An In-Memory-Computing Charge-Domain Ternary CNN Classifier

Xiangxing Yang, *Member, IEEE*, Keren Zhu, *Member, IEEE*, Xiyuan Tang, *Member, IEEE*, Meizhi Wang, *Student Member, IEEE*, Mingtao Zhan, *Student Member, IEEE*, Nanshu Lu, *Senior Member, IEEE*, Jaydeep P. Kulkarni, *Senior Member, IEEE*, David Z. Pan, *Fellow, IEEE*, Yongpan Liu, *Senior Member, IEEE*, and Nan Sun, *Senior Member, IEEE*

***Abstract*— The article presents a charge-domain computing ternary neural network (TNN) classifier with a complete four-layer neural network (NN) on a chip. The proposed ternary network provides 1.5-b resolution (0/+1/−1) for weights and activations, leading to 3.9× fewer operations (OPs) per inference than binary neural network (BNN) for the same Modified National Institute of Standards and Technology (MNIST) accuracy. The 1.5-b multiply-and-accumulate (MAC) is implemented by V_{CM}-based capacitor switching scheme, which inherently benefits from the reduced signal swing on the capacitive digital-to-analog converter (CDAC). Also, the V_{CM}-based MAC introduces sparsity during training, resulting in a lower switching rate. The prototype is fabricated in a 40-nm LP CMOS process with an active area of 0.98 mm^2, operates at 549 frames/s (FPS), and consumes 96 μW. With all OPs on the chip, it achieves 97.1% MNIST accuracy with 0.18 μJ per classification, which is the smallest to our knowledge for comparable MNIST classification accuracy.**

***Index Terms*— In-memory computing, mixed-signal processing, switched capacitors (SCs), ternary neural networks (TNNs).**

I. Introduction

DEEP neural networks (DNNs) have achieved state-of-the-art performance on various applications such as pattern recognition [1], image classification [2], and object detection [3]. The core operation (OP) of DNN inference is multiply-and-accumulate (MAC) calculation, as shown in Fig. 1. Challenges arise from the fact that modern DNN models require millions to billions of MAC OPs, making them difficult to deploy on edge platforms. When evaluating the total energy in a DNN model, the energy per inference can be expressed as

$$\frac{\text{Energy}}{\text{Inference}} = \frac{\text{Energy}}{\text{Operation}} \times \frac{\text{Operations}}{\text{Inference}} \tag{1}$$

where the energy per OP is affected by hardware design. OPs per inference is the number of MAC OPs in a DNN model. Both metrics should be considered in low-power design optimization.

The energy per OP can be broken down into memory access energy and computational energy. Many DNN accelerators have been designed to boost the computational energy efficiency of MAC OPs [4], [5]. To alleviate the memory accessing energy, recent works proposed in-memory-computing (IMC) [7], [8], [9], [10] and low-resolution neural networks [11]. The key concept of IMC is enabling the computational circuitry to access the stationary weights over many stored bits in a memory column, amortizing memory read and write energy. Binary neural network (BNN) is the extreme case among low-resolution neural networks [12]. BNN combined with IMC greatly improves the storage and MAC computing efficiency. With weights and activations restricted to ±1, the multiplication in BNN is simplified as 1-bit XNOR OP, making it well-suited for edge-based applications [13]. For the accumulation of bit products, collecting charge on a summation node has been proven as an energy-efficient way [15], [16], [17]. Charge-domain computing benefits from the high

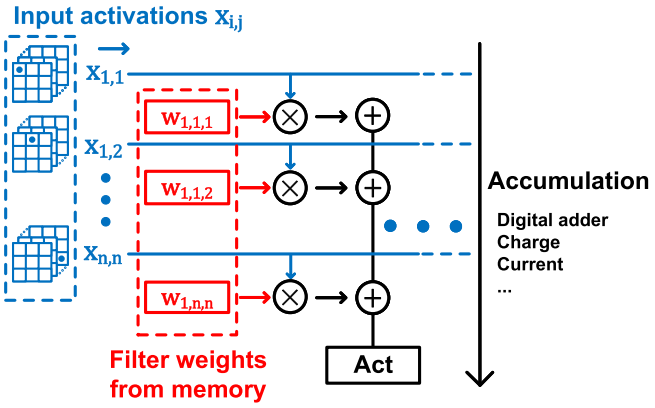


Fig. 1. Typical structure for MAC calculation arrays.

Manuscript received 12 February 2022; revised 13 October 2022 and 9 January 2023; accepted 15 January 2023. Date of publication 2 February 2023; date of current version 25 April 2023. This article was approved by Associate Editor Meng-Fan Chang. *(Corresponding author: Xiangxing Yang.)*

Xiangxing Yang, Keren Zhu, Meizhi Wang, Nanshu Lu, Jaydeep P. Kulkarni, and David Z. Pan are with the Department of Electrical and Computer Engineering, The University of Texas at Austin, Austin, TX 78712 USA (e-mail: yangxx@utexas.edu).

Xiyuan Tang is with the Institute for Artificial Intelligence and the School of Integrated Circuit, Peking University, Beijing 100871, China.

Mingtao Zhan and Yongpan Liu are with the Department of Electronic Engineering, Tsinghua University, Beijing 100084, China.

Nan Sun was with the Department of Electrical and Computer Engineering, The University of Texas at Austin, Austin, TX 78712 USA. He is now with the Department of Electronic Engineering, Tsinghua University, Beijing 100084, China.

Color versions of one or more figures in this article are available at https://doi.org/10.1109/JSSC.2023.3238725.

Digital Object Identifier 10.1109/JSSC.2023.3238725

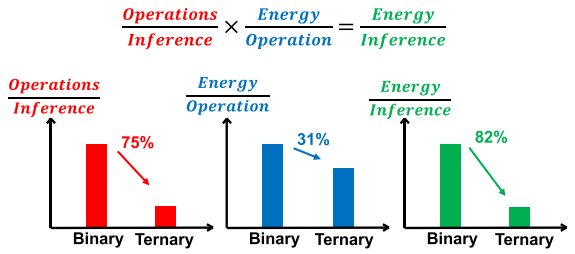


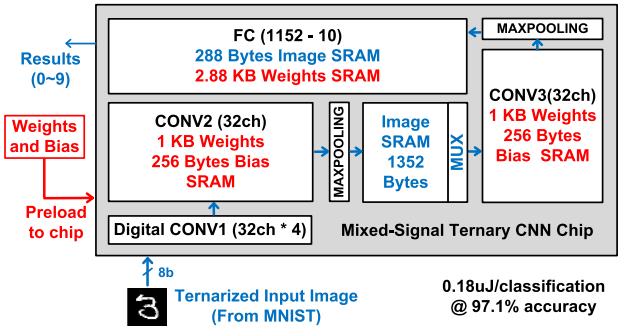
Fig. 2. Comparison of energy consumption between binary charge-domain computing and proposed ternary charge-domain computing under 98% MNIST accuracy.

linearity, stability, and accurate matching of metal-oxide-metal (MOM) capacitors [14]. In these designs, the output voltage of each synapse or bit cell is connected to the bottom plate of an MOM capacitor, where binary multiplication results (+1 and −1) are mapped to drain power voltage (VDD) and ground (GND). With the top plates of capacitors from all the synapses connected, the addition is performed via charge redistribution on this summing node, where the voltage expresses the MAC result. The integration of charge-domain computing and BNN has demonstrated state-of-the-art energy efficiency [15], [16].

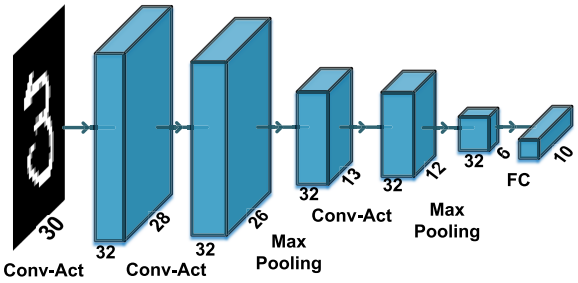
Nevertheless, when evaluating the overall energy per inference, a tradeoff between the energy per OP and OPs per inference needs to be considered [18]. The energy per OP scales down with the computing resolution [6]. Despite the 1-bit computation yielding low energy per OP, the BNN model incurs severe information loss. If given an accuracy requirement for a specific task, BNN requires a deeper network or more channels, namely more OPs per inference, than a full-resolution network [19].

This article presents a highly efficient ternary neural network (TNN) accelerator that reduces both energy per OP and OP per interference, thus greatly reducing energy per interference compared to a BNN. The proposed ternary network provides a 1.5-b resolution (0/+1/−1). A baseline test is conducted by building BNN and TNN models targeting the same 98% accuracy on the Modified National Institute of Standards and Technology (MNIST) dataset. As shown in Fig. 2, the proposed TNN model features 75% OPs/inference reduction compared to the BNN model. The 1.5-b MAC is implemented by a V_{CM}-based capacitor switching scheme, which inherently benefits from the reduced signal swing on the capacitive digital-to-analog converter (CDAC). As a result, it consumes 31% lower energy/OP than binary charge-domain MAC. The overall energy/inference is reduced by 82%. It consumes only 0.18 μJ total energy for MNIST classification, the smallest to our best knowledge for comparable classification accuracy.

This article is an extended version of [20], providing a detailed explanation and analysis of the design. The rest of this article is organized as follows. Section II presents the architecture and data path of the design. Section III presents the detailed circuit implementation of the 1.5-b SC neuron, the comparison between charge-domain BNN and TNN, and the impact of circuit nonidealities. Section IV presents the measurement results. Finally, Section V concludes the article.

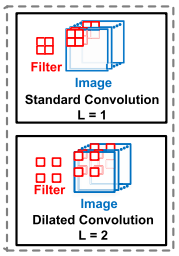


(a)



(b)

Layer	Type	Size	Channel	Filter Size	Dilated
1	CONV-TN	30x30	1(input)	2x2	2
2	CONV-TN	28x28	32		2
2p	MAX POOL	26x26			1
3	CONV-TN	13x13			1
3p	MAX POOL	12x12			1
4	FC	(6x6x32) 1152 - 10			



(c)

Fig. 3. (a) Overview of the proposed chip. Our design features a pipeline architecture, with all weights and biases memory on the chip. (b) Proposed TNN features 31% less energy/OP, 75% fewer OPs/inference, and 82% lower energy/inference than BNN. (c) Topology of the proposed TNN (description of the input feature size and the number of channels in each layer).

II. System Architecture

Fig. 3(a) shows the chip architecture and Fig. 3(b) and (c) shows the neural network (NN) topology. The pipeline data path consists of one digital CNN layer CONV1 at the input, two mixed-signal hidden CNN layers, CONV2 and CONV3, followed by max-pooling layers, one SRAM bank to store image data, and one mixed-signal fully connected (FC) layer at the end. Each convolutional layer has a 1.5-b quantizer at the output as an activation function, either implemented by digital logic or a pair of analog comparators. All weights, biases, and transition thresholds of activation functions are trained offline on Tensorflow. The TNN model requires 5.44 kB for total weights and biases. They are loaded to on-chip SRAM before classification via the write circuitry, including scan chains, address decoder, and bitline and wordline drivers. The weight memory is integrated with computation elements. It remains

stationary during the whole inference to mitigate the data movement cost.

The input 1-B grayscale pixels are ternarized to a one-channel, tri-level picture (black, white, and gray), padded to 30 × 30 pixels, then fed into the chip via 8b bus. To exploit hardware parallelism and regularity, the number of channels for each CNN layer is 32 with 2 × 2 valid convolutions. Dilated 2 × 2 filters are implemented on CONV1 and CONV2 to increase the receptive field to 3 × 3 [21]. Evaluated on the proposed four-layer model, the 2 × 2 dilated convolution costs 56% less area and CDAC power than 3 × 3 convolution, with the loss of 0.7% classification accuracy. To further improve classification accuracy with the architecture, a different number of bias units are evaluated in the python model. Experiments show allocating 20% of the full-scale range as biasing units would improve the classification accuracy by 0.8%. As a result, 32 biased units are introduced to CONV2 and CONV3 in addition to the 128-pixel convolutional window.

A. Design of the Input Layer CONV1

The architecture of the input layer CONV1 is shown in Fig. 4. CONV1 takes the ternarized image data and computes the multiplications of one-channel, 2 × 2 convolution window. The results of CONV1 are integers within the range of [−4, +4]. For the four-elements MAC OP, the charge-domain computing approach does not benefit from the power, area, simplicity, and error rate than the digital circuit, because of the parasitic capacitance on the charge summing node and the noise/offset from the activation comparators. As a result, the CONV1 is synthesized from Verilog and implemented in the RTL-to-GDS flow. Observed from the training results of the TNN model, the high and low activation thresholds remain less than one LSB, which indicates the activation of CONV1 only generates zero with zero MAC input. To reduce hardware cost, the activation function of CONV1 ($\mathrm{ACT_{CONV1}}$) utilizes fixed-step thresholds and is hardcoded as

$$\mathrm{ACT_{CONV1}}(X) = \begin{cases} 1, & \text{if } X > 0 \\ -1, & \text{if } X < 0 \\ 0, & \text{if } X = 0. \end{cases}$$

This simplification results in an area of 70 × 2.5 μm for each CONV1 channel, and the area-efficient implementation enables us to stack 4 of the 32-channel computational logics for higher throughput. The 128 pixels image data for CONV2 are generated in one clock cycle.

B. Datapath From CONV2 to CONV3

The convolution of CONV2 is shown in Fig. 5(a). CONV2 performs the dilated convolution of 32-channel input data with 32 filters. Each filter is applied independently to the input activation and generates the output image of a specific channel. As a result, the output also has 32 channels. The 128-pixel input activation from the output of CONV1 is broadcasted across 32 parallel filters, as shown in Fig. 5(b). Once the CONV2 input is ready, the 32-channel parallel switched-capacitor (SC) neuron CONV2 processes the data and then

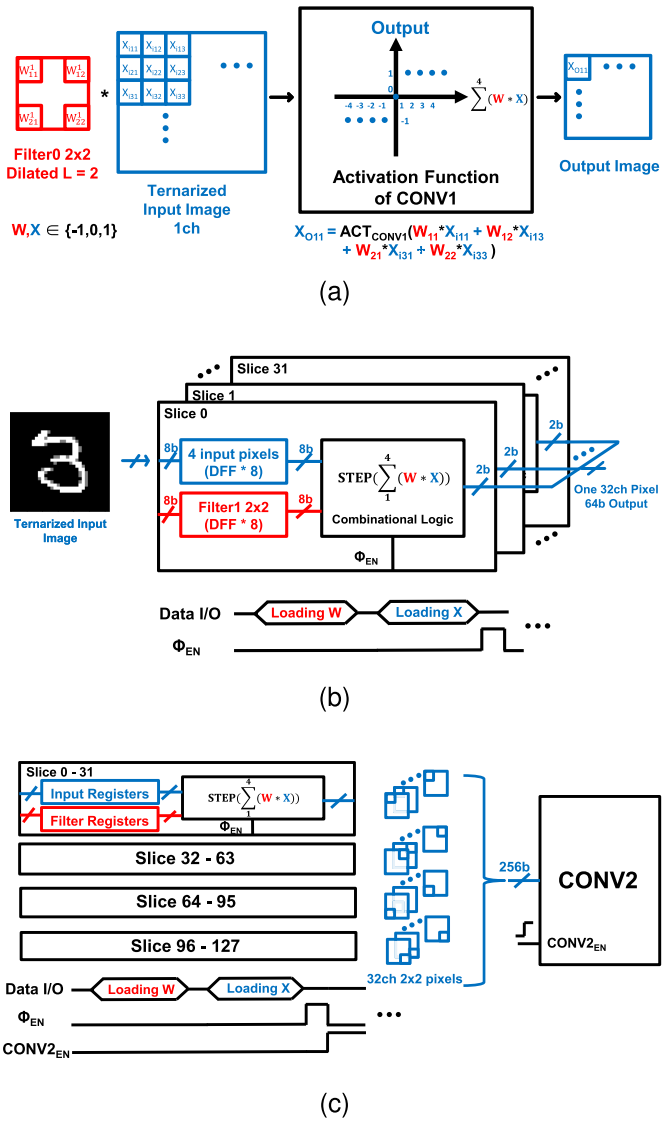


Fig. 4. (a) MAC calculation and activation function in CONV1. (b) Implementation of one channel in CONV1. (c) Structure of stack-4 in CONV1.

passes it into a set of 64-bit registers for temporary storage. The structure from CONV2 output to CONV3 input is shown in Fig. 6. Four sets of 64-b registers are placed in front of the max-pooling logic. It downsamples a 2 × 2 patch into one pixel. When one channel of CONV2 MAC calculation and activation is completed, the output image pixel is latched at the comparator output and stored in one of the 4-D flip-flops. The CONV2 output is accumulated four times with ϕ_0–ϕ_3 and is computed by the max-pooling logic. An image SRAM follows the max-pooling layer with a size of 64-b wide, 1352 B, for storing an entire frame of CONV3 input activations. CONV3 is implemented the same way as CONV2, and a 256-b register provides its input elements through crossbar multiplexers. Half of the 2 × 2 CONV3 input activation window is reused during the sliding window convolution by exchanging the two columns, thus saving the memory read energy by 50%. For CONV3, the data flow is the same as CONV2, and the 12 × 12 × 32 output image is downsampled to 6 × 6 × 32 for the FC layer.

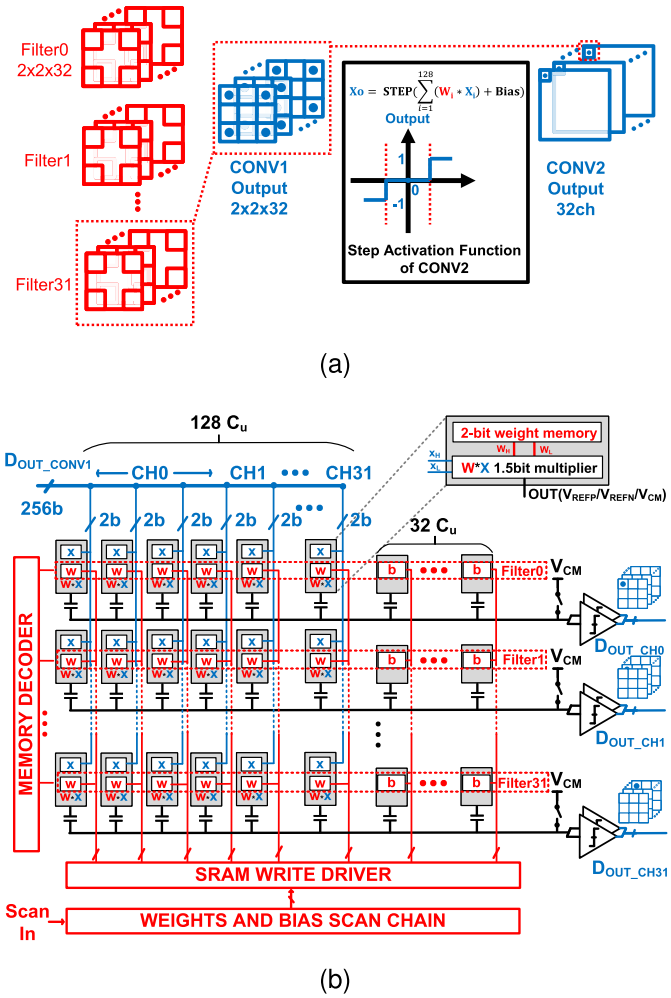


Fig. 5. (a) Convolution, MAC calculation, and the activation function in CONV2. (b) Architecture of CONV2, including SC neurons, SRAM write circuitry, and comparators for tri-level quantization. The input feature is broadcast across 32 parallel channels.

C. Architecture of FC Layer

An example of the convolution in the FC layer is shown in Fig. 7(a). The 32-channel input image is flattened and multiplied with weights of each category (number 0∼9), and the one with the highest multiplication result represents the classifier output. Fig. 7(b) illustrates the architecture of the FC layer. The FC input image is accumulated in the activation registers one row at a time. Thirty-six rows of synapses are designed to store the flattened input image, and each row represents one 32-channel image pixel. Each synapse in the FC layer has a 2-bit activation register and 20-bits weight memory shown in Fig. 7(c). After the previous max-pooling layer gets activated 36 times, a total of 1152 pixels are loaded to the activation register shown in Fig. 7(c). All weight memory for number 0∼9 is preloaded near the multipliers and then selected sequentially. The FC classification procedure is shown in Fig. 7(d). In the first cycle, the voltage representing number "0" redistributed and stored on C_1. The weights for number "1" are selected and the neurons act again, leaving the resulting voltage on C_2. Based on the comparison results, C_1 or C_2 with higher voltage is kept, and the other one is reset (RST) for storing the MAC result of the following number. After

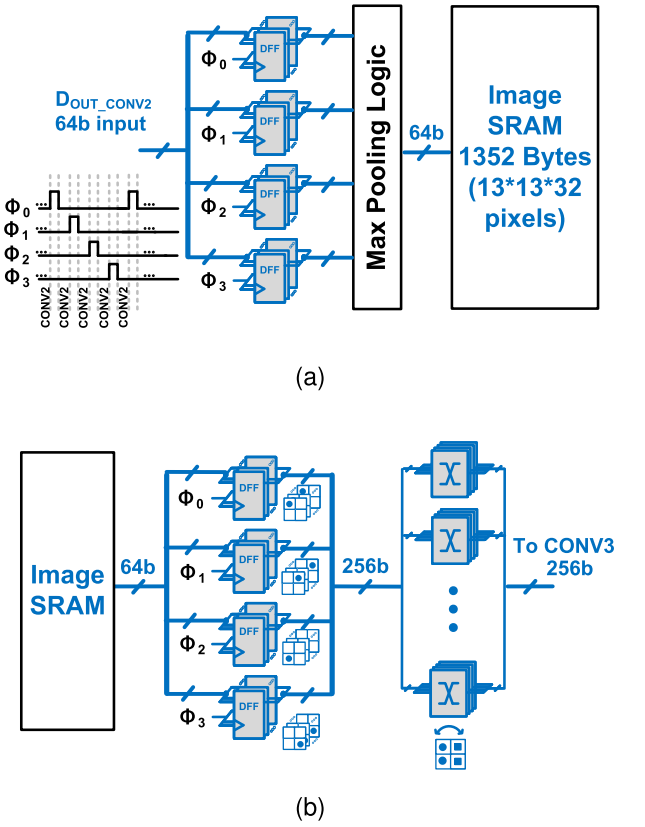


Fig. 6. (a) CONV2 output is accumulated four times and stored in SRAM after the downsampling of the max-pooling layer. (b) Read-out circuit of CONV3 input features.

nine comparisons, the final classification result is chosen as the number with the largest voltage on C_1/C_2.

III. Design and OPs of Ternary Switch-Capacitor Neuron

The design of the SC neuron is shown in Fig. 8. The fully-differential circuit consists of 320 synapses and unit capacitors for computing the convolution of 128 input activations and their corresponding weights. 20% of the total capacitance is used for biasing, and they are separated into 32 unit capacitors. The convolution sequence and the detailed design of each building block are discussed below.

A. Operating Sequence

Before the convolution begins, the weights are loaded into the synapse SRAM cells from the write driver. Then the top and bottom plates of the CDAC are connected to V_{CM} by asserting the RST line to clear all the charges. After pulling down the RST signal, the ideal MAC result is represented by the differential voltage on the capacitor top plates

$$\frac{V_{\mathrm{INP}} - V_{\mathrm{INN}}}{V_{\mathrm{REFP}} - V_{\mathrm{REFN}}} = \frac{C_u}{C_{\mathrm{total}}} \times \left(\sum_{i=1}^{128} (w_i \times x_i) + \sum_{j=1}^{32} \mathrm{bias}_j \right). \tag{2}$$

In this equation, w_i, x_i, and bias_j take on ternary values

$$w_i, x_i, \mathrm{bias}_j \in \{-1, 0, +1\}. \tag{3}$$

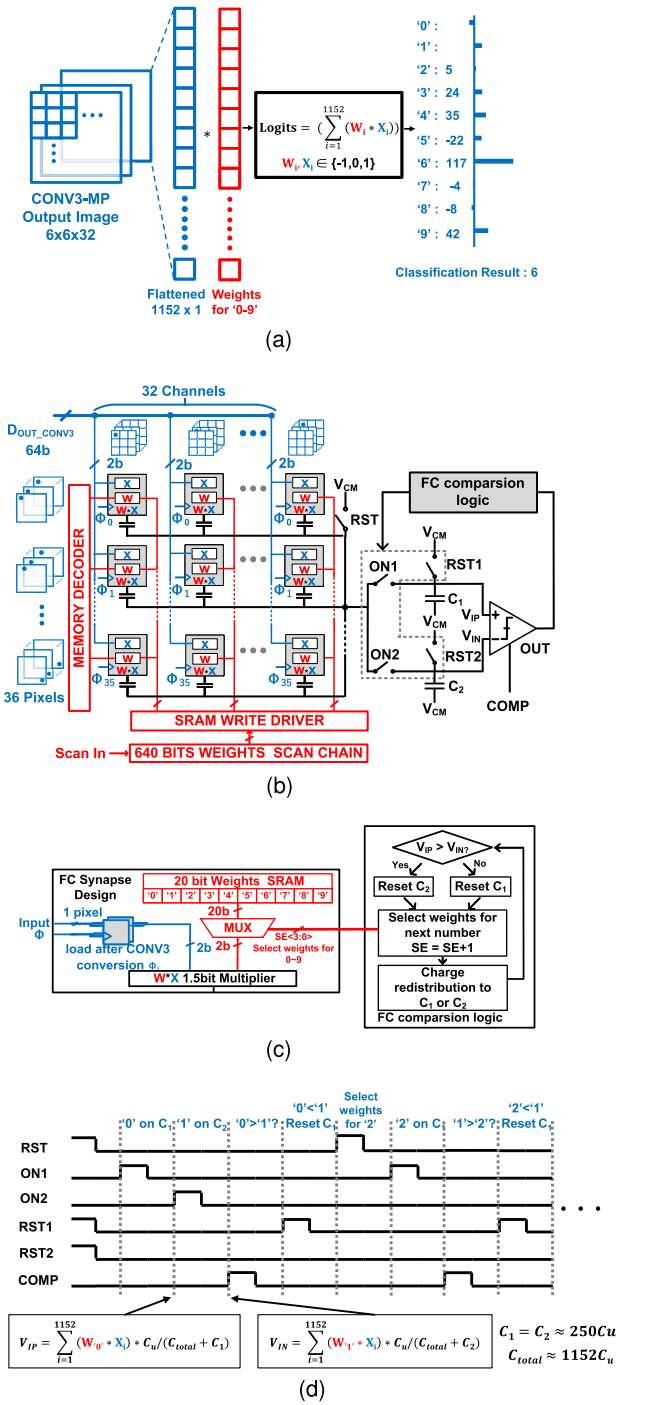


Fig. 7. (a) MAC calculation in the FC layer. (b) Architecture of the FC layer (only single-end is shown), including 1152 synapses and a comparator. (c) Design of the synapse in the FC layer and the comparison logic. (d) OP sequence of the FC layer.

After the weighted sum of multiplication results from filters and input pixels are computed by charge distribution, the clocked comparators are activated to perform the tri-level quantization. The 2-b result is latched then at the comparator output ports for the following the max-pooling layer.

B. Synapse Design

Each synapse, the unit cell of MAC elements, consists of two 6T SRAM cells and a local 1.5-b multiplier. The layout of a synapse is shown in Fig. 9. The synapse is designed with maximum density by routing the SRAM and logic gates with M1–M3, while M4–M6 above the transistors are used for the CDAC. A shielding layer is implemented on M4, and the 3.5 fF unit capacitor is formed by metal fingers on M5 and M6, with its top plate routed on M6 to minimize the parasitic capacitance. The height of the synapse is designed to match two rows of a standard cell. The size is designed to be 2.75 $\times$ 7.5 μm.

The local 1.5-b multiplier takes 2-b activation inputs from the 256-b data bus and the weight inputs from the two SRAM cells. The two standard 6T SRAM cells are directly connected to computation logic without a read-out circuit. They remain stationary during inference, amortizing the power from the charging bitline. The logic gates in the multiplier calculate the 1.5-b multiplication results and select the proper output voltage on the 1.5-b CDAC. The ternary value 0/+1/−1 is coded as 0X/10/11. Fig. 10 shows how this way of encoding translates to hardware simplicity. The 1.5-b multiplication is performed efficiently by one AND and XOR. Then the 1.5-b CDAC only needs one more AND and OR to select one of the tri-level voltages for the output. Due to the mismatch of the MOM capacitors, the unit capacitance for each synapse is not the same. Monte-Carlo simulation of the capacitor mismatch is shown in Fig. 11(a). The design point of the unit capacitor value features a 0.37% mismatch. The mismatch variation is included in the Python neural network model. Simulation results in Fig. 11(b) shows the capacitor array mismatch does not affect the classification accuracy.

C. Comparator Design

Two comparators with positive/negative thresholds perform the ternary activation function. The strong-arm latch-based comparator with two input pairs and offset cancellation digital-to-analog converter (DAC) is shown in Fig. 12(a) [24]. One of the input pairs takes the differential voltage $V_{\mathrm{IN}} = V_{\mathrm{INP}} - V_{\mathrm{INN}}$ from the charge summing nodes. The other one $V_{\mathrm{STEP+}}$ and $V_{\mathrm{STEP-}}$ is generated from off-chip DAC as the activation thresholds. $V_{\mathrm{STEP+}}$ and $V_{\mathrm{STEP-}}$ remain constant during the whole inference without driving a low-impedance load. They can be provided by low-power resistor ladder or CDAC if implemented on-chip. Ideally, the transfer function from the comparators is

$$\mathrm{ACT}_{\mathrm{CONV2}}(V_{\mathrm{IN}}) = \begin{cases} 1, & \text{if } V_{\mathrm{IN}} > V_{\mathrm{STEP+}} - V_{\mathrm{STEP-}} \\ -1, & \text{if } V_{\mathrm{IN}} < V_{\mathrm{STEP-}} - V_{\mathrm{STEP+}} \\ 0, & \text{otherwise.} \end{cases}$$

Illustrated in Fig. 12(b), the comparators do not resolve perfect activation thresholds in the presence of offset. According to the Monte-Carlo simulation, they exhibit an 8.1-mV rms offset which translates to 5.6 LSB. The effect of comparator offsets on classification accuracy is shown in Fig. 12(c). To suppress the offsets within 1 LSB, thus achieving an accurate activation function, one-time foreground calibration is realized by creating unbalanced capacitor loading. To ensure the calibration CDAC fully covers the maximum range of comparator offset, the CDAC is sized to be capable of creating a 4σ offset, −32 to 32 mV. To guarantee an accurate activation

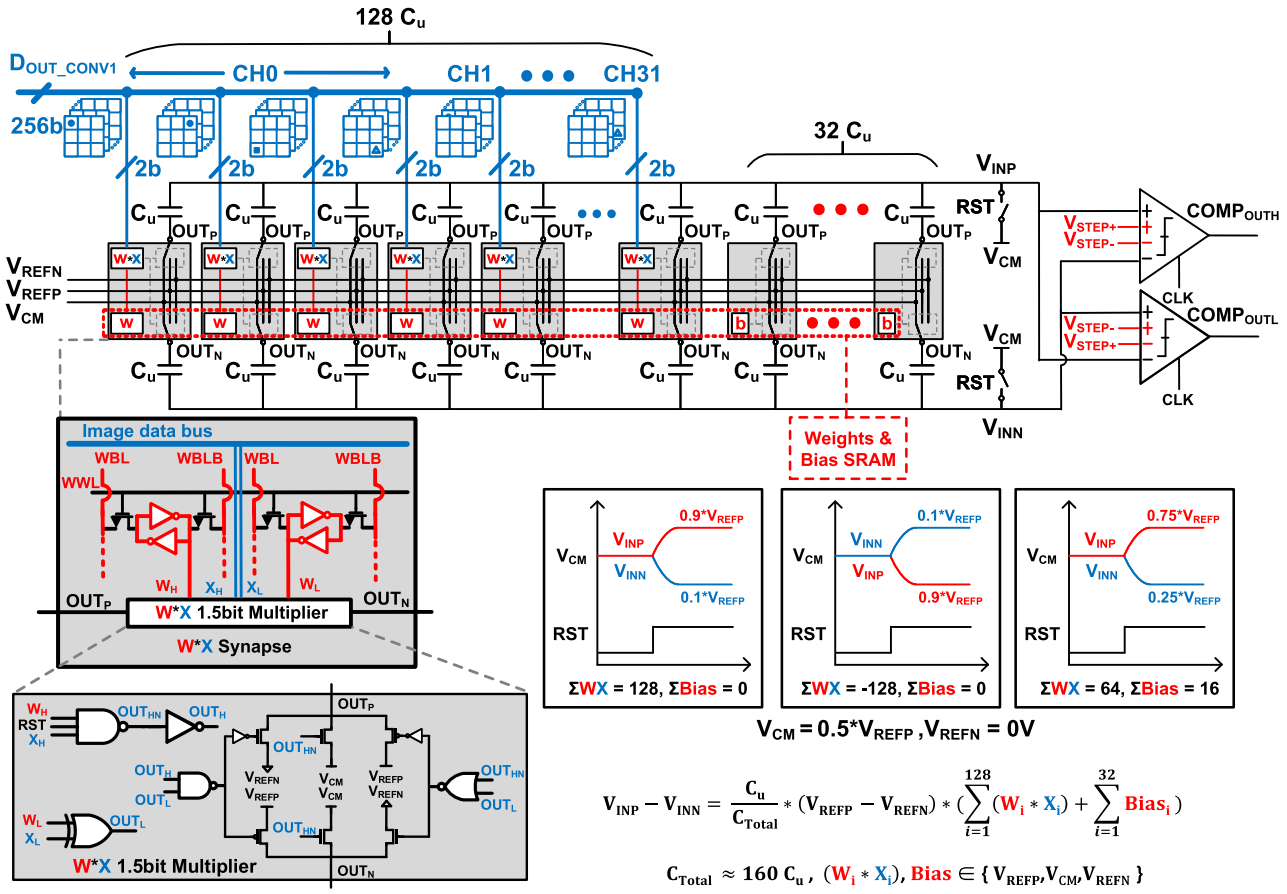


Fig. 8. 1.5-b SC neuron implemented with V_{CM}-based switching scheme. Summation is mapped to charge redistribution and two comparators perform tri-level quantization.

threshold, the minimum step voltage of the DAC needs to be lower than the LSB voltage of the MAC array. Therefore, each calibration CDAC is split to 5-b resolution, 1 mV step size. Fig. 12(d) describes the calibration process. At the beginning of the calibration procedure, the comparator input terminals are connected to V_{CM}. Then the calibration code is sent into the 10-b SRAM, starting from the minimum value. After each set of code is written to the CDAC, the comparator is activated 1000 times to provide an estimation of output probability. When it gives out about the same amount of 0's and 1's, the calibration is marked as completed, and the current offset code remains in the SRAM.

D. Comparison Between the Mixed-Signal BNN and TNN

Fig. 13(a) shows a simplified structure of binary SC neurons. In this case, +1/−1 are mapped as V_{REFP} and V_{REFN} in the voltage domain, and the two-level quantization is done by one comparator. For proposed tri-level computation in Fig. 13(b), 0/+1/−1 are represented by V_{CM}, V_{REFP}, and V_{REFN}, respectively. In the ternary synapse, zero input weight or activation means no switching activity after RST, while nonzero multiplication result drives the capacitor to switch from V_{CM} to V_{REFP} or V_{REFN}. In terms of energy/OP, the introduction of V_{CM} reduces voltage swing on CDAC to half rail-to-rail, providing 31% MAC power saving than two-level CDAC based on simulation. Furthermore, the fully differential

TABLE I
BNN MODEL FOR 98% MNIST ACCURACY

Layer	Type	Size	Channel	Filter Size
1	CONV-BN	30×30	1(input)	2×2
2	CONV-BN	28×28	128	
2p	MAX-POOL	26×26		
3	CONV-BN	13×13	64	
3p	MAX-POOL	12×12		
4	FC	(Flatten 6×6×64) 2304→10		

SC neuron benefits from the constant common-mode voltage at the comparator inputs, but the BNN implementation in [15] requires an extra common-mode setting section. Fig. 13(c) shows a comparison of the charge-domain BNN and TNN. The computation circuits are simple in both cases. The synapse of BNN is implemented with four logic gates [15], while the TNN synapse consists of four logic gates and six more transistor switches. To illustrate the OPs/inference reduction in TNN, we conduct a baseline test by building BNN and TNN models targeting the same 98% accuracy on the MNIST dataset. The topology is shown in Tables I and II.

The TNN model only takes 3.57×10^7 OPs for each classification task, while the BNN model consumes 1.38×10^8 OPs for the same level of accuracy. In this case, the TNN model benefits from 75% OPs/inference reduction without accuracy loss.

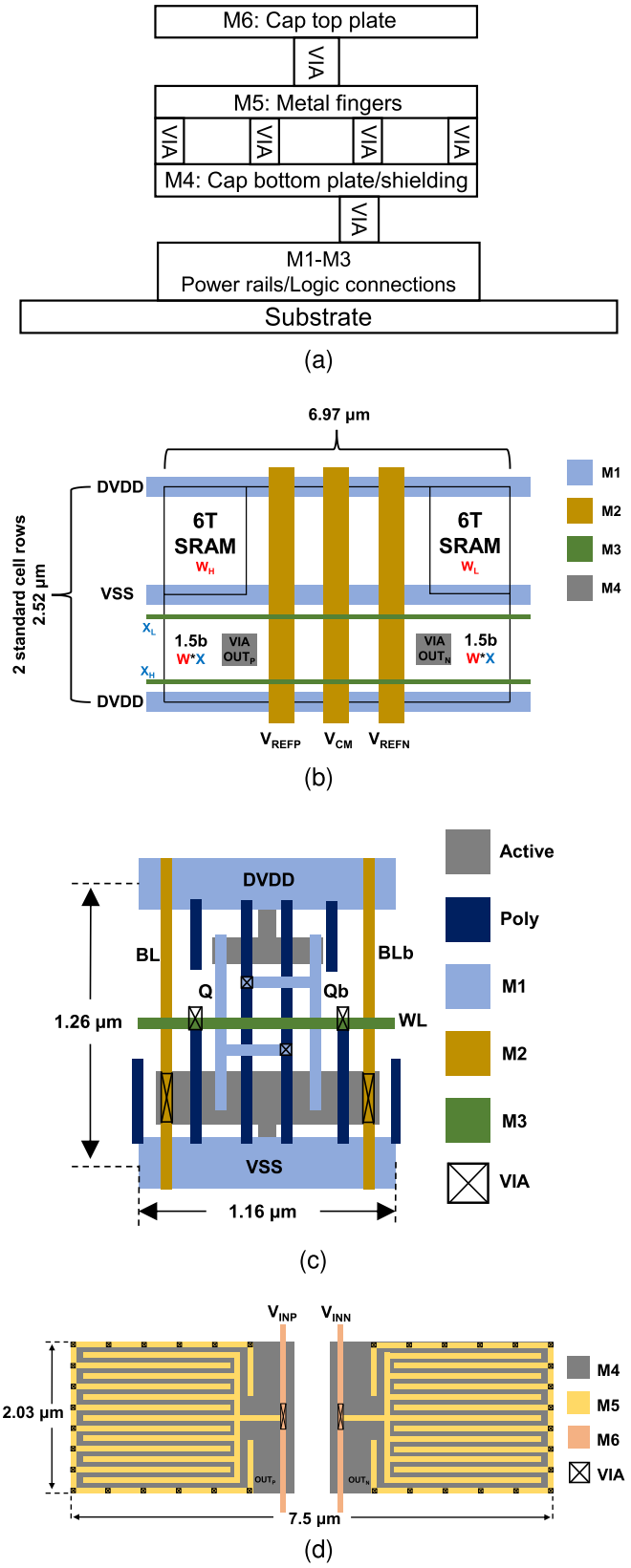


Fig. 9. (a) Cross section view of the synapse layout. The layout of a (b) ternary synapse and (c) 6T SRAM cell. (d) Capacitor layout.

In addition, extra sparsity can be introduced during training by enforcing more zero weights, resulting in further switching activity reduction. Fig. 14 shows a tradeoff among sparsity, classification accuracy, and normalized energy consumption of a ternary SC neuron.

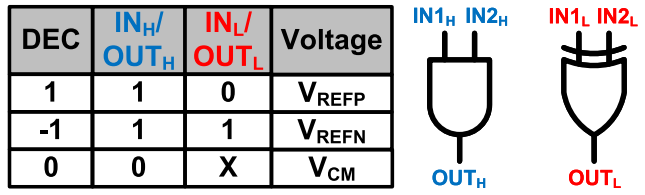


Fig. 10. Encoding of the ternary values and the 1.5-b multiplier.

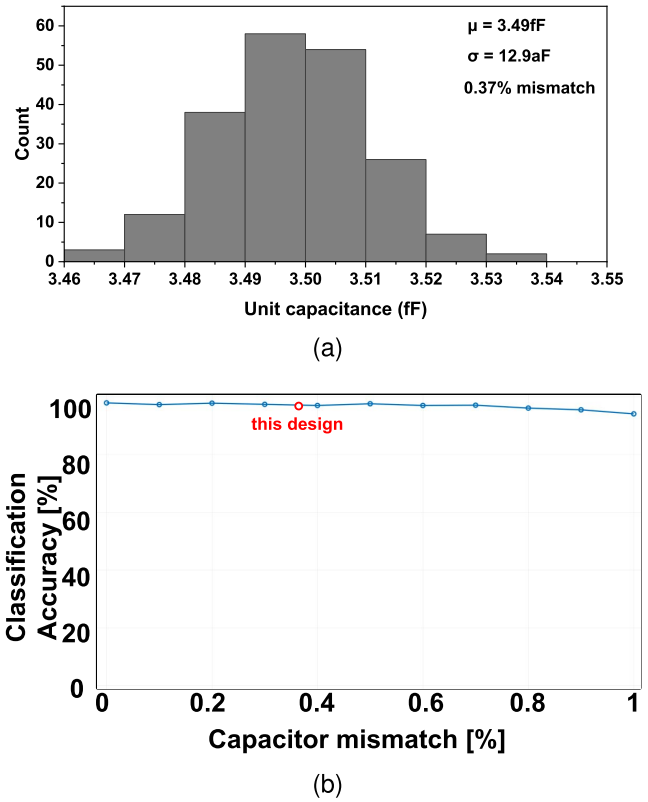


Fig. 11. (a) Capacitor mismatch simulation. (b) Classification accuracy versus capacitor mismatch.

TABLE II
TNN MODEL FOR 98% MNIST ACCURACY

Layer	Type	Size	Channel	Filter Size
1	CONV-TN	30×30	1(input)	2×2
2	CONV-TN	28×28	32	
2p	MAX-POOL	26×26		
3	CONV-TN	13×13		
3p	MAX-POOL	12×12		
4	FC	(Flatten 6×6×32) 1152→10		

IV. MEASUREMENT AND SIMULATION RESULTS

A. MNIST Evaluation

Fig. 15 shows a die photograph of the chip. The prototype is fabricated in 40-nm LP CMOS and occupies an active area of 0.96 mm^2. The measurement setup is shown in Fig. 16. After the neural network model is trained on Tensorflow, all the weights, biases, and activation thresholds are exported to an SD card, which is then picked up by a microcontroller.

TABLE III
COMPARISON TABLE

	This work		JSSC'21 X. Si [10]	JSSC'19 D. Bankman [15]	JSSC'22 J.-W. Su [22]	JSSC'22 J.-W.Su [23]	JSSC'19 H. Valavi [16]
Technology	**40nm**		28nm	28nm	28nm	28nm	65nm
Area(mm^2)	**0.98**		0.324	4.6	N/A	0.468	12.6
Area Eff.(GOPS/mm^2)	**469**[1]		N/A	67	N/A	1640	1498
Operating VDD(V)	**0.8/0.7/0.9**		0.7-0.9	0.8/0.6	0.85-1	N/A	0.94/0.68/1.2
Energy Eff.(TOPS/W)	**556**[2]		273.76[4]	532	122.2[4]	377.24[4]	866
Bit Precision	**1.5-b**		1-8b	1b	1-8b	1-8b	1b
Dataset	**MNIST**		CIFAR-10	CIFAR-10	CIFAR-10	CIFAR-10	MNIST
Accuracy (chip / software)	**97.1%**[3]/97.9%		92.02%/N/A	86.05%/88.6%	91.94%/N/A	91.85%/N/A	98.6%/98.92%
FPS	**549**		N/A	237	N/A	N/A	651
Power(mW)	**0.096**		N/A	0.899	N/A	N/A	N/A
Operations / Inference	**TNN**	**BNN**	N/A	N/A	N/A	N/A	5.3×10^8
	$\mathbf{3.57\times10^7}$	$\mathbf{1.38\times10^8}$					
Avg. MACs Energy/Inference	**0.09μJ**	**0.52μJ**	N/A	N/A	N/A	N/A	0.8μJ
Avg. Total Energy/Inference	**0.18μJ**	**0.7μJ**	N/A	3.8μJ	N/A	N/A	N/A
Sparsity-aware	**Yes**		No	No	No	No	No
All operations on chip	**Yes**		No	Yes	No	No	No

[1] Based on SC neuron
[2] Based on MACs energy efficiency
[3] 10 runs average on 10,000 test set images
[4] Normalized to 1b

The microcontroller (STM32H753) is used on the characterization printed circuit board (PCB) for programming the weights into on-chip memory, controlling a thresholds generation DAC (AD5669R), calibrating the comparator offsets, plotting output activations for debugging, and triggering the field-programmable gate array (FPGA) (EP4CE10) to generate critical control signals. To identify and analyze the error source, the TNN model is reconstructed in the microcontroller. It runs the inference parallel with the prototype to compare the ideal and measured output.

Fig. 17(a) shows the energy breakdown. Analog VDD (AVDD) powers comparators, and digital VDD (DVDD) powers the supply for SRAM and digital circuits, including CONV1, max-pooling layer, and control logic. V_{REFP} and V_{CM} serve as the supply for SC neurons, including the CDACs and synapses. The accuracy evaluation on the MNIST dataset is entirely from measurement. As shown in Fig. 17(b), randomly selected pictures are sent into the prototype from the FPGA for power consumption measurement. A high-speed oscilloscope is used to capture the supply current waveform through serial resistors on different power domains. For MNIST classification, this chip operates at 549 FPS with 0.7 V DVDD, 0.8 V AVDD, 0.9 V V_{REFP}, and 0.45 V V_{CM}, leading to an average MAC energy of 0.18 μJ/classification. The measured classification accuracy is 97.1%, which is 0.8% lowered than the ideal software model due to circuit noise, mismatch, and charge leakage.

B. Comparison

Table III compares this work with prior arts. This work efficiently realizes the wide vector summation in the charge domain. In [10] and [22], the MAC OP is achieved by discharging a bitline capacitor with a certain current. Careful design consideration or tradeoff analysis is needed to ensure stability and PVT variation suppression. While in our proposed work, mapping the MAC OP into charge redistribution benefits from the accurate matching of MOM capacitors in modern CMOS technology. Compared to the designs in [15], [16], and [23] with charge-sharing MAC OP, the proposed ternary synapse features an inherently fully differential architecture. It eliminates the concern of undefined input common-mode voltage of the comparator or sense amplifier. Compared to [15] and [16] using BNN, the proposed TNN model benefits from fewer OPs/inference and less switching activity. Moreover, this work performs all OPs on-chip, while [10], [16], [22], and [23] have only MAC OP.

TABLE IV
CIFAR-10 EVALUATION

Dataset	CIFAR-10	
Model	VGG-style	
Quantization	BNN	TNN
Simulated accuracy	83.13%	87.70%
Avg. MACs Energy/Inference (calculation)	1.55μJ	1.18μJ

TABLE V
4-bit ENCODING

DEC	$X/W_{[3:2]}$	$X/W_{[1:0]}$	BIN
-4	-1	-1	1111
-3	-1	0	110X
-2	-1	1	1110
-1	0	-1	0X11
0	0	0	0X0X
1	0	1	0X10
2	1	-1	1011
3	1	0	100X
4	1	1	1010

C. CIFAR-10 Evaluation

As the neural network architecture is dedicated to MNIST classification, the prototype is not capable of performing all OPs on-chip for the Canadian Institute For Advanced Research-10 (CIFAR-10) dataset. The CIFAR-10 classification result is obtained from a measurement based simulation on

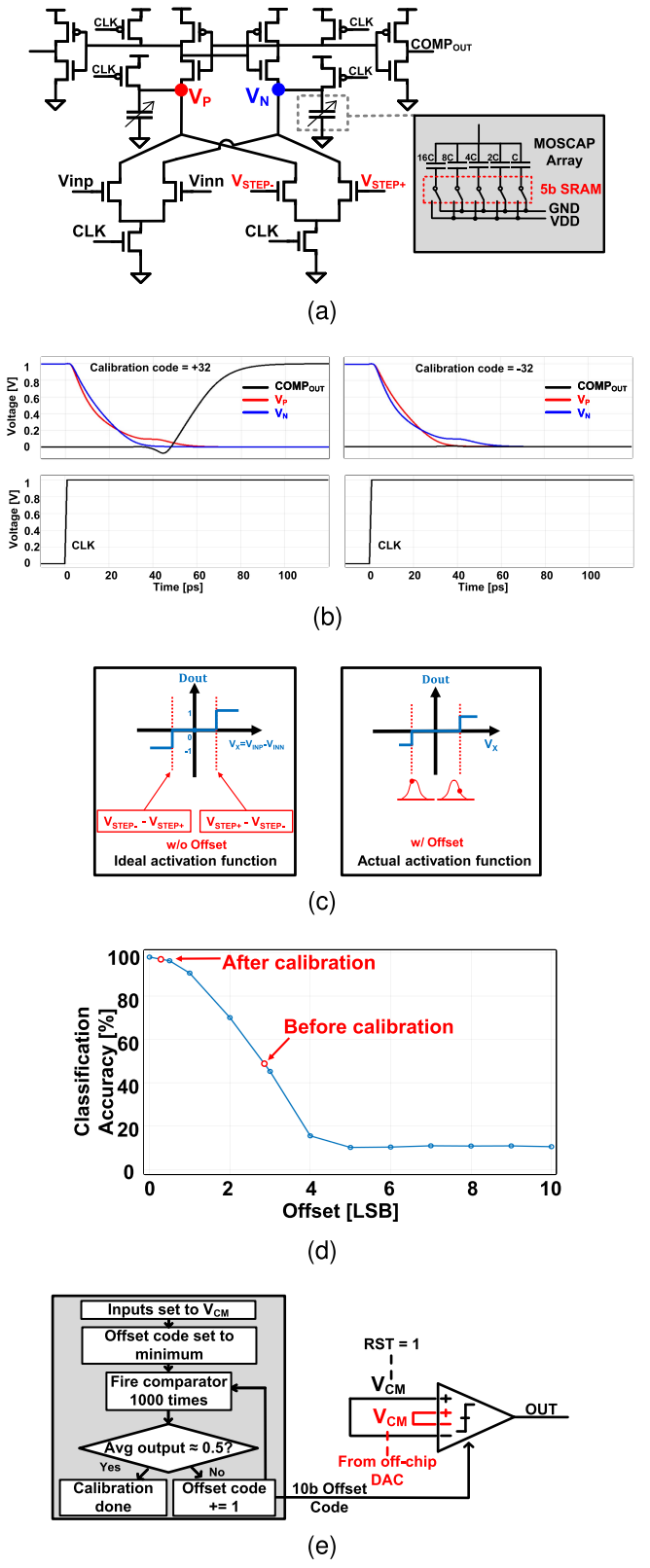
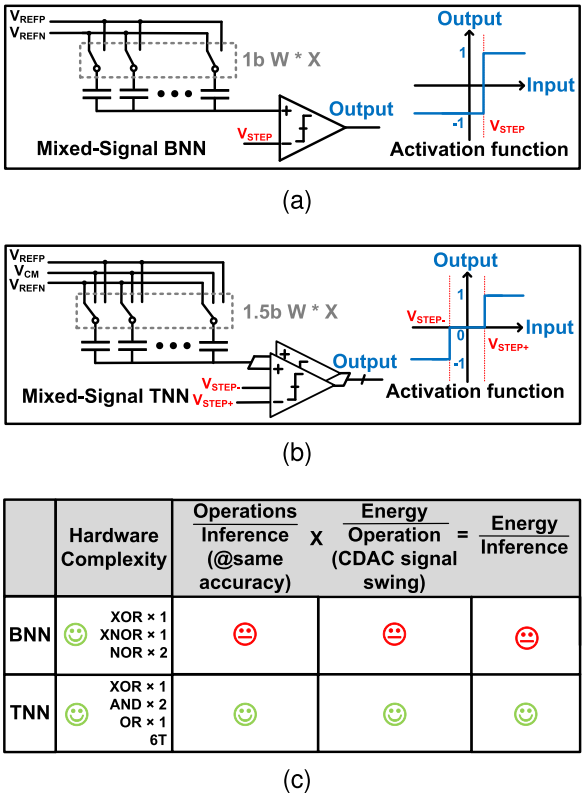


Fig. 12. (a) Strong-arm latch comparator with offset cancellation DAC. (b) Comparator waveform with all inputs connected to V_{CM}. (c) Activation function affected by comparator offsets. (d) Simulation of classification accuracy versus comparator offset. (e) Offset calibration procedure.



	Hardware Complexity	Operations / Inference (@same accuracy)	X	Energy / Operation (CDAC signal swing)	=	Energy / Inference
BNN	XOR × 1 XNOR × 1 NOR × 2	😐		😐		😐
TNN	XOR × 1 AND × 2 OR × 1 6T	🙂		🙂		🙂

(c)

Fig. 13. (a) BNN SC neuron. (b) TNN SC neuron. (c) Comparison of charge-domain BNN and TNN.

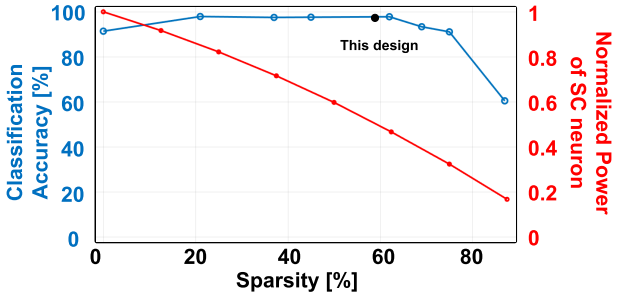


Fig. 14. Tradeoff between the percentage of 0 s, normalized power consumption, and top-1 classification accuracy.

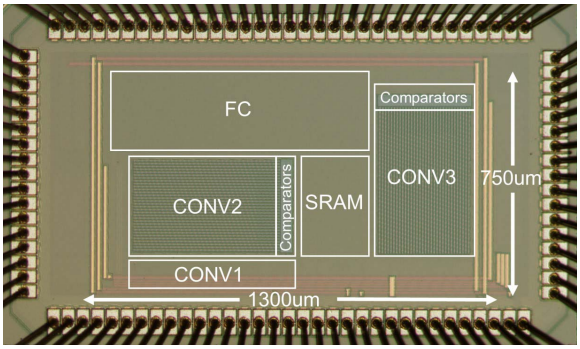


Fig. 15. Die photograph.

a Visual Geometry Group (VGG) style neural network as shown in Fig. 18. The VGG network has six convolutional layers and three FC layers. The MAC results from all convolutional layers are simulated from a 1152-synapse ternary network macrobuilt in Python. The 3 × 3 × 128 activations in a 128-channel convolutional window are mapped to the

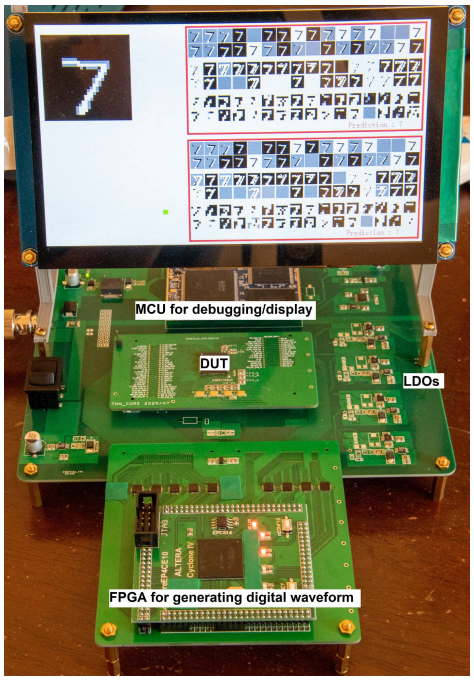
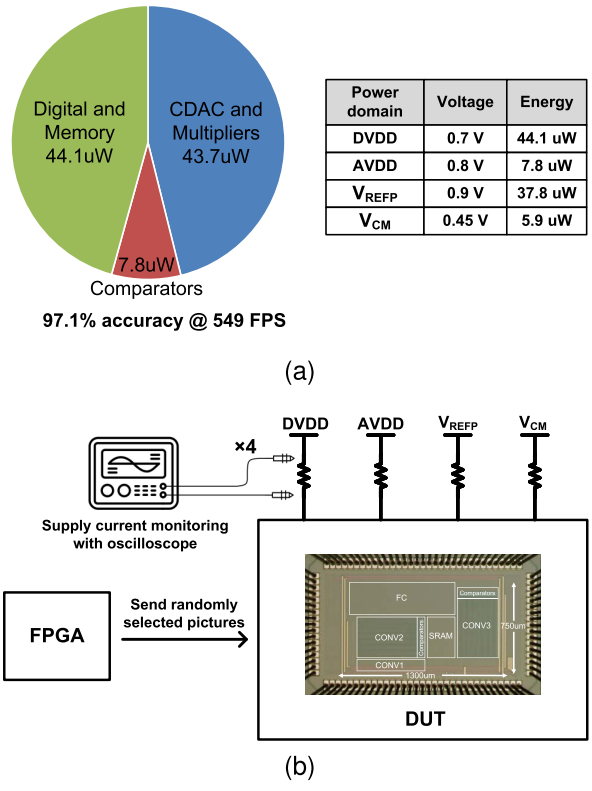


Fig. 16. Measurement setup.



Power domain	Voltage	Energy
DVDD	0.7 V	44.1 uW
AVDD	0.8 V	7.8 uW
V_{REFP}	0.9 V	37.8 uW
V_{CM}	0.45 V	5.9 uW

Fig. 17. (a) Power breakdown. (b) Power consumption measurement setup.

1152 synapses. To process the 256 channels per pixel in L5 and L6 with 128 neurons, each 256-channel convolution is divided into two 128-channel groups. For comparison with the binary network model, we evaluated the accuracy of CIFAR-10 with binary resolution using the same neural network topology. Table IV summarizes the simulated accuracy and MAC energy consumption of convolutional layers. The NN with ternary

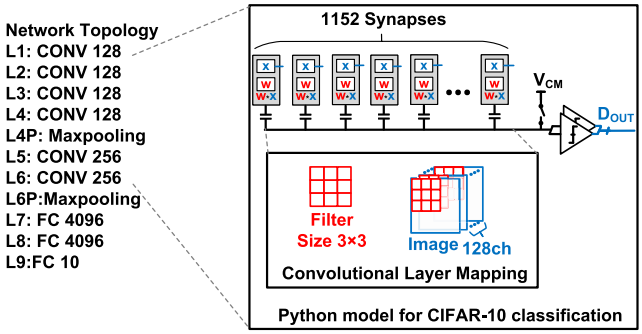


Fig. 18. Simulation-based CIFAR-10 evaluation.

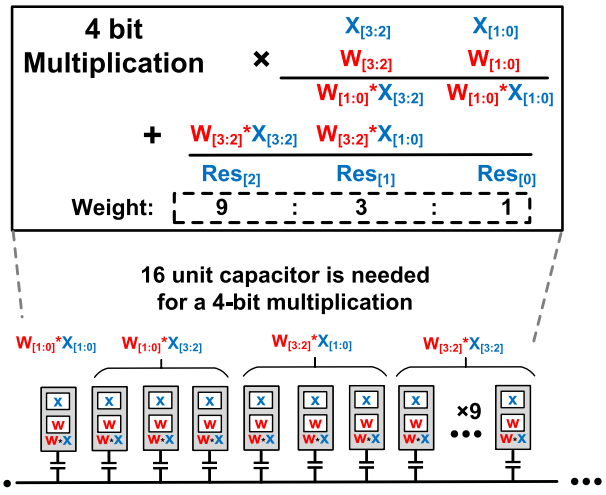


Fig. 19. Example of 4-bit extension using the proposed ternary synapse.

accuracy demonstrates a 4.57% higher accuracy over the binary model, while consuming 31% less MAC energy.

D. Scalability for Multibit Extension

In this section, we discuss a design approach to extending the proposed ternary synapse to multibit resolution. To support a range of −4 to +4 for activations and weights, 4-bit encoding is needed as shown in Table V. The idea is illustrated in Fig. 19. The multiplication of 4b-by-4b MAC result can be broken into four partial ternary products. The four partial products have the weights of 9:3:1. They can be merged by

$$\begin{aligned} \mathrm{MAC}_{\mathrm{result}} = 9 \times \left(\sum W_{[3:2]} \times X_{[3:2]}\right) \\ \times 3 \times \left(\sum W_{[3:2]} \times X_{[1:0]} + \sum W_{[1:0]} \times X_{[3:2]}\right) \\ + 1 \times \left(\sum W_{[1:0]} \times X_{[1:0]}\right). \end{aligned} \tag{4}$$

In this way, the 4-bit multiplication can be mapped into the charge sharing of 16 unit capacitors. As the comparators in the prototype only resolve 1.5-b output resolution, this approach would require an ADC with 4-b or resolution on the charge-sharing node to provide enough accuracy for output activations.

V. Conclusion

This article proposed a charge-domain computing TNN accelerator with a fully on-chip neural network. The four-layer

neural network model is organized in a pipeline structure, with all weight and biases memory remaining stationary during the whole inference. By mapping the 1.5-b MAC calculation into V_{CM}-based capacitor switching scheme, the power of MAC calculation is reduced by 31% compared to binary charge-domain computing. Evaluated on the MNIST dataset, the TNN model features 4× fewer OPs/inference than the BNN model for the same accuracy level, leading to a 75% reduction of total energy/inference. The proposed chip consumes only 0.18 μJ total energy for MNIST classification, which is the smallest to our best knowledge for comparable classification accuracy.

ACKNOWLEDGMENT

The authors thank the TSMC University Shuttle Program for chip fabrication

REFERENCES

[1] P. Rajpurkar, A. Y. Hannun, M. Haghpanahi, C. Bourn, and A. Y. Ng, "Cardiologist-level arrhythmia detection with convolutional neural networks," 2017, *arXiv:1707.01836*.

[2] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, vol. 25. Stateline, NV, USA, Dec. 2012, pp. 1097–1105.

[3] P. Rajpurkar, A. Y. Hannun, M. Haghpanahi, C. Bourn, and A. Y. Ng, "Cardiologist-level arrhythmia detection with convolutional neural networks," 2017, *arXiv:1707.01836*.

[4] Y. Chen, T. Krishna, J. Emer, and V. Sze, "Eyeriss: An energy efficient reconfigurable accelerator for deep convolutional neural networks," *IEEE J. Solid-State Circuits*, vol. 52, no. 1, pp. 127–138, Jan. 2017.

[5] Y.-H. Chen, T.-J. Yang, J. Emer, and V. Sze, "Eyeriss v2: A flexible accelerator for emerging deep neural networks on mobile devices," 2018, *arXiv:1807.07928*.

[6] M. Horowitz, "1.1 computing's energy problem (and what we can do about it)," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2014, pp. 10–14.

[7] Z. Jiang, S. Yin, M. Seok, and J.-S. Seo, "XNOR-SRAM: In-memory computing SRAM macro for binary/ternary deep neural networks," in *Proc. IEEE Symp. VLSI Technol.*, Jun. 2018, pp. 173–174.

[8] A. Biswas and A. P. Chandrakasan, "Conv-RAM: An energy-efficient SRAM with embedded convolution computation for low-power CNN-based machine learning applications," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2018, pp. 488–490.

[9] J. Zhang, Z. Wang, and N. Verma, "In-memory computation of a machine-learning classifier in a standard 6T SRAM array," *IEEE J. Solid-State Circuits*, vol. 52, no. 4, pp. 915–924, Apr. 2017.

[10] X. Si et al., "A local computing cell and 6T SRAM-based computing-in-memory macro with 8-b MAC operation for edge AI chips," *IEEE J. Solid-State Circuits*, vol. 56, no. 9, pp. 2817–2831, Sep. 2021.

[11] I. Hubara, M. Courbariaux, D. Soudry, R. El-Yaniv, and Y. Bengio, "Quantized neural networks: Training neural networks with low precision weights and activations," *J. Mach. Learn. Res.*, vol. 18, no. 1, pp. 6869–6898, 2017.

[12] X. Lin, C. Zhao, and W. Pan, "Towards accurate binary convolutional neural network," 2017, *arXiv:1711.11294*.

[13] M. Courbariaux, I. Hubara, D. Soudry, R. El-Yaniv, and Y. Bengio, "Binarized neural networks: Training deep neural networks with weights and activations constrained to +1 or -1," 2016, *arXiv:1602.02830*.

[14] V. Tripathi and B. Murmann, "Mismatch characterization of small metal fringe capacitors," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 61, no. 8, pp. 2236–2242, Aug. 2014.

[15] 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," *IEEE J. Solid-State Circuits*, vol. 54, no. 1, pp. 158–172, Jan. 2019.

[16] H. Valavi, P. J. Ramadge, E. Nestler, and N. Verma, "A 64-tile 2.4-Mb in-memory-computing CNN accelerator employing charge-domain compute," *IEEE J. Solid-State Circuits*, vol. 54, no. 6, pp. 1789–1799, Jun. 2019.

[17] H. Jia et al., "A programmable neural-network inference accelerator based on scalable in-memory computing," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2021, pp. 236–237.

[18] V. Sze, Y.-H. Chen, T.-J. Yang, and J. S. Emer, "How to evaluate deep neural network processors: TOPS/W (alone) considered harmful," *IEEE Solid State Circuits Mag.*, vol. 12, no. 3, pp. 28–41, Aug. 2020.

[19] S. Garg, A. Jain, J. Lou, and M. Nahmias, "Confounding tradeoffs for neural network quantization," 2021, *arXiv:2102.06366*.

[20] X. Yang et al., "An in-memory-computing charge-domain ternary CNN classifier," in *Proc. IEEE Custom Integr. Circuits Conf. (CICC)*, Apr. 2021, pp. 1–2.

[21] F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," 2015, *arXiv:1511.07122*.

[22] J.-W. Su et al., "Two-way transpose multibit 6T SRAM computing-in-memory macro for inference-training AI edge chips," *IEEE J. Solid-State Circuits*, vol. 57, no. 2, pp. 609–624, Feb. 2022.

[23] J.-W. Su et al., "A 8-b-precision 6T SRAM computing-in-memory macro using segmented-bitline charge-sharing scheme for AI edge chips," *IEEE J. Solid-State Circuits*, pp. 1–16, Aug. 2022.

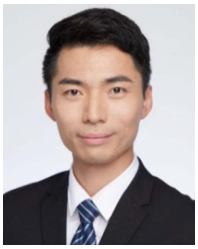
[24] B. Razavi, "The StrongARM latch [a circuit for all seasons]," *IEEE Solid State Circuits Mag.*, vol. 7, no. 2, pp. 12–17, Jun. 2015.

Xiangxing Yang (Member, IEEE) received the B.S. degree in electronics engineering from the University of Electronic Science and Technology of China, Chengdu, China, in 2016, and the M.S. and Ph.D. degrees from the Department of Electrical and Computer Engineering, The University of Texas at Austin, Austin, TX, USA, in 2021 and 2022, respectively.

He is currently with pSemi Corporation, Austin. His research interests include analog and mixed-signal circuit design for edge computing.

Keren Zhu (Member, IEEE) received the B.S. degree in electrical engineering from the University of Wisconsin–Madison, Madison, WI, USA, in 2016, and the Ph.D. degree from The University of Texas at Austin (UT-Austin), Austin, TX, USA, in 2022.

He is currently a Post-Doctoral Research Fellow at the Department of Electrical and Computer Engineering, UT-Austin. His research interests include physical design automation, analog integrated circuit design automation, machine learning for EDA, and computing system with emerging technologies.



Xiyuan Tang (Member, IEEE) received the B.Sc. degree (Hons.) from the School of Microelectronics, Shanghai Jiao Tong University, Shanghai, China, in 2012, and the M.S. and Ph.D. degrees in electrical engineering from The University of Texas at Austin, Austin, TX, USA, in 2014 and 2019, respectively.

He was a Design Engineer with Silicon Laboratories, Austin, from 2014 to 2017, where he was involved in the RF receiver design. He was a Post-Doctoral Researcher with The University of Texas at Austin from 2019 to 2021. He is currently an Assistant Professor with Peking University, Beijing, China. His research interests include digitally assisted data converters, low-power mixed-signal circuits, and analog data processing.

Meizhi Wang (Student Member, IEEE) received the B.S. degree in electrical engineering, the B.S. degree in system science and engineering, and the M.S. degree in electrical engineering from the Washington University in St. Louis, St. Louis, MO, USA, in 2018. She is currently pursuing the Ph.D. degree with the Electrical and Computer Engineering Department, The University of Texas at Austin, Austin, TX, USA, specializing in the integrated circuits and systems track.

Her research interests include hardware security and low-power VLSI design.

Mingtao Zhan (Student Member, IEEE) received the B.S. degree (Hons.) from the Department of Electrical Engineering, Tsinghua University, Beijing, China, in 2020, where he is currently pursuing the Ph.D. degree.

His research interests include analog and mixed-signal integrated circuit design.

Nanshu Lu (Senior Member, IEEE) received the B.Eng. degree (Hons.) from Tsinghua University, Beijing, China, and the Ph.D. degree from Harvard University, Cambridge, MA, USA.

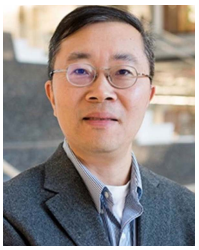
She did her Beckman Postdoctoral Fellowship at the University of Illinois at Urbana-Champaign (UIUC), Champaign, IL, USA. She is currently the Frank and Kay Reese Professor at the University of Texas at Austin (UT-Austin), Austin, TX, USA. Her research concerns the mechanics, materials, manufacture, and human or robot integration of soft electronics. For example, she has invented wearable e-tattoos for digitizing the human body and soft e-skins for robots to gain human-like sensations. She has published 110 journal articles with more than 21 000 citations. She is a Clarivate (Web of Science) highly cited researcher.

Dr. Lu is a fellow of the American Society of Mechanical Engineers (ASME). She is on the Board of Directors of the Society of Engineering Science (SES). She has been named 35 innovators under 35 by MIT Technology Review (TR 35) and iCANX/ACS Nano Inaugural Rising Star. She has received the NSF CAREER Award, the ONR and AFOSR Young Investigator Awards, and the ASME Thomas J. R. Hughes Young Investigator Award. She has been selected as one of the five great innovators on campus and five world-changing women at UT-Austin. She is currently an Associate Editor of *Journal of Applied Mechanics* and *Nano Letters*. She is on the Editorial Board of *Advanced Electronic Materials*, IEEE JOURNAL ON FLEXIBLE ELECTRONICS, and *Sensors*.

Jaydeep P. Kulkarni (Senior Member, IEEE) received the B.E. degree from the University of Pune, Pune, India, in 2002, the M.Tech. degree from the Indian Institute of Science (IISc), Bengaluru, India, in 2004, and the Ph.D. degree from Purdue University, West Lafayette, IN, USA, in 2009, all in electronics/electrical engineering.

From 2009 to 2017, he was with the Intel Circuit Research Laboratory, Hillsboro, OR, USA, where he worked on energy-efficient integrated circuit technologies. He is currently an Assistant Professor of electrical and computer engineering with The University of Texas at Austin, Austin, TX, USA, where he is also a fellow of the AMD Endowed Chair in Computer Engineering and the Silicon Labs Chair in Electrical Engineering. He has filed 35 patents and published 100 articles in refereed journals and conferences. His research is focused on machine-learning hardware accelerators, in-memory computing, emerging nanodevices, hardware security, heterogeneous/3-D integration, and cryogenic computing.

Dr. Kulkarni is a member of the Association for Computing Machinery (ACM). He received the Best M.Tech. Student Award from IISc, the Intel Foundation Ph.D. Fellowship Award, the SRC Best Paper and Inventor Recognition Awards, the Purdue Outstanding Doctoral Dissertation Award, seven Intel Divisional Recognition Awards, the 2015 IEEE TRANSACTIONS ON VERY LARGE SCALE INTEGRATION (VLSI) SYSTEMS Best Paper Award, the SRC Outstanding Industrial Liaison Award, the Micrometer Foundation Faculty Awards, the Intel Rising Star Faculty Award, and the NSF Career Award. He has served as the Conference General Co-Chair of 2018 International Symposium on Low Power Electronics and Design (ISLPED) and is participating in the technical program committees of the Custom Integrated Circuits Conference (CICC), the International Conference on Computer-Aided Design (ICCAD), the Design Automation Conference (DAC), and the International Conference on Artificial Intelligence Circuits and Systems (AICAS) conferences. He is also serving as the Chair for the IEEE Central Texas SSCS/CAS Joint Chapter. He is also serving as an Associate Editor for the IEEE SOLID-STATE CIRCUITS LETTERS, the IEEE TRANSACTIONS ON VERY LARGE SCALE INTEGRATION (VLSI) SYSTEMS, and IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS—II: EXPRESS BRIEFS.



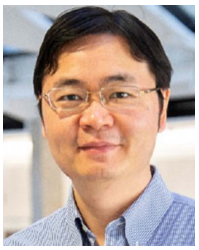
David Z. Pan (Fellow, IEEE) received the B.S. degree from Peking University, Beijing, China, in 1992, and the M.S. and Ph.D. degrees from the University of California at Los Angeles (UCLA), Los Angeles, CA, USA, in 1994 and 2000, respectively.

From 2000 to 2003, he was a Research Staff Member with the IBM T. J. Watson Research Center, Yorktown Heights, NY, USA. He is currently a Professor and a Holder of the Silicon Laboratories Endowed Chair in Electrical Engineering, The University of Texas at Austin, Austin, TX, USA. He has published over 420 journal articles and refereed conference papers. He holds eight U.S. patents. He has graduated 40 Ph.D. students/post-docs who are holding key academic and industry positions. His research interests include electronic design automation, design for manufacturing, machine learning, hardware acceleration, design/CAD for analog/mixed-signal design, and emerging technologies.

Yongpan Liu (Senior Member, IEEE) received the B.S., M.S., and Ph.D. degrees from Tsinghua University, Beijing, China, in 1999, 2002, and 2007, respectively.

He is currently a tenured Full Professor with the Department of Electronic Engineering, Tsinghua University.

Dr. Liu is a Program Committee Member of ISSCC, ASSCC, and Design Automation Conference (DAC). He received the Under 40 Young Innovators Award DAC 2017, the Best Paper/Poster Award from ASP-DAC 2021 and 2017, the Micro Top Pick 2016, and HPCA 2015, and the Design Contest Awards of ISLPED in 2012, 2013, and 2019. He has served as the General Secretary for ASP-DAC 2021 and the Technical Program Chair for NVMSA 2019. He was an Associate Editor of the IEEE TRANSACTIONS ON COMPUTER-AIDED DESIGN OF INTEGRATED CIRCUITS AND SYSTEMS, IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS—II: EXPRESS BRIEFS, and *IET Cyber-Physical Systems: Theory & Applications*. He has served as the A-SSCC2020 Tutorial Speaker and the IEEE CASS Distinguished Lecturer in 2021.



Nan Sun (Senior Member, IEEE) received the B.S. degree from Tsinghua University, Beijing, China, in 2006, and the Ph.D. degree from Harvard University, Cambridge, MA, USA, in 2010.

He was an Assistant and then a tenured Associate Professor with The University of Texas at Austin. He has been a Professor with Tsinghua University since 2020. He has graduated 26 Ph.D. students, ten of whom are professors at top universities in the USA and China.

Dr. Sun received the NSF Career Award in 2013 and the IEEE SSCS New Frontier Award in 2020. He has published more than 30 IEEE JOURNAL OF SOLID-STATE CIRCUITS articles and more than 50 ISSCC/VLSI/CICC/ESSCIRC papers. He serves on the TPC of ISSCC, CICC, and ASSCC. He was an Associate Editor of IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS—I: REGULAR PAPERS and a Guest Editor of IEEE JOURNAL OF SOLID-STATE CIRCUITS. He serves as a Distinguished Lecturer for both IEEE Circuits-and-Systems Society and IEEE Solid-State Circuits Society.