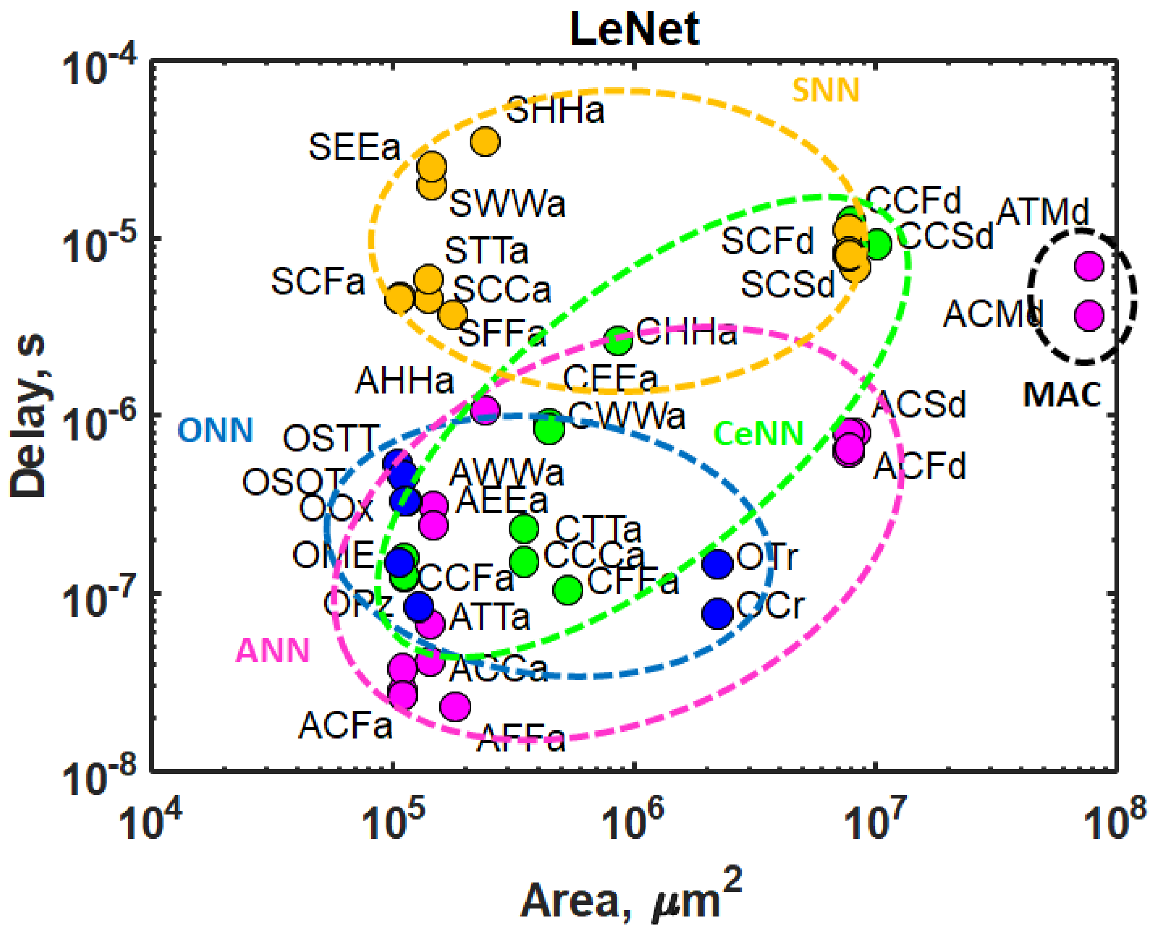


Figure 15. Delay vs. area for LeNet CoNN.

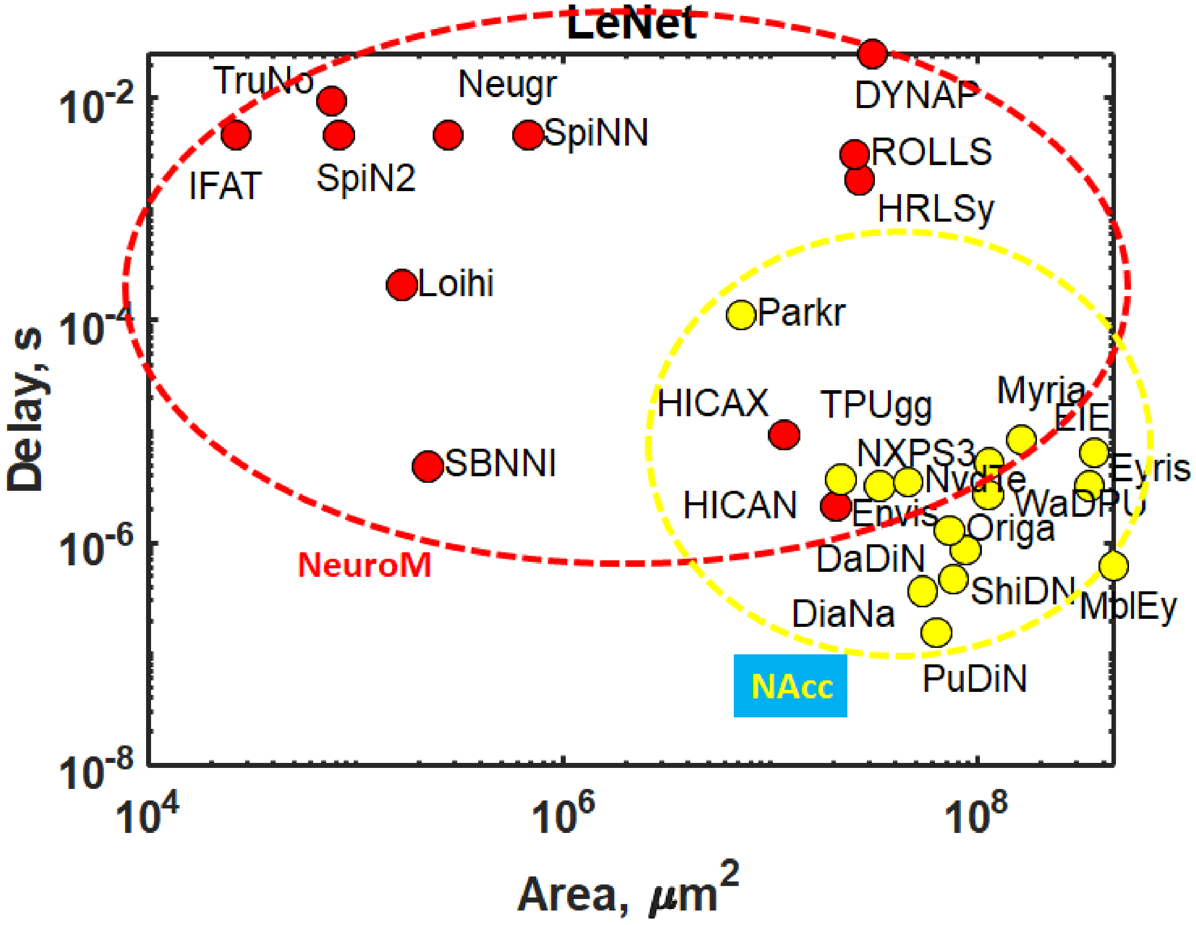


Figure 16. Delay vs. area for LeNet CoNN.

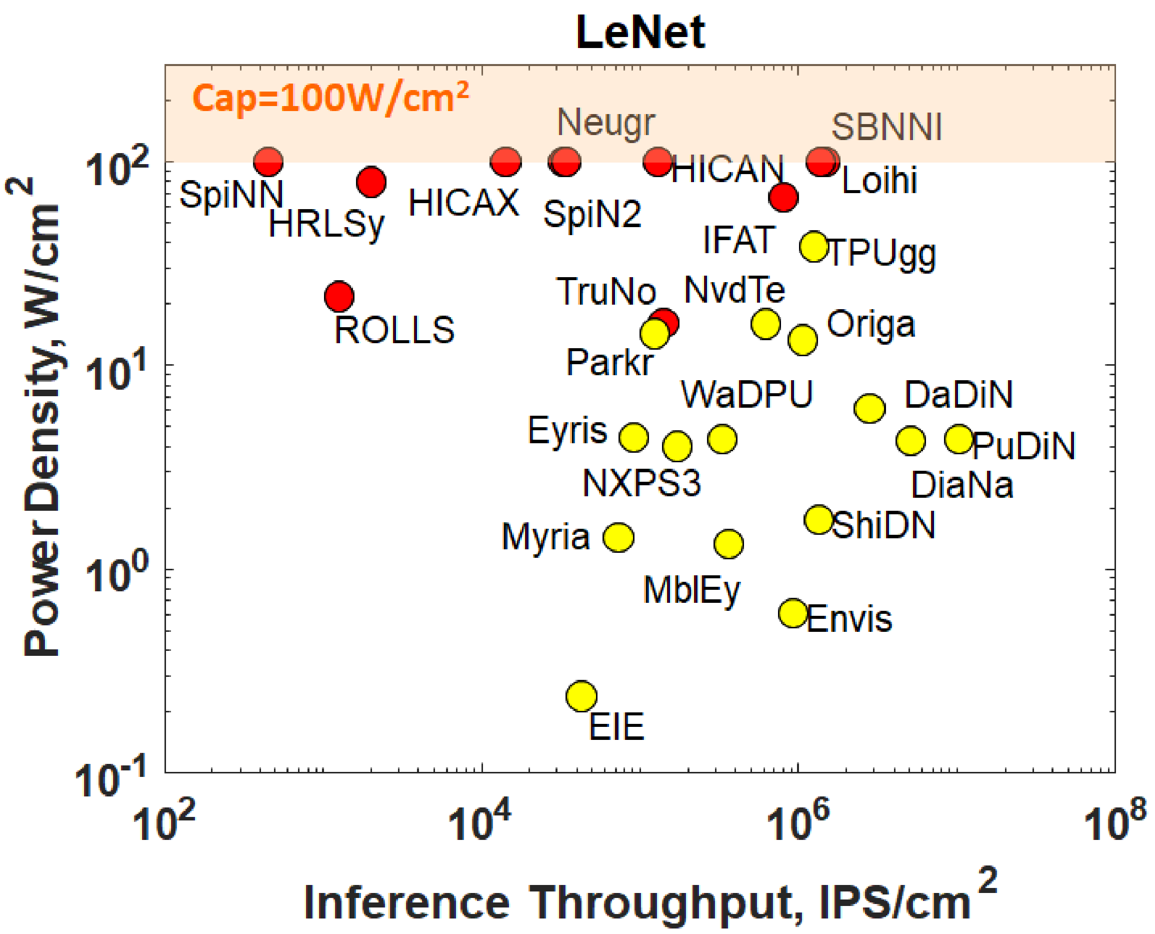


Figure 17. Dissipated power density vs. inference operation throughput per unit area in a circuit implementing the LeNet convolutional neural network, includes benchmarks for prototype neuromorphic chips and neural accelerators.

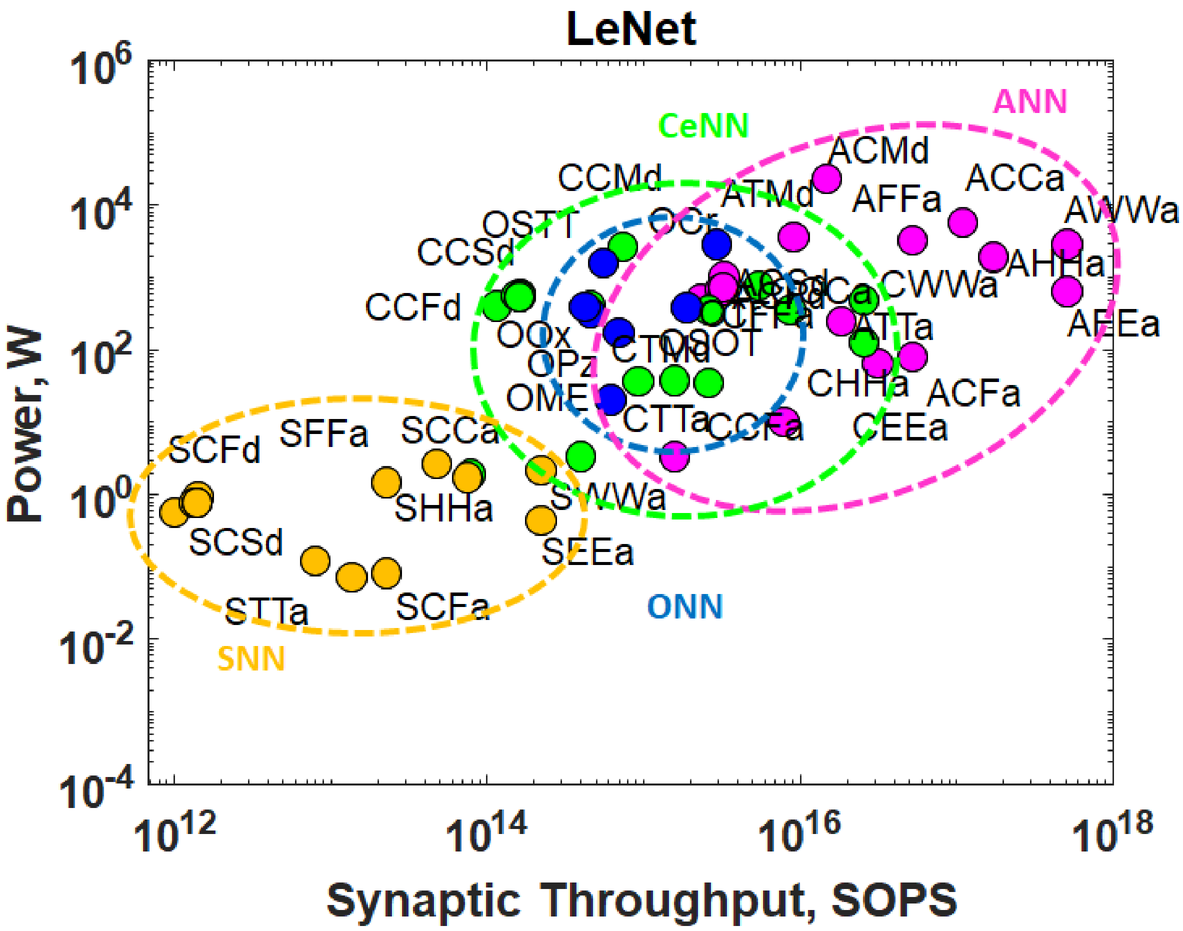


Figure 18. Power vs. synaptic throughput for LeNet.

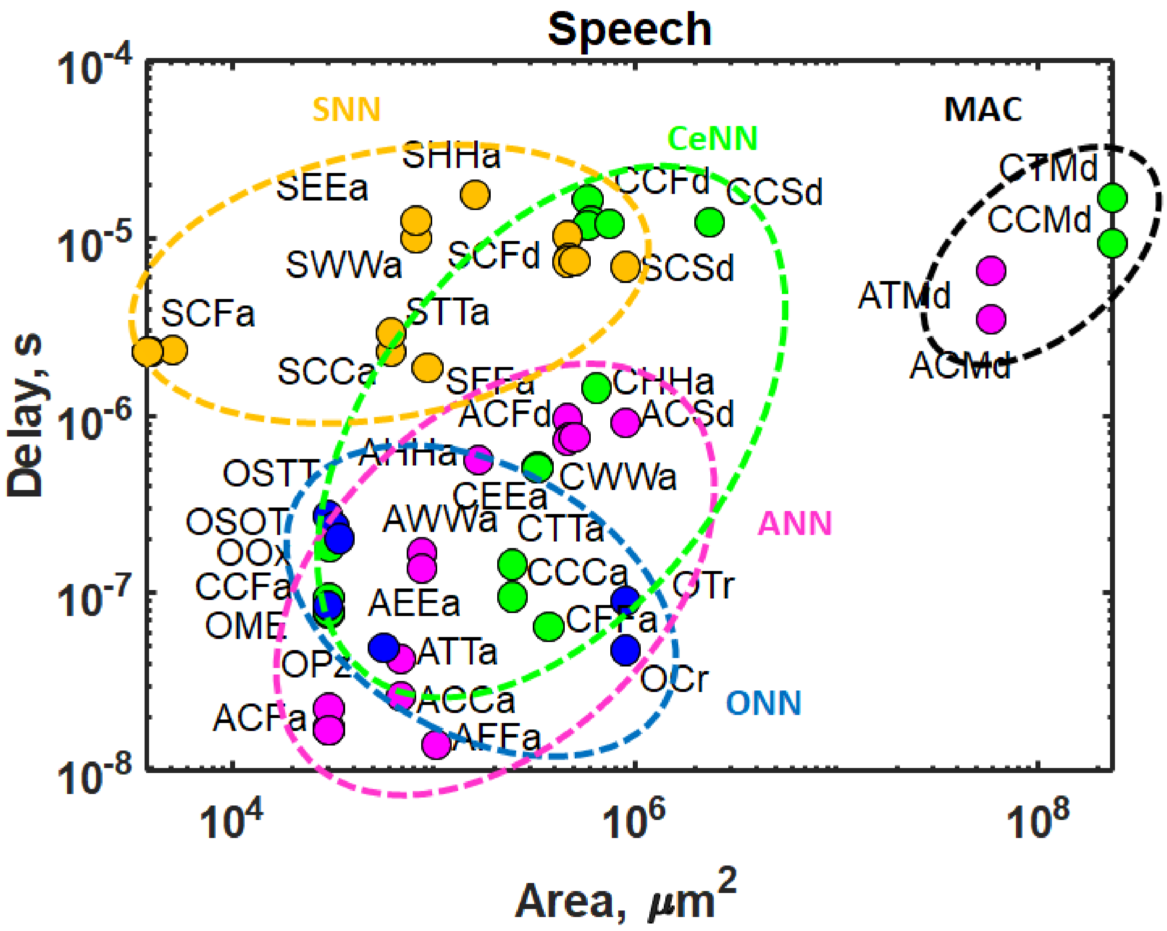


Figure 19. Delay vs. area for the speech recognition.

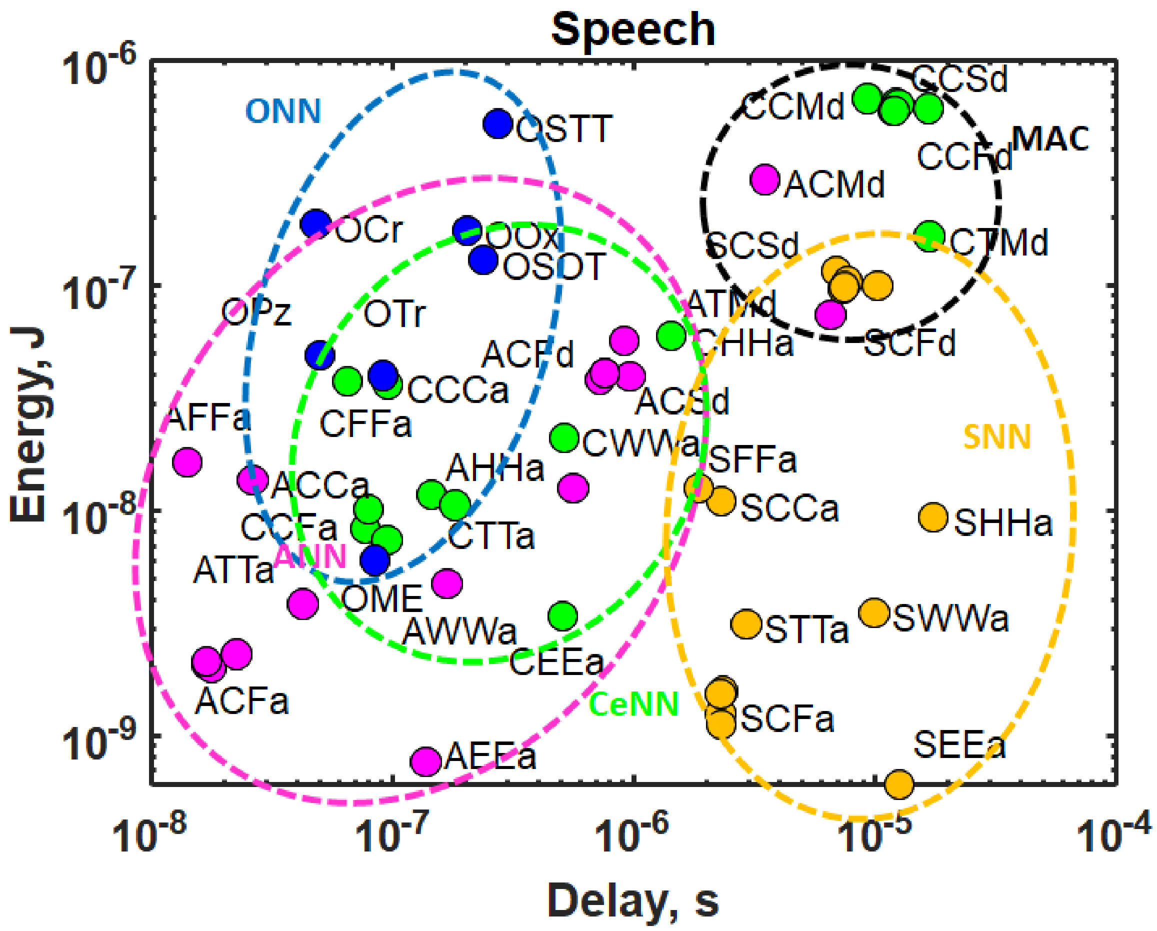


Figure 20. Energy vs. delay for the speech recognition.

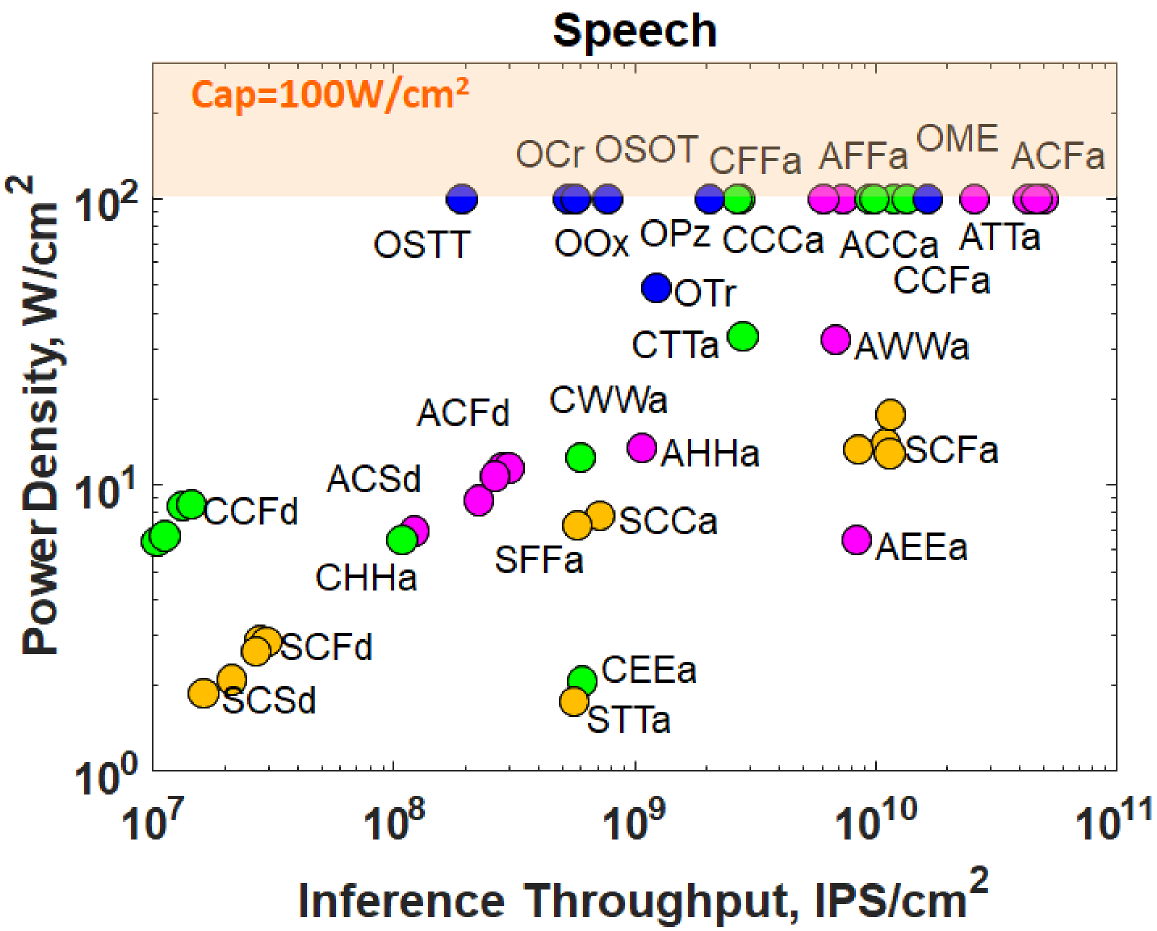


Figure 21. Power density vs. inference throughput for the speech recognition.

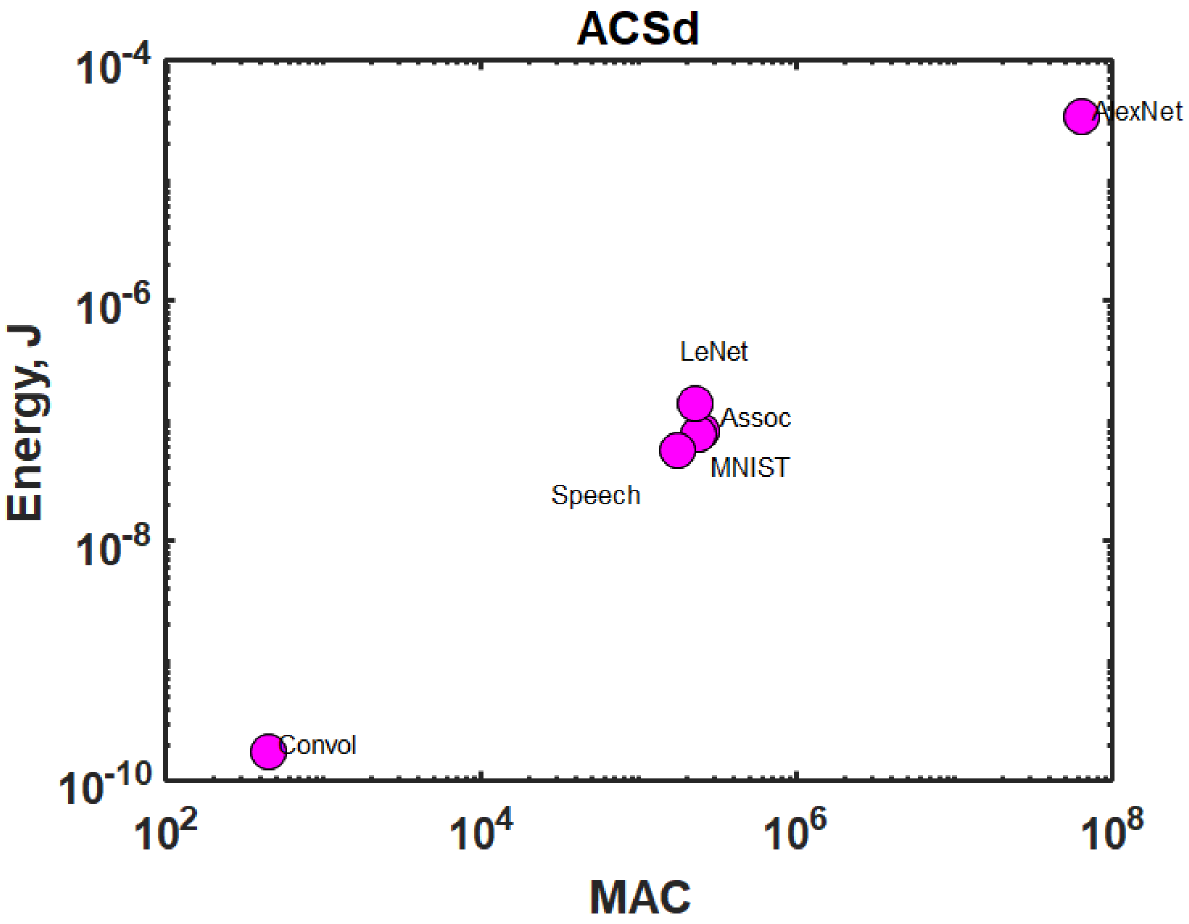


Figure 22. Energy vs. MAC in digital neurons and SRAM synapses for various workloads.

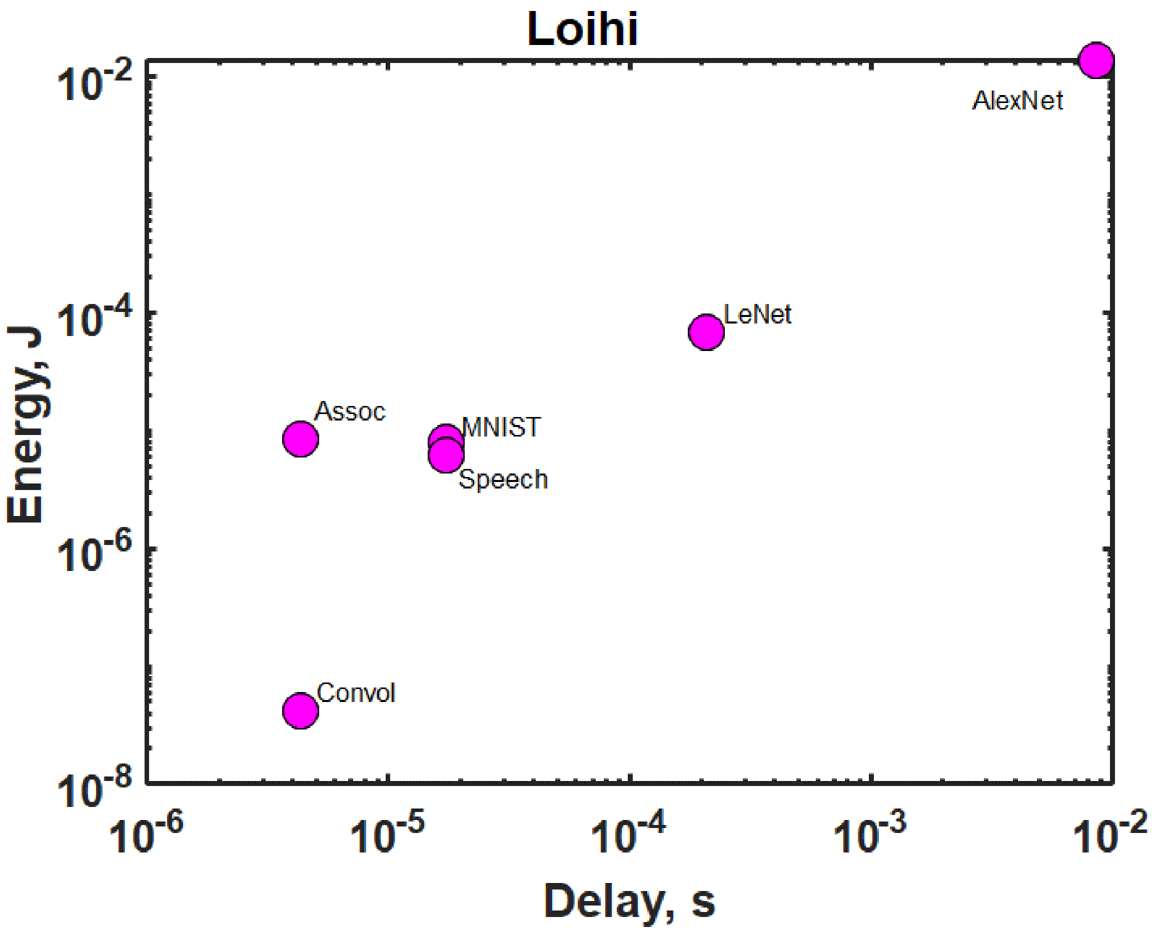


Figure 23. Energy vs. delay in Loihi for various workloads.

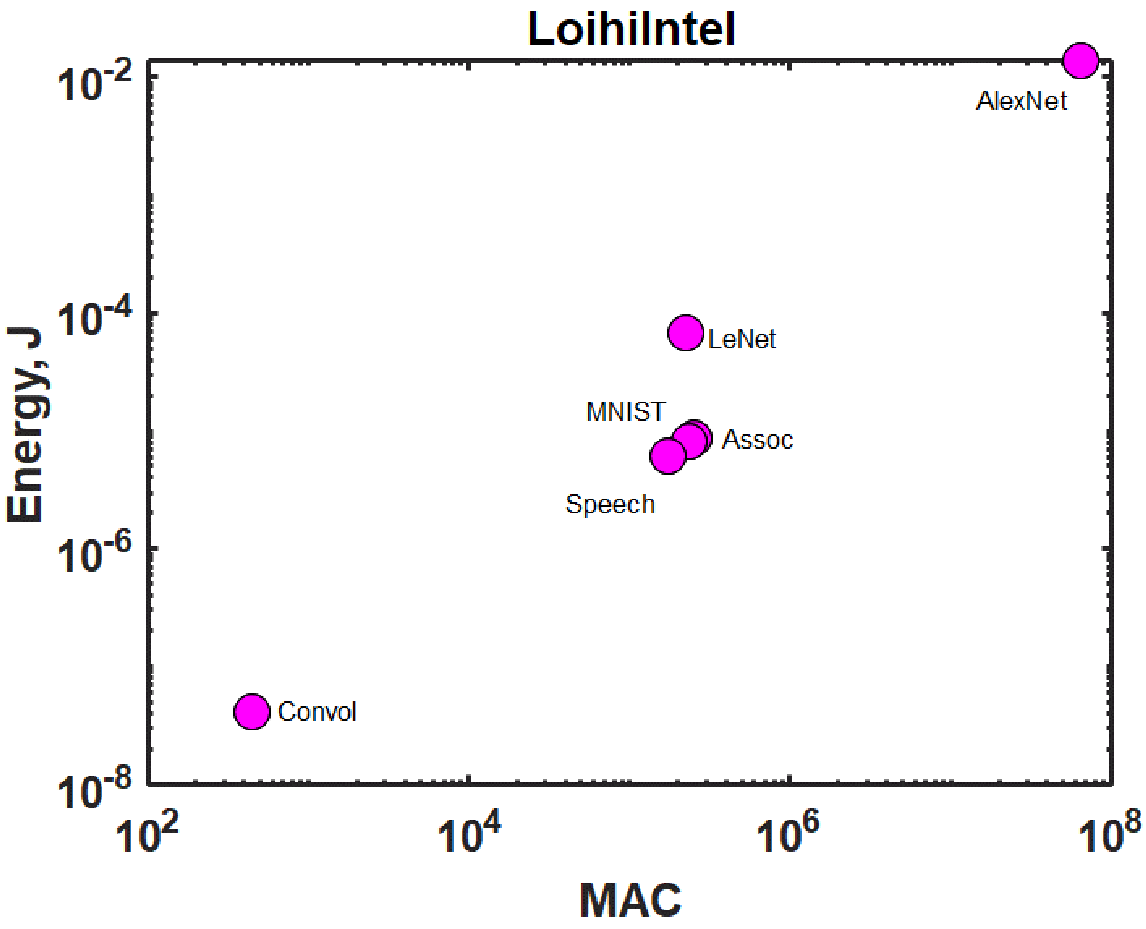
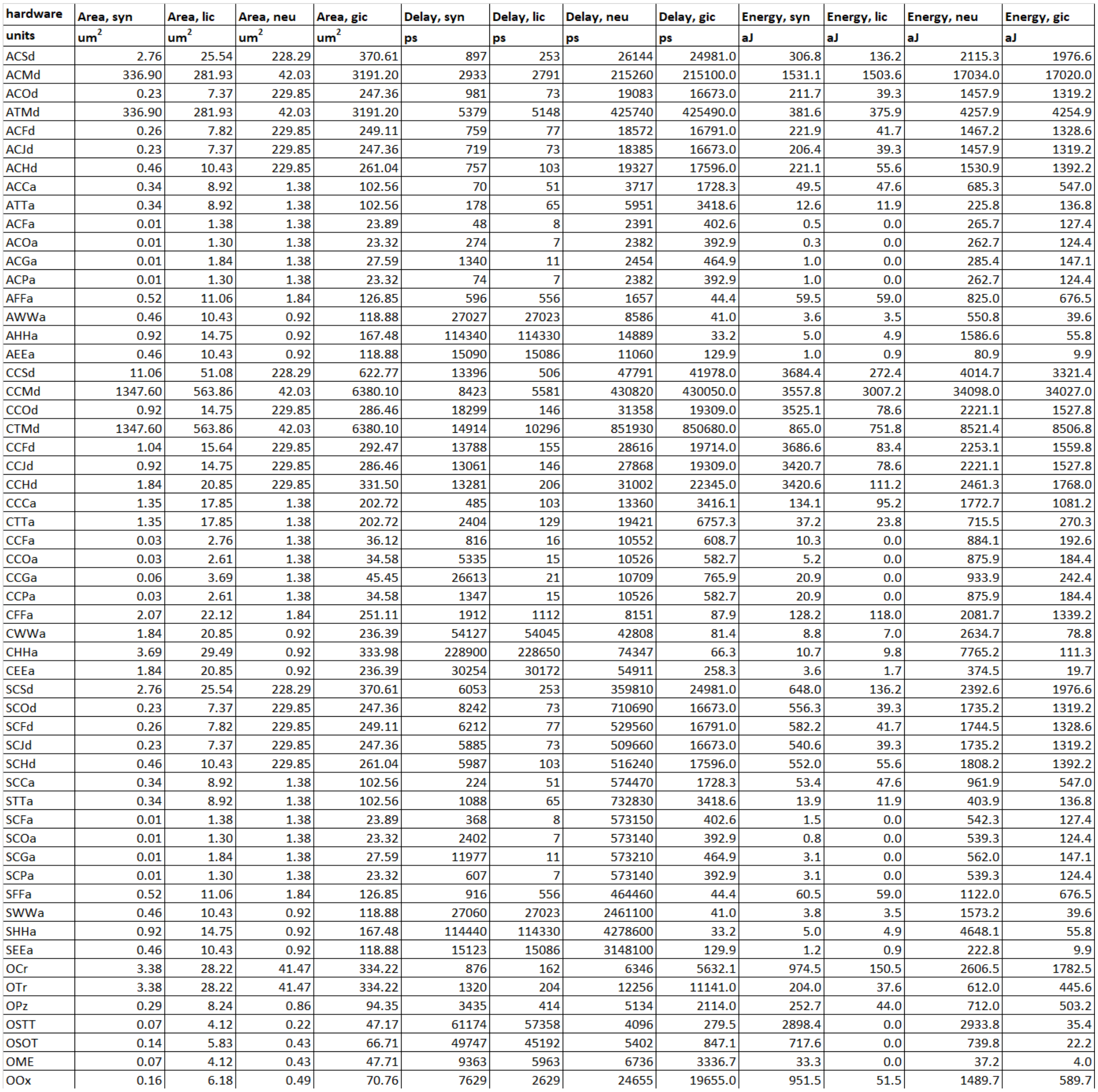


Figure 24. Energy vs. MAC in Loihi for various workloads.

Table 6. Performance benchmarks for the combinations of devices and network types.



hardware	Area, syn	Area, lic	Area, neu	Area, gic	Delay, syn	Delay, lic	Delay, neu	Delay, gic	Energy, syn	Energy, lic	Energy, neu	Energy, gic
units	um^2	um^2	um^2	um^2	ps	ps	ps	ps	aJ	aJ	aJ	aJ
ACSd	2.76	25.54	228.29	370.61	897	253	26144	24981.0	306.8	136.2	2115.3	1976.6
ACMd	336.90	281.93	42.03	3191.20	2933	2791	215260	215100.0	1531.1	1503.6	17034.0	17020.0
ACOd	0.23	7.37	229.85	247.36	981	73	19083	16673.0	211.7	39.3	1457.9	1319.2
ATMd	336.90	281.93	42.03	3191.20	5379	5148	425740	425490.0	381.6	375.9	4257.9	4254.9
ACFd	0.26	7.82	229.85	249.11	759	77	18572	16791.0	221.9	41.7	1467.2	1328.6
ACJd	0.23	7.37	229.85	247.36	719	73	18385	16673.0	206.4	39.3	1457.9	1319.2
ACHd	0.46	10.43	229.85	261.04	757	103	19327	17596.0	221.1	55.6	1530.9	1392.2
ACCa	0.34	8.92	1.38	102.56	70	51	3717	1728.3	49.5	47.6	685.3	547.0
ATTa	0.34	8.92	1.38	102.56	178	65	5951	3418.6	12.6	11.9	225.8	136.8
ACFa	0.01	1.38	1.38	23.89	48	8	2391	402.6	0.5	0.0	265.7	127.4
ACOa	0.01	1.30	1.38	23.32	274	7	2382	392.9	0.3	0.0	262.7	124.4
ACGa	0.01	1.84	1.38	27.59	1340	11	2454	464.9	1.0	0.0	285.4	147.1
ACPa	0.01	1.30	1.38	23.32	74	7	2382	392.9	1.0	0.0	262.7	124.4
AFFa	0.52	11.06	1.84	126.85	596	556	1657	44.4	59.5	59.0	825.0	676.5
AWWa	0.46	10.43	0.92	118.88	27027	27023	8586	41.0	3.6	3.5	550.8	39.6
AHHa	0.92	14.75	0.92	167.48	114340	114330	14889	33.2	5.0	4.9	1586.6	55.8
AEEa	0.46	10.43	0.92	118.88	15090	15086	11060	129.9	1.0	0.9	80.9	9.9
CCSd	11.06	51.08	228.29	622.77	13396	506	47791	41978.0	3684.4	272.4	4014.7	3321.4
CCMd	1347.60	563.86	42.03	6380.10	8423	5581	430820	430050.0	3557.8	3007.2	34098.0	34027.0
CCOd	0.92	14.75	229.85	286.46	18299	146	31358	19309.0	3525.1	78.6	2221.1	1527.8
CTMd	1347.60	563.86	42.03	6380.10	14914	10296	851930	850680.0	865.0	751.8	8521.4	8506.8
CCFd	1.04	15.64	229.85	292.47	13788	155	28616	19714.0	3686.6	83.4	2253.1	1559.8
CCJd	0.92	14.75	229.85	286.46	13061	146	27868	19309.0	3420.7	78.6	2221.1	1527.8
CCHd	1.84	20.85	229.85	331.50	13281	206	31002	22345.0	3420.6	111.2	2461.3	1768.0
CCCa	1.35	17.85	1.38	202.72	485	103	13360	3416.1	134.1	95.2	1772.7	1081.2
CTTa	1.35	17.85	1.38	202.72	2404	129	19421	6757.3	37.2	23.8	715.5	270.3
CCFa	0.03	2.76	1.38	36.12	816	16	10552	608.7	10.3	0.0	884.1	192.6
CCOa	0.03	2.61	1.38	34.58	5335	15	10526	582.7	5.2	0.0	875.9	184.4
CCGa	0.06	3.69	1.38	45.45	26613	21	10709	765.9	20.9	0.0	933.9	242.4
CCPa	0.03	2.61	1.38	34.58	1347	15	10526	582.7	20.9	0.0	875.9	184.4
CFFa	2.07	22.12	1.84	251.11	1912	1112	8151	87.9	128.2	118.0	2081.7	1339.2
CWWa	1.84	20.85	0.92	236.39	54127	54045	42808	81.4	8.8	7.0	2634.7	78.8
CHHa	3.69	29.49	0.92	333.98	228900	228650	74347	66.3	10.7	9.8	7765.2	111.3
CEEa	1.84	20.85	0.92	236.39	30254	30172	54911	258.3	3.6	1.7	374.5	19.7
SCSd	2.76	25.54	228.29	370.61	6053	253	359810	24981.0	648.0	136.2	2392.6	1976.6
SCOd	0.23	7.37	229.85	247.36	8242	73	710690	16673.0	556.3	39.3	1735.2	1319.2
SCFd	0.26	7.82	229.85	249.11	6212	77	529560	16791.0	582.2	41.7	1744.5	1328.6
SCJd	0.23	7.37	229.85	247.36	5885	73	509660	16673.0	540.6	39.3	1735.2	1319.2
SCHd	0.46	10.43	229.85	261.04	5987	103	516240	17596.0	552.0	55.6	1808.2	1392.2
SCCa	0.34	8.92	1.38	102.56	224	51	574470	1728.3	53.4	47.6	961.9	547.0
STTa	0.34	8.92	1.38	102.56	1088	65	732830	3418.6	13.9	11.9	403.9	136.8
SCFa	0.01	1.38	1.38	23.89	368	8	573150	402.6	1.5	0.0	542.3	127.4
SCOa	0.01	1.30	1.38	23.32	2402	7	573140	392.9	0.8	0.0	539.3	124.4
SCGa	0.01	1.84	1.38	27.59	11977	11	573210	464.9	3.1	0.0	562.0	147.1
SCPa	0.01	1.30	1.38	23.32	607	7	573140	392.9	3.1	0.0	539.3	124.4
SFFa	0.52	11.06	1.84	126.85	916	556	464460	44.4	60.5	59.0	1122.0	676.5
SWWa	0.46	10.43	0.92	118.88	27060	27023	2461100	41.0	3.8	3.5	1573.2	39.6
SHHa	0.92	14.75	0.92	167.48	114440	114330	4278600	33.2	5.0	4.9	4648.1	55.8
SEEa	0.46	10.43	0.92	118.88	15123	15086	3148100	129.9	1.2	0.9	222.8	9.9
OCr	3.38	28.22	41.47	334.22	876	162	6346	5632.1	974.5	150.5	2606.5	1782.5
OTr	3.38	28.22	41.47	334.22	1320	204	12256	11141.0	204.0	37.6	612.0	445.6
OPz	0.29	8.24	0.86	94.35	3435	414	5134	2114.0	252.7	44.0	712.0	503.2
OSTT	0.07	4.12	0.22	47.17	61174	57358	4096	279.5	2898.4	0.0	2933.8	35.4
OSOT	0.14	5.83	0.43	66.71	49747	45192	5402	847.1	717.6	0.0	739.8	22.2
OME	0.07	4.12	0.43	47.71	9363	5963	6736	3336.7	33.3	0.0	37.2	4.0
OOx	0.16	6.18	0.49	70.76	7629	2629	24655	19655.0	951.5	51.5	1489.7	589.7

9. References

[1] A. Sengupta, A. Banerjee, and K. Roy, "Hybrid Spintronic-CMOS Spiking Neural Network with On-Chip Learning: Devices, Circuits, and Systems", Phys. Rev. Appl. 6, 064003 (2016).

[2] M. J. Marinella, S. Agarwal, A. Hsia, I. Richter, R. Jacobs-Gedrim, J. Niroula, S. J. Plimpton, E. Ipek, and C. D. James, "Multiscale Co-Design Analysis of Energy, Latency, Area, and

Accuracy of a ReRAM Analog Neural Training Accelerator," in *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, vol. 8, no. 1, pp. 86-101, March 2018.

[3] P. Chen, X. Peng and S. Yu, "NeuroSim+: An integrated device-to-algorithm framework for benchmarking synaptic devices and array architectures," *2017 IEEE International Electron Devices Meeting (IEDM)*, San Francisco, CA, 2017, pp. 6.1.1-6.1.4.

[4] C. Pan, A. Naeemi, "Non-Boolean Computing Benchmarking for Beyond-CMOS Devices Based on Cellular Neural Network", IEEE J. Explor. Comput. Devices and Circuits (2016).

[5] I. Palit, B. Sedighi, Q. Lou, M. Niemier, J. Nahas, X. S. Hu, "Analytical Models for Calculating Power and Performance of a CNN System", unpublished.

[6] G. Srinivasan, A. Sengupta, and K. Roy, "Magnetic Tunnel Junction Based Long-Term Short-Term Stochastic Synapse for a Spiking Neural Network with On-Chip STDP Learning", Scientific Reports 6, 29545 (2016).

[7] X. Wang, Y. Chen, H. Xi, H. Li, and D. Dimitrov, "Spintronic memristor through spin-torque-induced magnetization motion," IEEE Electron Device Lett., vol. 30, no. 3, pp. 294–297, Mar. 2009.

[8] A. W. Stephan, J. Hu, S. J. Koester, "Benchmarking Inverse Rashba-Edelstein Magnetoelectric Devices for Neuromorphic Computing", available online https://arxiv.org/abs/1811.08624 (2018).

[9] J. Grollier, D. Querlioz and M. D. Stiles, "Spintronic Nanodevices for Bioinspired Computing," in Proceedings of the IEEE, vol. 104, no. 10, pp. 2024-2039, Oct. 2016.

[10] M. Jerry *et al*., "Ferroelectric FET analog synapse for acceleration of deep neural network training," *2017 IEEE International Electron Devices Meeting (IEDM),* San Francisco, CA, 2017, pp. 6.2.1-6.2.4.

[11] E. W. Kinder, C. Alessandri, P. Pandey, G. Karbasian, S. Salahuddin and A. Seabaugh, "Partial switching of ferroelectrics for synaptic weight storage," *2017 75th Annual Device Research Conference (DRC)*, South Bend, IN, 2017, pp. 1-2.

[12] M. Hu, H. Li, Y. Chen, Q. Wu, G. S. Rose and R. W. Linderman, "Memristor Crossbar-Based Neuromorphic Computing System: A Case Study," in *IEEE Transactions on Neural Networks and Learning Systems*, vol. 25, no. 10, pp. 1864-1878, Oct. 2014.

[13] C. Liu, B. Yan, C. Yang, L. Song, Z. Li, B. Liu, Y. Chen, H. Li, Q. Wu, H. Jiang, "A spiking neuromorphic design with resistive crossbar," *2015 52nd ACM/EDAC/IEEE Design Automation Conference (DAC),* San Francisco, CA, 2015, pp. 1-6.

[14] F. Merrikh-Bayat, X. Guo, M. Klachko, M. Prezioso, K. K. Likharev and D. B. Strukov, "High-Performance Mixed-Signal Neurocomputing With Nanoscale Floating-Gate Memory Cell

Arrays," in *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 10, pp. 4782-4790, Oct. 2018.

[15] M. Bavandpour, M. R. Mahmoodi and D. B. Strukov, "Energy-Efficient Time-Domain Vector-by-Matrix Multiplier for Neurocomputing and Beyond," in IEEE Transactions on Circuits and Systems II: Express Briefs. (2019).

[16] Vincent, A.F., Larroque, J., Zhao, W.S., Romdhane, N.B., Bichler, O., Gamrat, C., Klein, J.O., Galdin-Retailleau, S. and Querlioz, D., "Spin-transfer torque magnetic memory as a stochastic memristive synapse". In 2014 IEEE International Symposium on Circuits and Systems (ISCAS), pp. 1074-1077 (2014).

[17] Ramasubramanian, S.G., Venkatesan, R., Sharad, M., Roy, K. and Raghunathan, A., "SPINDLE: SPINtronic deep learning engine for large-scale neuromorphic computing", In Proceedings of the 2014 international symposium on Low power electronics and design, pp. 15-20 (2014).

[18] Q. Lou, C. Pan, J. McGuinness, A. Horvath, A. Naeemi, M. Niemier, and X. S. Hu, "A Mixed Signal Architecture for Convolutional Neural Networks", ACM Journal on Emerging Technologies in Computing Systems (JETC), v. 15, no. 2, art. 19, April 2019.

[19] C. Lee, S. Shakib Sarwar, and K. Roy, "Enabling Spike-based Backpropagation in State-of-the-art Deep Neural Network Architectures", available https://arxiv.org/abs/1903.06379 (2019).

[20] D. E. Nikonov, G. Csaba, W. Porod, T. Shibata, D. Voils, D. Hammerstrom, I. A. Young, and G. I. Bourianoff, "Coupled-Oscillator Associative Memory Array Operation for Pattern Recognition", IEEE J. Explor. Comput. Devices and Circuits v.1, pp. 85-93 (2015).

[21] Nikonov, D.E., Young, I.A. and Bourianoff, G.I., "Convolutional networks for image processing by coupled oscillator arrays". arXiv preprint arXiv:1409.4469 (2014).

[22] Merolla, P.A., J.V. Arthur, R. Alvarez-Icaza, A S. Cassidy, J. Sawada, F. Akopyan, B.L. Jackson, N. Imam, C. Guo, Y. Nakamura, B. Brezzo, I. Vo, S.K. Esser, R. Appuswamy, B. Taba, A. Amir, M.D. Flickner, W.P. Risk, R. Manohar, and D. S. Modha, "A million spiking-neuron integrated circuit with a scalable communication network and interface," *Science*, 345(6197): 668–673, 2014.

[23] Schemmel, J., D. Bruderle, A. Grubl, M. Hock, K. Meier, and S. Millner, "A waferscale neuromorphic hardware system for large-scale neural modeling," *Proc. 2010 IEEE Int. Symp. Circuits and Systems (ISCAS)*, 1947–1950, 2010.

[24] S. A. Aamir, Y. Stradmann, P. Müller, C. Pehle, A. Hartel, A. Grübl, J. Schemmel, K. Meier, "An Accelerated LIF Neuronal Network Array for a Large Scale Mixed-Signal Neuromorphic Architecture", available online arXiv 1804.01906 (2018).

[25] J. M. Cruz-Albrecht, T. Derosier and N. Srinivasa, “A scalable neural chip with synaptic electronics using CMOS integrated memristors”, Nanotechnology 24, 384011 (2013).

[26] E. Painkras *et al.*, "SpiNNaker: A 1-W 18-Core System-on-Chip for Massively-Parallel Neural Network Simulation," in *IEEE Journal of Solid-State Circuits*, vol. 48, no. 8, pp. 1943-1953, Aug. 2013.

[27] E. Stromatias, F. Galluppi, C. Patterson and S. Furber, "Power analysis of large-scale, real-time neural networks on SpiNNaker," *The 2013 International Joint Conference on Neural Networks (IJCNN)*, Dallas, TX, pp. 1-8 (2013).

[28] J. Partzsch, S. Hoppner, M. Eberlein, R. Schuffny, C. Mayr, D. R. Lester, and S. Furber, “A fixed point exponential function accelerator for a neuromorphic many-core system,” in 2017 IEEE International Symposium on Circuits and Systems (ISCAS), May 2017, pp. 1–4.

[29] A. Cassidy et al., “Real-time Scalable Cortical Computing at 46 Giga-Synaptic OPS/Watt with ~100× Speedup in Time-to-Solution and ~100,000× Reduction in Energy-to-Solution”, Proc. of International Conference for High Performance Computing, Networking, Storage and Analysis, SC14 (2014).

[30] Benjamin, B., P. Gao, E. McQuinn, S. Choudhary, A. Chandrasekaran, J. Bussat, R. Alvarez-Icaza, J. Arthur, P. Merolla, and K. Boahen, “Neurogrid: A mixed analog-digital multichip system for large-scale neural simulations,” *Proc. IEEE*, 102(5):699–716, 2014.

[31] Park, J., S. Ha, T. Yu, E. Neftci, and G. Cauwenberghs, “65k-neuron 73-Mevents/s 22-pJ/event asynchronous micro-pipelined integrate-and-fire array transceiver,” *Proc. 2014 IEEE Biomedical Circuits and Systems Conf. (BioCAS)*, 2014.

[32] N. Qiao, H. Mostafa, F. Corradi, M. Osswald, F. Stefanini, D. Sumislawska, and G. Indiveri, “A reconfigurable on-line learning spiking neuromorphic processor comprising 256 neurons and 128K synapses”, Frontiers in Neuroscience, v. 9, 141 (2015).

[33] G. Indiveri, F. Corradi and N. Qiao, "Neuromorphic architectures for spiking deep neural networks," *2015 IEEE International Electron Devices Meeting (IEDM)*, Washington, DC, 2015, pp. 4.2.1-4.2.4.

[34] N. Qiao and G. Indiveri, "Scaling mixed-signal neuromorphic processors to 28 nm FD-SOI technologies," *2016 IEEE Biomedical Circuits and Systems Conference (BioCAS)*, Shanghai, 2016, pp. 552-555.

[35] Davies, M., Srinivasa, N., Lin, T., Chinya, G., Cao, Y., Choday, S., Dimou, G., Joshi, P., Imam, N., Jain, S., Liao, Y., Lin, C., Lines, A., Liu, R., Mathaikutty, D., McCoy, S., Paul, A., Tse, J., Venkataramanan, G., Weng, Y., Wild, A., Yang, Y., and Wang, H. "Loihi: A Neuromorphic Manycore Processor with On-Chip Learning," in *IEEE Micro*, vol. 38, no. 1, pp. 82-99, January/February 2018.

[36] A. Lines, P. Joshi, R. Liu, S. McCoy, J. Tse, Y.-H. Weng, and M. Davies, “Loihi Asynchronous Neuromorphic Research Chip”, Proceedings of 24th IEEE International Symposium on Asynchronous Circuits and Systems, Vienna, May 13-16, 2018.

[37] G. K. Chen et al., “A 4096-neuron 1M-synapse 3.8pJ/SOP Spiking Neural Network with On-chip STDP Learning and Sparse Weights in 10nm FinFET CMOS”, Proc. VLSI Symposium, C24-1, 2018.

[38] P. Blouw, X. Choo, E. Hunsberger, and C. Eliasmith, “Benchmarking Keyword Spotting Efficiency on Neuromorphic Hardware”, available https://arxiv.org/abs/1812.01739 (2018).

[39] Chen, T., Du, Z., Sun, N., Wang, J., Wu, C., Chen, Y. and Temam, O., “Diannao: A small-footprint high-throughput accelerator for ubiquitous machine-learning”. In ACM Sigplan Notices (Vol. 49, No. 4, pp. 269-284). ACM, Feb. 2014.

[40] Chen, Y., Luo, T., Liu, S., Zhang, S., He, L., Wang, J., Li, L., Chen, T., Xu, Z., Sun, N. and Temam, O., “Dadiannao: A machine-learning supercomputer”. In Proceedings of the 47th Annual IEEE/ACM International Symposium on Microarchitecture (pp. 609-622). IEEE Computer Society, Dec. 2014.

[41] Liu, D., Chen, T., Liu, S., Zhou, J., Zhou, S., Teman, O., Feng, X., Zhou, X. and Chen, Y., “Pudiannao: A polyvalent machine learning accelerator”. In ACM SIGARCH Computer Architecture News (Vol. 43, No. 1, pp. 369-381). ACM, Mar. 2015.

[42] Du, Z., Fasthuber, R., Chen, T., Ienne, P., Li, L., Luo, T., Feng, X., Chen, Y. and Temam, “ShiDianNao: Shifting vision processing closer to the sensor”. In ACM SIGARCH Computer Architecture News (Vol. 43, No. 3, pp. 92-104). ACM, June 2015.

[43] Y.-H. Chen, T. Krishna, J. Emer, and V. Sze, “Eyeriss: An energy-efficient reconfigurable accelerator for deep convolutional neural networks,” in IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers, Feb. 2016, pp. 262–263.

[44] S. Han *et al*., "EIE: Efficient Inference Engine on Compressed Deep Neural Network," *2016 ACM/IEEE 43rd Annual International Symposium on Computer Architecture (ISCA)*, Seoul, 2016, pp. 243-254.

[45] R. Andri, L. Cavigelli, D. Rossi and L. Benini, "YodaNN: An Ultra-Low Power Convolutional Neural Network Accelerator Based on Binary Weights," 2016 IEEE Computer Society Annual Symposium on VLSI (ISVLSI), Pittsburgh, PA, 2016, pp. 236-241.

[46] L. Cavigelli and L. Benini, “Origami: A 803-GOp/s/W Convolutional Network Accelerator”, IEEE J. Trans. Circuits and Systems, v. 27, p 2461 (2016). GLVLSI 2015.

[47] B. Moons, R. Uytterhoeven, W. Dehaene, M. Verhelst, “Envision: A 0.26-to-10TOPS/W Subword-Parallel Dynamic-Voltage Accuracy-Frequency-Scalable Convolutional Neural

Network Processor in 28nm FDSOI”, IEEE International Solid-State Circuits Conference, 2017, pp 246-247 (2017).

[48] N. P. Jouppi et al., “In-Datacenter Performance Analysis of a Tensor Processing Unit$^{\text{TM}}$”, Proceeding ISCA '17 Proceedings of the 44th Annual International Symposium on Computer Architecture, 1-12, Toronto, ON, Canada, June 24 - 28, 2017.

[49] L. Gwennap, M. Demler, and L. Case, “A Guide to Processors for Deep Learning”, Linley Group.

[50] D. Moloney, B. Barry, R. Richmond, F. Connor, C. Brick, and D. Donohoe, “Myriad 2: Eye of the computational vision storm,” in IEEE Hot Chips Symposium (HCS), Aug. 2014, pp. 1–18.