Exploiting Wireless Technology for Energy-Efficient Accelerators with Multiple Dataflows and Precision

Siqin Liu, *Student Member, IEEE,* Talha Furkan Canan, *Student Member, IEEE*, Harshavardhan Chenji *Member, IEEE*, Soumyasanta Laha *Member, IEEE*, Savas Kaya, *Senior Member, IEEE*, and Avinash Karanth *Senior Member, IEEE*

***Abstract*—As model size and the number of layers increase, Deep Neural Networks (DNNs) demand enormous computational power and throughput to meet exceedingly high prediction accuracy's of today's machine learning (ML) applications. Spatial hardware accelerators have been proposed that optimize the dataflow and exploit sparsity to provide a significant decrease in power consumption. As spatial architectures are traditionally designed with metallic interconnects, significant power is expended for data movement for different dataflows. In this paper, we exploit extended wireless technology to design a power-efficient and high-throughput DNN accelerator, e-WiNN, that can be configured for all representative dataflows and arithmetic precisions. We leverage novel circuit design by utilizing Dadda-algorithm based Multiply-and-Accumulate (MAC) circuits for 4-bit, 8-bit and 16-bit inputs to reduce area, power and delay constraints in 14 nm predictive technology. Our novel wireless transmitter integrates on-off keying (OOK) modulator with power amplifier that results in significant energy savings. To reduce the area overhead, we cluster wireless transceivers into groups of four such that both weights and input features can be effectively multicast to reduce the data movement. The energy efficient transceiver circuit is implemented in state-of-the-art BSIM 32 nm FinFET technology model and our link budget considers required RF power for different frequencies and inter-PE distance at three different antenna directivities including isotropic. Our detailed RTL modeling and cycle-accurate simulation results show that e-WiNN achieves 36.3% latency reduction and 76.1% energy saving when compared to state-of-art wire interconnected accelerators; 70.3% area reduction and 41.6% energy saving at the cost of 11% latency increase when compared to prior wireless accelerators on various neural networks (AlexNet, VGG16, and ResNet-9/50).**

***Index Terms*—Wireless technology, transceiver design, dataflows, precision.**

I. Introduction

With the ever increasing computational demands of emerging applications, the use of Deep Neural Network (DNN) based hardware accelerators [1], [2], [3] has significantly increased in recent years. Hardware accelerators are specialized computational units that are tailored to a particular application; they trade off plasticity, programmability and configurability for improved energy efficiency and throughput. Examples of hardware platforms used for acceleration include graphics processing units (GPUs), field programmable gate arrays (FPGAs) and application specific integrated chips (ASICs) [4], [5]. In this article, we focus on designing ASIC-based hardware accelerator due to improved energy-efficiency and configurability.

DNNs have demonstrated superhuman accuracy for a myriad of applications such as image processing, speech recognition, autonomous systems and edge computing. They have become larger and deeper, imposing stringent requirements to efficiently move vast amounts of data for parallel computation. ASIC-based spatial accelerators have been proposed to achieve high parallelism with arrays of Processing Elements (PEs) and energy efficient data movement using Network-on-Chip (NoC) architectures. However, the increasingly large DNN models impose high bandwidth and low latency communication demands between PEs, which is a fundamental challenge for metallic NoC architectures.

Researchers have proposed other emerging technologies such as wireless and photonics for replacing or augmenting metallic interconnects due to improved performance/Watt, bandwidth-density and reconfigurablity advantages over traditional electrical links. Wireless technology can significantly alleviate the cost as wireless technology offers several degrees of freedom compared to electrical - broadcast/multicast data, single hop communication across longer distances and frequency division multiplexing (FDM) where multiple frequencies/channels can be used to send data simultaneously. Wireless technology could improve energy-efficient and provide higher performance for hardware accelerators designed for DNN applications where the same data has to be broadcast to multiple PEs simultaneously (weights or input maps). But, it is an open question whether wireless technology can scale to large PE arrays, given the cost of fabricating a wireless transceiver at every PE. Prior work have proposed wireless hubs or clusters to reduce the cost of wireless transceivers and optimize the performance and energy-efficiency [6], [7], [8].

Recent research has enabled DNN layers and models which trade off prediction accuracy for computational complexity [9]. This is achieved through the use of lower arithmetic precision as well as quantization of the inputs and weights in the DNN. It has been shown that 4 bits quantization from original 8 bits fixed bitwidth can reduce the latency and energy consumption by $2\times$ while the accuracy drops by only 4.7% [10]. Depending on the application where resources are constrained (such as autonomous vehicles, wearable devices, and predictive main-

S. Liu, T. F. Canan, H. Chenji, S. Kaya and A. Karanth are with the School of Electrical Engineering and Computer Science, Ohio University, Athens, OH 45701 USA e-mail: ls847719@ohio.edu, tc675716@ohio.edu, chenji@ohio.edu, kaya@ohio.edu and karanth@ohio.edu

S. Laha is with the Department of Electrical and Computer Engineering, California State University, Fresno, CA 93740 USA e-mail: laha@csufresno.edu

Manuscript received September 30, 2021.

tenance), it could be acceptable to operate the DNN at slightly lower accuracy while achieving significant reduction in energy consumption and has been done so in many practical scenarios. Analyzing the precision-quantization for hardware accelerators with wireless interconnects has not been unexplored.

In prior work, wireless interconnect based neural network (WiNN) has been proposed to reduce the latency by $3\times$ and the energy consumption by $1.5\times$ [11]. However, there are several shortcomings of the prior work. WiNN was designed with a fixed dataflow called Wireless-for-Multicast (MW) that enabled multicast using wireless technology. However, such a fixed dataflow could be sub-optimal for different DNN layer types. For example, certain convolutional layers show better performance for different dataflows such as row-stationary or weight stationary which was not explored with WiNN architecture. WiNN was designed with wireless transceiver assigned to each PE. While this allowed highest flexibility for transmitting and receiving wireless data, wireless transceivers consumed substantial power (15.5%) and area overhead (43.5%). Finally, WiNN assumed fixed precision (32 bits) and did not consider the accuracy-performance design trade-off with varying precision.

In this article, we propose **e-WiNN** (extended WiNN) by exploiting wireless technology to design a power-efficient and high-throughput DNN accelerator that can be configured for different dataflows and different arithmetic precisions at runtime. We leverage novel circuit design by utilizing Dadda-algorithm based Multiply-and-Accumulate (MAC) circuits for 4-bit, 8-bit and 16-bit inputs to reduce area, power and delay constraints in 14 nm predictive technology. Our novel wireless transmitter integrates on-off keying (OOK) modulator with power amplifier that results in significant energy savings. To reduce the area overhead, we cluster wireless transceivers into groups of four such that both weights and input features can be effectively multicast to reduce the data movement. The energy efficient transceiver circuit is implemented in state-of-the-art BSIM 32 nm FinFET technology model and our link budget considers required RF power for different frequencies and inter-PE distance at three different antenna directivities including isotropic. The transceiver power dissipation is 25 mW. The main contributions of this article are as follows:

- **Clustered PEs and Multiple Dataflows:** We propose a wireless interconnected neural network accelerator by clustering PEs. By employing clustering and wireless transceiver sharing, we reduce the overhead of the wireless interconnection by 2.1x compared with the prior work (WiNN) with the same throughput and power constraints.
- **Precision and Transceiver Circuits:** To reduce area, power and delay overhead, Dadda-algorithm based MAC circuits with 4-bit, 8-bit and 16-bit inputs are designed and simulated with 14 nm technology libraries. Combining with power-efficient OOK modulator, we further improve energy-efficiency.
- **Performance Evaluation:** Our detailed RTL modeling and cycle-accurate simulation results show that e-WiNN achieves 36.3% latency reduction and 76.1% energy saving when compared to state-of-art wire interconnected accelerators; 70.3% area reduction and 41.6% energy saving at the cost of 11% latency increase when compared to prior wireless accelerators on various neural networks (AlexNet, VGG16, and ResNet-9/50).

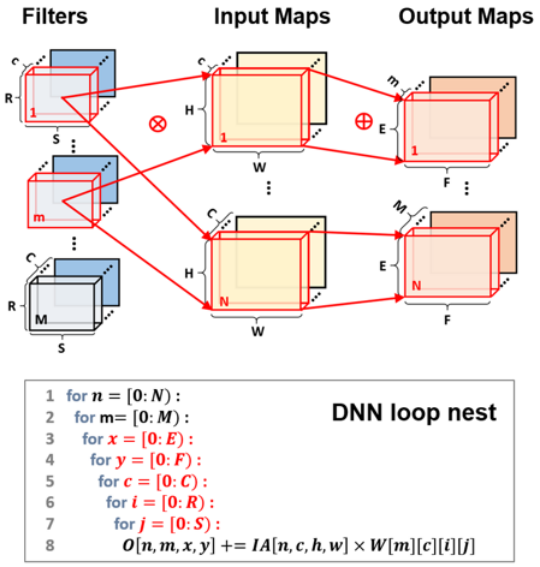

Figure 1: Convolutional operation in DNN. Input feature maps are convolved with distinct filters in each channel (top). The results of the convolution at each point are accumulated across all channels to obtain the output feature maps. Complete convolution operation in CNN consists of a 7D nested for loop (bottom).

II. Background & Motivation

In this section, we briefly explain the background on DNN computation for different applications. We also discuss various dataflows used in hardware accelerators.

A. DNN primer

Deep Neural network (DNN) is an artificial neural network (ANN) with multiple hidden layers between the input and output layers that can be trained to model the behavior of complex non-linear functions. Convolutional Neural Networks (CNNs) are a class of DNNs that are widely used for image processing. Although our approach can be applied to various DNN architectures such as fully-connected (FC), recurrent neural network (RNN), and long short-term memory (LSTM), we focus on 2D convolution in this paper because the computation of the convolutional layer dominates the complexity and energy consumption [12]. Convolutional layers convolve the input in the form of a raw image or an input activation map (the output of a previous convolution layer) with a filter to produce an output feature map as shown in Fig. 1. S and R are the width and height of the filter volume; W and H are the width and height of the input map volume; F and E are the width and height of output map volume respectively. C is the channel for both weight and input map, M is the number of filter volumes, and U is a the given stride size. Complete

convolution operation in CNN consists of a 7D nested for loop as shown at the bottom of Fig. 1.

Tensors in DNNs consist of seven dimensions arranged in a complex manner. For example, the row/column indices of output can be inferred by input and filter indices (i.e., $E = H - R + 1, F = W - S + 1$). The input channel index c appears in both filter and input activation. The output channel m corresponds to the indices of filter volumes. The output volume n indicates the batch index of the input. Because of these specific data access patterns, various transformations while executing the computation can be performed by keeping one of the data structure stationary (i.e. unchanged in a PE), which can significantly reduce data movement between global/local buffers in DNN accelerators.

B. Accelerators and Network-on-Chips

DNN accelerators are specialized architectures for running DNN workloads in a highly parallel and energy-efficient manner. A sea of PEs are tightly interconnected to achieve the inherent parallelism in DNN applications. Each PE consists of light-weight multiply-and-accumulate (MAC) units, register files, and controllers, as opposed to conventional monolithic arithmetic logic unit (ALU) with deep pipeline stages. Most DNN accelerators include a shared SRAM global buffer (L2) to store data for the large PE array to avoid frequently accessing off-chip memories. Data delivery is elaborately orchestrated in DNN accelerators, because the PE operation is dependant on data arrival, and the PE stalls if any one of two operands is unavailable due to memory or network delays.

It is ubiquitous in DNN accelerators to use Network on Chip (Chip) to efficiently orchestrate data movement within the large PE array. Spatial accelerators operate in a dataflow manner, making the NoC a critical component to establish sufficient communication throughput [13]. Prior works rely on specialized buses [14], 2D mesh NoCs [15], and hierarchical meshes [16]. Furthermore, light-weight switches are developed to achieve energy efficiency [17]. Tree-based configurable interconnection fabrics are proposed to enable flexible mapping of multiple dataflows [18]. However, as modern DNN models is getting larger and deeper, it imposes a fundamental challenge for metallic NoC architecture to adapt to the high bandwidth and low latency communication demands between PEs. Emerging technology, for instance wireless interconnections, has been applied to meet the challenge. The inherent multicast capability, single hop, and distance independent on-chip communication is revealed to be superior in realizing efficient dataflows in DNN workloads [7], [8]. The drawback of integrating wireless interconnections in accelerators is the significant power and area overhead of transceiver circuits and antennas.

C. Data Reuse and Dataflow

Despite the various execution order of the DNN nested loop and partition algorithms, prior work has summarized four behaviors of DNN accelerators reusing data over time and space - spatial/temporal multicasting (input/filter tensors) and spatial/temporal reduction (output tensors) [19]. Spatial/temporal multicasting fetches a data pixel from global buffer (GB) once, reuses the data element by either spatially multicasting to several PEs or uses the data element multiple times within the same PE to reduce remote buffer access which saves energy and reduces latency. Spatial/temporal reduction reduces the accumulated partial sums (Psums) over either multiple PEs or multiple time windows. For example, an adder tree could be used for spatial reduction. Most DNN accelerators endeavor to maximize the opportunities of data reuse in these four patterns to achieve energy efficiency. We further extend the observation by identifying another data reuse pattern (inter-PE propagation), which contributes significantly to energy consumption. **Inter-PE Propagation** transmits a data pixel (input/filter) to the neighboring PE multiple times until exhausted. The data pixel is subsequently reused among the PE array both spatially and temporally to reduce the expensive remote global buffer accesses (energy cost of memory access between GB and PE is $6\times$ more than that of inter-PE communication [14]. These data reuse opportunities make the on-chip communication design critical for data movement to achieve high throughput and energy efficiency.

Conventional dataflow taxonomy is built on how the accelerator takes advantage of the temporal data reuse [1]. For instance, a weight being reused in the same PE for multiple time window instances is called weight stationary (WS), which takes advantage of temporal multicasting. Weights can be multicast to multiple PEs to further exploit spatial multicasting. Output stationary (OS) dataflow is a typical case of exploiting temporal reduction of Psums. Row stationary (RS) dataflow exploits both temporal and spatial multicasting of input and filter tensors along horizontal and diagonal PEs respectively. Spatial and temporal reduction of Psums is also explored in RS. However, RS supports all the four data reuse behaviours at the cost of reduced number of spatial multicast receivers. Multicast-for-wireless (MW) dataflow has been proposed to maximize the spatial multicast opportunities in the data reuse patterns such that the multicast capabilities of wireless interconnections are efficiently explored to reduce the communication overhead [11].

In this paper, we specifically address the data reuse behaviour associated with various dataflows by using energy-efficient wireless transceivers and new NoC topologies to achieve high throughput and energy efficiency with reduced area overhead for DNN applications.

III. Proposed e-WiNN Architecture

A. Micro-architecture

The proposed e-WiNN architecture is depicted in Fig. 2, consisting of the on-chip global SRAM buffer (GB), data dispatcher (DP), spatial 2D PE array and centralized accelerator controller (AC). The PE array is constructed by clustering with a concentration factor of N ($N = n^2 = 4$). The goal with clustering PEs is to limit the number of wireless transceivers per cluster and to scale the system effectively. Increasing the number of PEs beyond 4 will increase the energy and latency for intra-cluster data movement. Clusters are interconnected by wired links in a $m \times m$ mesh topology. The total number

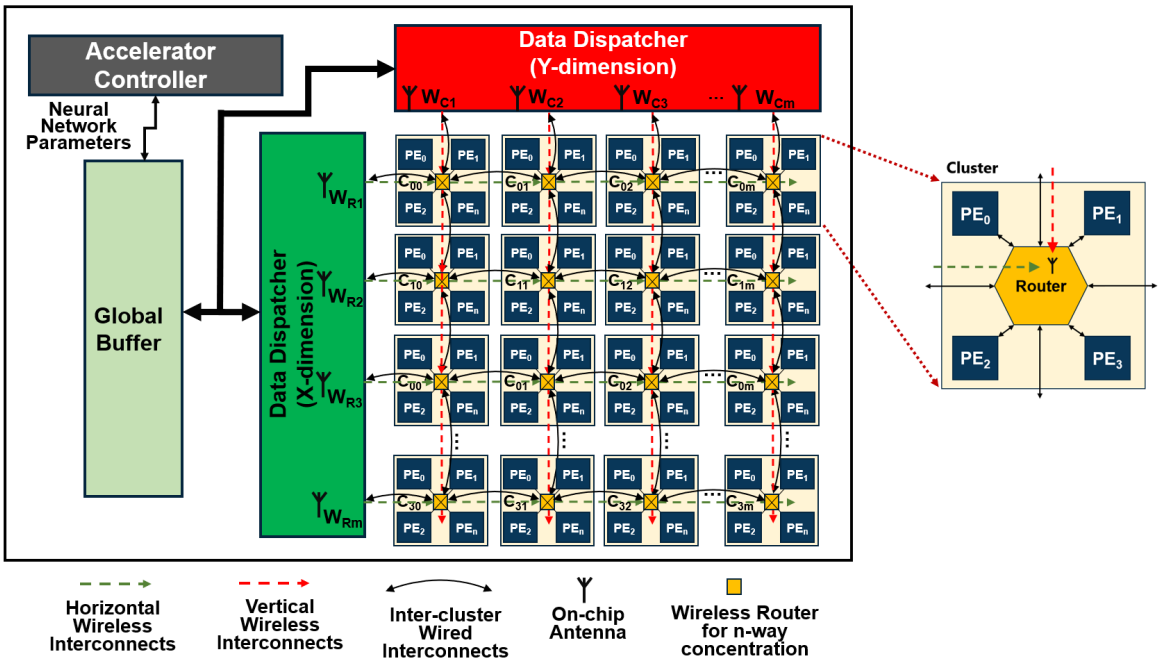

Figure 2: An overview of proposed e-WiNN accelerator. The two-dimensional clustered PE array is interconnected by wires in a mesh topology and by global X-Y dimension order wireless interconnects through wireless routers (yellow box).

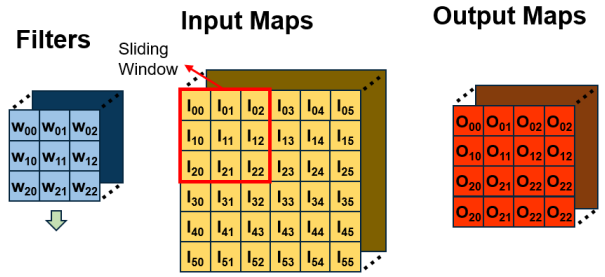

Figure 3: The pedagogical example with a 3×3 filter, 6×6 input, and 4×4 output.

of PEs can be computed as m^2n^2. Each cluster connects to neighboring cluster routers using full duplex wired links in each direction (N/S/W/E). In addition, each cluster has a wireless receiver, that is used to unidirectionally receive data from the global X and Y wireless transmitters in the data dispatcher. The cluster microarchitecture is shown in Fig. 4. The two front-end filters work at band-pass frequencies (circuits in green receives at 70GHz and circuits in red receives at 60GHz) to separately demodulate weights and input activations. The centralized wireless router connects all the local PEs through wired links. Weights and input activations can be multicast/ unicast from data dispatcher to all the PE clusters in one hop and to PEs inside the cluster with one additional hop. Controlled DeMUX switches between unicast and multicast modes to further take advantage of multicast inside the cluster. Therefore, Each PE can either receive the input data from the global data dispatcher or from neighboring PEs in the cluster.

The data dispatcher is the key component that orchestrates

Table I: Data reuse and distribution for various dataflows

Dataflows		MW	WS	OS	RS
Data reuse	Spatial reduction of outputs	×	×	✓	✓
	Temopral reduction of outputs	✓	×	×	✓
	Spatial multicast of weights	×	✓	×	✓
	Temopral multicast of weights	✓	✓	✓	✓
	Spatial multicast of inputs	×	×	×	✓
	Temopral multicast of inputs	×	×	×	✓
	Inter-PE propagate of inputs	✓	×	✓	×

the data delivery and implements the dataflow by selecting the appropriate weights/input activations that needs to be sent to the corresponding PE clusters. We support the flexibility of various dataflows at compilation time. That is, the accelerator controller would search for the most suitable dataflow based on the current neural network workload parameters through the preloaded look-up-table and reconfigure the data dispatcher. Inside the data dispatcher, multiple wireless transceivers and antennas are integrated to build up directional wireless channels in X and Y dimensions to minimize the interference between neighboring rows/columns. Data dispatcher assigns one transmitter and antenna for each row and column of the PE clusters, which comprises a X-Y dimension-order wireless network.

B. Dataflow Implementation

We briefly explain the dataflows implemented on e-WiNN using a pedagogical example in Fig. 3, then we elaborate the detail of how the workload is partitioned, mapped, and processed on e-WiNN with varying dataflows in Fig. 5.

Generally, DNN workloads involve with three types of data, i.e., weight filters, input and output activations. These data

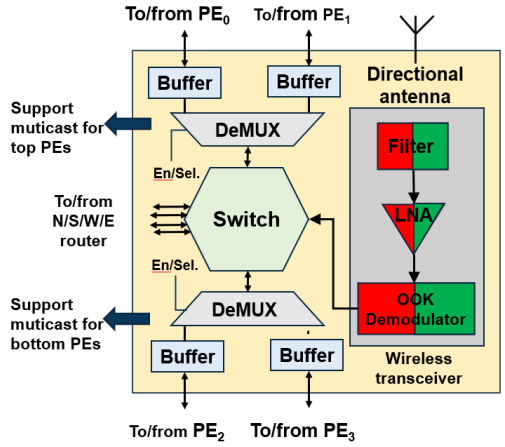

Figure 4: The microachitecture of the wireless router. 9-radix switch supports the data transmission from 4 PEs in the cluster, 4 neighboring switches to the north, south, west, and east, and the wireless transceiver.

are further associated with three traffic patterns, i.e., weights and input activation distribution, and output collection. Output activations may be cached back into global buffer as a form of partial sum (Psum) before exhaustive accumulation due to limited storage in the PE, which result in Psum collection and distribution. We explore these data movement patterns of four representative dataflows, multicast-for-wireless (MW) [11], weight stationary (WS), output stationary (OS), and row stationary (RS) [14]. Each dataflow style has myriad variant forms. We instantiate each dataflow based on two criteria. First, space multicasting can be maximally explored so that wireless interconnection plays a role. Second, the Psum reduction is prioritized inside each PE in scheduling (outer loops). We trade-off the area overheads of storing Psums to allocate more hardware resources for weights and inputs delivery to achieve overall high throughput.

We demonstrate the data delivery in Fig. 5 with an example e-WiNN configuration of 2×2 cluster array using the 2D convolution example in Fig. 3. We start by visualizing the interconnections and data mapping onto e-WiNN in the second row. For example in MW dataflow, horizontal wireless channels (F_{R1} and F_{R2}) multicast weight data to all the clusters in the row, while vertical wireless channels (F_{C1} and F_{C2}) multicast input maps to all the clusters in the column. The data dispatcher schedules the data delivery based on the characteristics of the selected MW dataflow. At $t0$, we initialize the PE array by multicasting W_{00} to all PE clusters, while unicast the first 4×4 patch (sliding window) of input map. At the next cycle, a new weight pixel is fetched and multicast, and a row/column of input map sliding window is multicast to the row/ column of clusters. The remaining input feature data of the sliding window propagates to the neighbouring PE cluster by the inter-cluster wired links. Detailed indices of weights and input maps for each cycle are listed in the table at the last row, as well as the corresponding interconnections.

The second and third rows in Fig. 5 provide an insight of the data reuse for all the three data structures. X axis denotes the indices of output data. Four horizontal black dots are spatially mapped to four clusters respectively. At $t1$, another partial sum is computed for the same output data in the cluster. Therefore, Psum data movement is avoided and no local register file is required. Y axis denotes the indices of weights. Weights are multicast to all clusters in the row at each time window, following the spatial multicast pattern. Input feature maps are multicast to clusters in the column, except for $t0$, because there is no data reuse opportunities at the initialization stage. After the initialization, most of the input feature maps are reused by the neighboring clusters through inter-cluster wired links to reduce global buffer accesses. As concluded in Table I, the data reuse in MW dataflow consists of temporal reduction of outputs, spatial multicast of weights, and inter-cluster propagation of input maps.

In weight stationary dataflow, weights are reused both spatially and temporally. Only one global transmitter is required to broadcast the weights. We constraint the wireless beam width within the 5 dBi antenna directivity as discussed in the transceiver design section. As weights are maximally reused in WS, no input and output maps data reuse can be explored. The wireless interconnection topology in output stationary dataflow is identical to that in MW, as well as the input feature map delivery. However, partial sums of a output data are spatially mapped to the cluster array in OS. Inter-cluster Psum reduction is involved to obtain the output. The advantage of OS compared to MW is the temporal reuse of weights.

The angle of antennas in the column data dispatcher is adjusted for row stationary dataflow. Input feature maps are distributed diagonally and reused by the PE clusters along the direction. Row wireless channels multicast the weights to the clusters in the row, while Psums accumulate vertically through wired links. RS is proved to be superior to other wired link based dataflows [14] due to intensive utilization of data reuse for all the three data structures. Nevertheless, it comes at the cost of reduced multicast opportunities in weights and input feature maps (For instance, in Fig. 5, input feature maps can not be multicast to corner clusters), which downgrades the superiority when evaluated on e-WiNN. We will further analyze characteristics of these dataflows and the impact on overall latency and energy performance in the simulation section.

C. Reconfigurable MAC Circuits

A major computational bottleneck in neural network accelerators is MAC operation, especially when building matrix and vector elements [20]. Since MAC circuits are also an important part of DNN architecture, their performance plays a crucial role on the system level assessment of the e-WiNN accelerator proposed here.

1) N-Bit MAC Design Based on Dadda Multiplier: To reduce area, power and delay overhead, MAC circuits with 4-bit, 8-bit and 16-bit inputs are designed using standard Dadda-algorithm and simulated with 14 nm technology libraries. Each MAC is designed with unsigned bit representation to reduce hardware complexity. For an n-bit input MAC, there is a 2n bit result array at the end of Dadda multiplier (see Fig.6). It

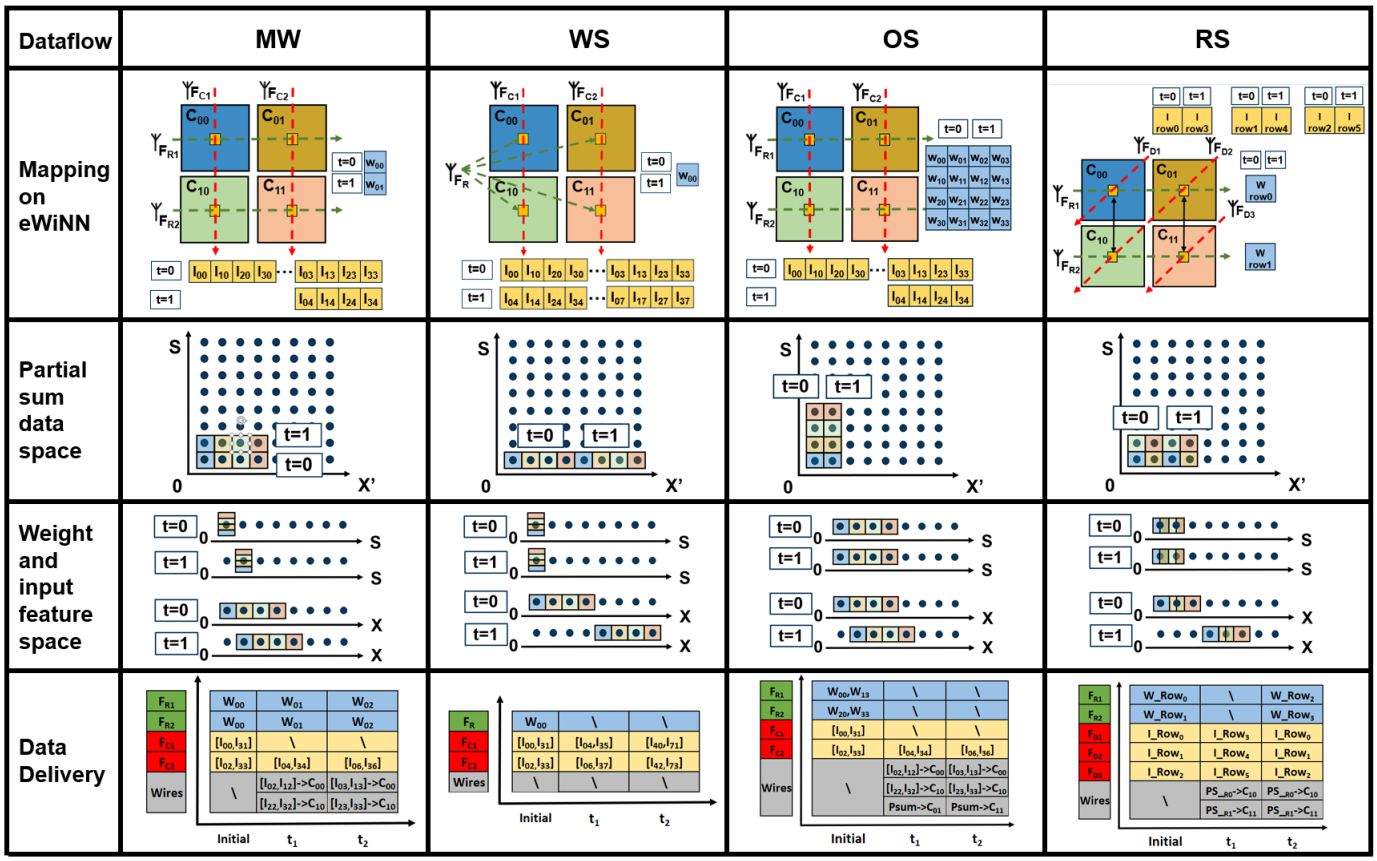

Figure 5: Data delivery, mapping, and partitioning presented over 2D convolution presented in Fig. 3. The first row shows four representative dataflows evaluated on e-WiNN. The second row shows the physical mapping and execution timing of partial sum. Each black circle represents four partial sums, evenly assigned to each PE inside the cluster. The third row shows the mapping for weight and input feature data. The last row tabulates the timing of each wireless channel and wired link for transmitting all data structures over the first three time window instances.

is based on the simplification of each partial product column as much as possible to a given column height at each stage and use the number of half and full adders wisely to keep power consumption and delay minimal. As compared to the Wallace-tree approach, Dadda multipliers require fewer gates to achieve the final summation products, so that the resulting partial product reduction circuitry (PPRC) needs less area and power while operating faster. [21]

Two N-bit wide inputs to the MAC circuit are first fed into the partial product generation circuit (PPGC) at which the N^2 partial product terms are created as a result of partial product generation process. In the next step, all partial product terms are used as inputs for PPRC. When partial product reduction finishes, two partial product reduction sum (PPRS) arrays become available at the output of PPRC. Then, the 2-bit final addition is performed in the adder circuit, as shown in Fig.6. Afterwards, the sum is directed to accumulator. Here, there are three different options designed to control the accumulation operation: synchronous load flag to clear all input registers, the data flag bit to check the availability of PPRM register's output and a reset bit (asynchronous clear) to clear the inputs. The PPRS array controlled with separate flags: whenever a CLK signal is enabled or the data flag bit is "Logic 1". The result is sent to an adder block where the 2-bit sum is calculated. If synchronous load flag bit is true, the data in the accumulator is deleted. Otherwise, at each clock rising edge input data are written to registers, synchronous load control bit is updated and the final addition operation is performed. If an asynchronous clear signal is applied the module is reset and all the input registers except the data in the memory register/accumulator gets deleted. Separating, the synchronous load and asynchronous clear flags to control different parts of registers provide system designer to flexibility for input and output data flow control. This design serves as a baseline for comparison with the novel precision-controlled MAC unit below. figure/

2) 16-Bit Variable Precision MAC Unit: Next, we introduce a novel variable-precision MAC circuit with 16-bit wide inputs which is capable of giving an output of 2N+1, where N is 4, 8 or 16 bits, depending on the input control signal. A similar approach has been recently suggested [22] that also considers floating-point representation of weights. In our new design, the redundancy associated with having multiple (parallel) 4, 8 or 16-bit blocks dedicated for different precision levels is eliminated. Considering all of the input bits are fully utilized (i.e., both inputs are 16 bits), first, they are applied to the

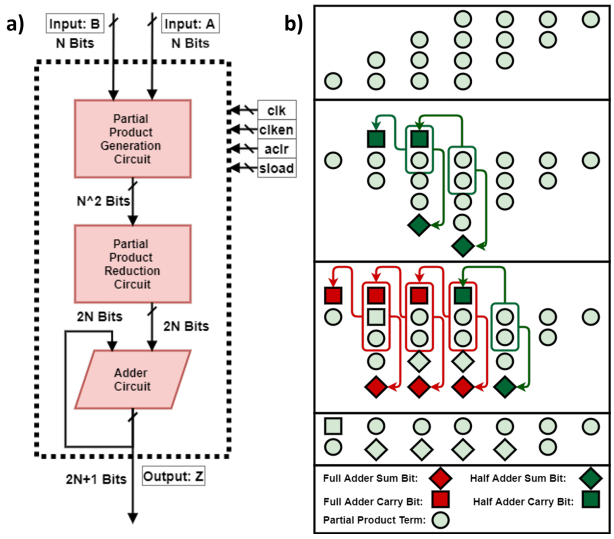

Figure 6: a) The block diagram of standard N-Bit Dadda-Multiplier based MAC circuit b) 4-Bit Dadda reduction performed in PPRC block

PPRC which gives a 16×16 bits wide output, containing all the generated partial product terms. This part is common to all precision levels. Next, depending of the resolution needed, the generated partial product terms are routed into PPRC blocks, selectively. This decreases the number of PPGC units that would have been repeated in parallel implementations for different resolutions. Following the PPRC process, the precision control bit determines which 2N-bit multiplication data is sent to the adder block for the final accumulation. The adder block in the re-designed VP-MAC block is replaced with a 32-bit Carry Lookahead adder (CLA) which is selected to minimize delays regarding the carry calculation [23]. This helps to avoid a bottleneck caused by rising-edge-enabled clock during accumulation operation. As a result, VP-MAC circuit should perform the same 4/8/16-bit operations with lesser area, power dissipation, delay, while also accounting for the overflow condition.

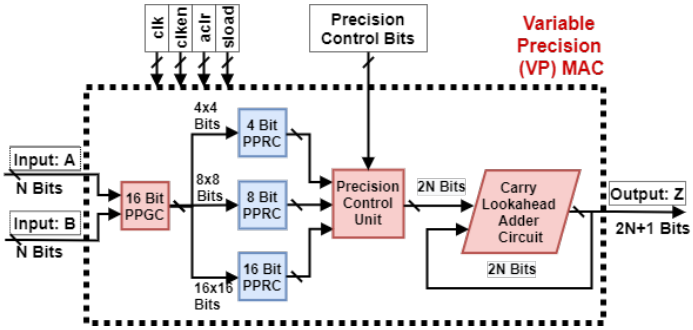

Figure 7: The block diagram of variable-precision (VP) MAC circuit with re-utilized partial products and carry-look-ahead accumulator.

The encoding and decoding of TRX are both changed to be consistent with the varying precisions. The bitwidth of each data is a preloaded hyperparameter to the TRX to correctly match the precision of the MAC. As the flexibility of varying precision is only implemented offline, no extra reconfiguring hardware is consumed in the TRX. The serializer and modulator of TRX remains intact for varying precisions.

D. Wireless TRx

In MW, the transmission along the rows and columns takes place simultaneously using two adjacent but different frequency channels. The use of the directional antennas alleviates the design of multiple transceiver in different frequency bands, thus avoiding design complexity and migration to power hungry BiCMOS or III-V technology if we were to scale up in frequency. Thanks to recent advances [24], [25], [26] especially in additive manufacturing and 3D chip integration, such highly directive antennas are easier to pursue either by in-plane dielectric engineering, structural guiding via etching, bonding or packaging elements. While certainly non trivial to build and test, use of directive antennas can ensure higher flexibility for dataflow in the proposed WiNN accelerator, as evident in the link budget analysis (Fig. 8).

As an illustration of the concept and indicative of the potential of e-WiNN, the current design proposes to use 60 GHz and 70 GHz as the two frequency bands for transmission for the weights and inputs, respectively. A quarter wave monopole antenna with a very high directivity (around 5 dBi) has been considered for the MW. Such high-directivity requires either the use of metasurfaces or superstructures over the chip surface [25], [26] or loaded dielectrics [24] along with a quarter-wave monopole antenna. The spacing of the adjacent antennas is more than $\frac{2D^2}{\lambda}$ to ensure the antennas do not lie in the near field region of each other. The maximum linear dimension (D) of the antenna is assumed to be 2 mm. With λ = 5 mm for 60 GHz, the spacing between adjacent antennas is kept at a minimum distance of $\sim$ 4 mm to avoid any near field effect of each other as depicted in Fig. 9.

1) Link Budget: A link budget is evaluated for the wireless communication of the transmitter data following OOK modulation considering multiple design environments at a BER of 10^{-12}. As can be seen from Fig. 8, the required transmit power decreases significantly with increasing antenna directivity for both the inter-PE distance and frequency. We assume the maximum number of PEs in e-WiNN is 4096 to meet the stringent area and power requirements in most edge applications. Based on a fabricated and tested prototype of the 4096-core multi-chip-module architecture, we estimate the maximum inter-PE distance as 48 mm [27]. A floorplan estimation is depicted in Fig. 9. The simulated transmit power of the power amplifier (PA) in our work is -1.5 dBm. As can be observed from Fig. 8 that data even at 16 Gbps can be transmitted to a distance of 48 mm at 60 & 70 GHz frequencies even for an isotropic antenna to support 4096 core architecture as we scale up in the future. This is because the required transmit power is -2.83 dBm and -1.5 dBm at 60 GHz and 70 GHz respectively as can be observed from Fig. 8. It is important to note that the data rate used in this work is 8 Gbps as listed in Table IV. To accomplish the same task at the current 8 Gbps, the required transmit power for 60 GHz and 70 GHz

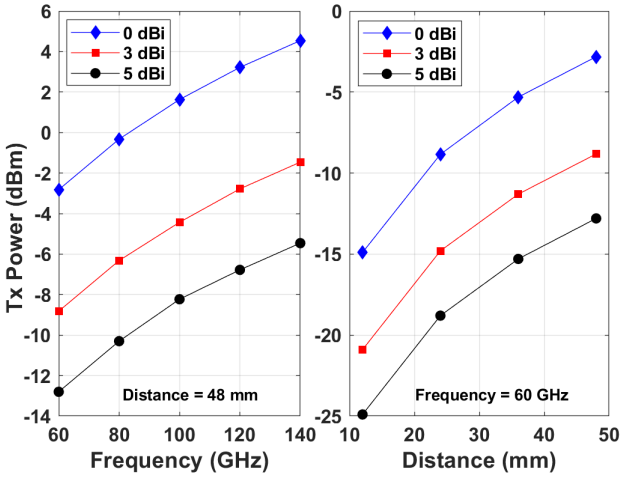

Figure 8: The link budget analysis for 3 different antenna directivity including isotropic: The RF power requirement for different frequencies at the 48 mm inter-PE distance and the RF power requirement for different inter-PE distances at 60 GHz frequency.

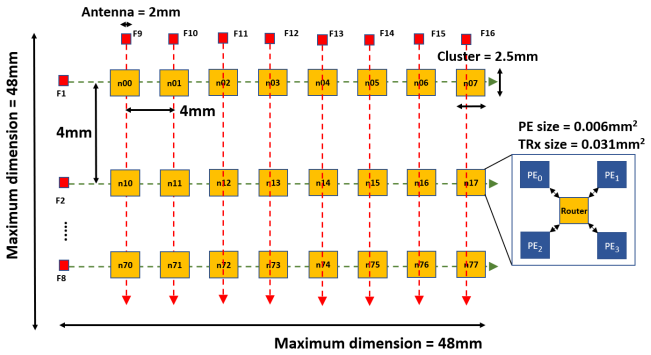

Figure 9: The floorplan estimation with the maximum chiplet dimension 48mm, incorporating 8×8 clusters and 16 global antennas.

center frequencies are much less at -5.84 dBm and -4.5 dBm respectively. As can be seen from the link budget analysis once we migrate to higher frequencies directive antennas can replace isotropic to achieve this. As earlier mentioned, this design takes into account directive antennas to work with two frequency channels without time multiplexing with an assumed directivity of maximum 5 dBi. The work on link budget in our study is validated as we compare it to a recent work on intra-chip communication [28] that is supported by simulation results from channel modelling. The current study approximates the medium to be air/vacuum while making the loss computations. This approximation is based on the fact that the actual channel will be three-dimensional with the quarter wave monopole antenna.

2) OOK Transceiver: The wireless communication is achieved using the incoherent Amplitude Shift Keying (ASK) based On-Off Keying (OOK) modulation. The modulation scheme uses an integrated modulator and power amplifier in the transmitter and an envelope detector at the receiver for the demodulation as depicted in Fig. 10a and 11a in 32 nm FinFET technology model. This is an upgradation in technology model for power efficiency form our earlier design An earlier design in 45 nm technology of the transceiver [11]. The proposed CGM-FinFET transmitter consists of a differential LC oscillator and a two-stage cascade common-source PA, stage one of which also acts as the OOK modulator. The data is fed into the driver transistor M4 when there is logic '1' that enables the switch M5. The single transistor switch thus acts as the modulator of such an OOK transmitter. The OOK receiver uses an energy efficient Low Noise Amplifier (LNA) and an envelope detector to demodulate the OOK modulated signal. An energy and area efficient active Dickson Rectifier [29] has been used for the envelope detection of the OOK modulated signal.

3) Transceiver Performance: The OOK modulation ensures the power efficiency of the transceiver by avoiding the design of phase shift keying (PSK) modulators and phase locked loops (PLL). As explained earlier in the Link Budget, the output power of the PA is -1.5 dBm ensures the signal to transmit to a maximum distance of 48 mm at 60 GHz to support a migration to a 4096 core architecture supporting scalability of the design. Since the separation between the transmitter and the receiver are in the range of few millimeters, the gain of the amplifier is intentionally kept low at a peak gain of around 5 dB to minimize the power dissipation. As depicted in Fig. 10b, the bandwidth is ∼20 GHz ranging from 60 GHz to 80 GHz at 1.5 dB. The LNA in the receiver achieves a peak gain of 8 dB (Fig. 11b). The observed noise figure and the 1-dB compression point is 6.5 dB and -5.4 dBm respectively. The bandwidth is ∼25 GHz ranging from 60 GHz to 85 GHz at a gain of 2 dB. The Dickson Rectifier demodulates the OOK modulated signal to retrieve the digital data at the receiver (Fig. 12). Both the transmitter and receiver has been designed with the UC Berkeley CGM technology model at 32 nm [30].

IV. Performance Evaluation

To evaluate the performance of e-WiNN, we simulate the proposed architecture e-WiNN over representative DNNs workloads, i.e., ResNet-9/50, VGG16, and AlexNet [31], [32], [33]. ResNet-9 and ResNet-50 enables us to evaluate the impact when there is a difference in the number of layers. VGG16 relies on deep depth-wise convolutional layers, exhibiting a wide range of the number of filter/input channels. AlexNet uses large filter size but shallow depth. We selectively construct the benchmarks by these DNNs to provide a comprehensive examination of e-WiNN on various applications.

The propagation delay, power consumption and area overhead of each electrical component are obtained through SAED14nm EDK and PDK from synpopsys solvnet educational PDK. The transceiver circuit is implemented in the CGM 45 nm FinFET technology model from UC Berkeley [30]. We use a cycle accurate network simulator to obtain the latency and throughput of implementing the DNN benchmarks. By multiplying the active clock cycles of each component

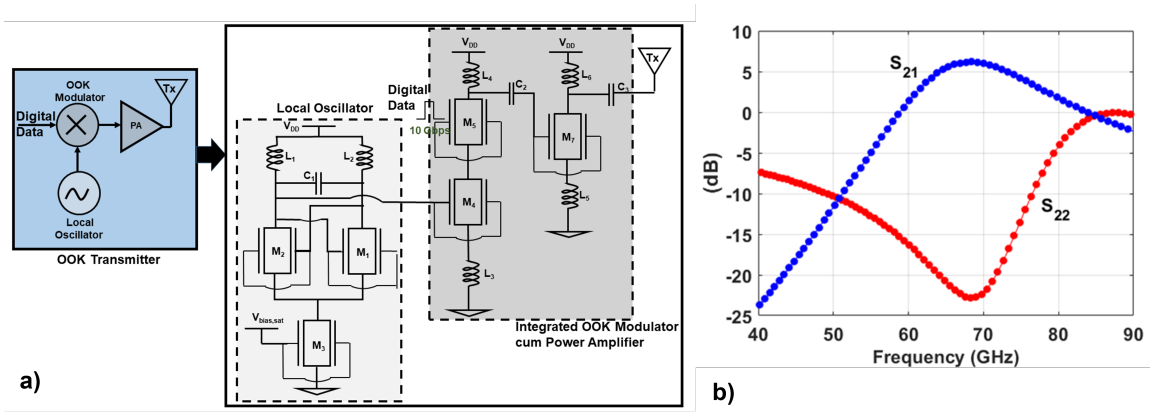

Figure 10: The OOK transmitter circuit: a) Block diagram and the circuit implemented in 32nm FinFET technology. The single transistor switch, M_5, acts as the OOK modulator. This is an upgradation in technology model for power efficiency from our earlier design [11] that uses 45 nm technology. b) The Gain (S_{21}) and Output Reflection Loss (S_{22}) of the PA.

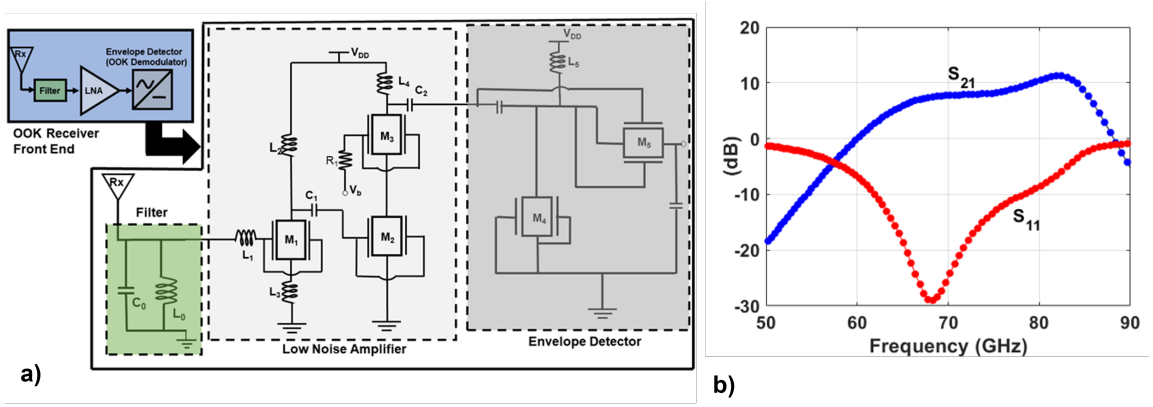

Figure 11: The OOK receiver circuit: a) Block diagram and the circuit implemented in 32nm FinFET technology. This is an upgradation in technology model for power efficiency from our earlier design [11] that uses 45 nm technology. b)The Gain (S_{21}) and Input Reflection Loss (S_{11}) of the LNA.

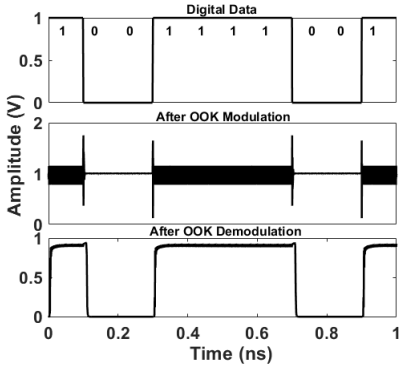

Figure 12: OOK Modulation and Demodulation of the digital data.

with the synthesized power, we obtain the implementation energy cost. Wireless transceiver support both unicast and multicast transmission. In multicast/broadcast, energy cost of a single transmitter and a number of receivers are computed. To compute the energy of an wired traffic inside the cluster, we compute the hops multiplied by the per-hop energy.

We compare the performance of e-WiNN, WiNN, and a conventional 2D mesh PE array on ResNet-9 in Fig. 13. The 2D mesh PE array is constructed in a systolic fashion and interconnected by a light-weight router architecture as proposed in [13] with a X-Y dimension routing algorithm. The latency, energy cost, and the energy-delay-product (EDP) are shown in the top, middle, and bottom plot respectively. We implement ResNet-9 on these architectures with all dataflow types as explained in Section III.B, i.e., output stationary (-OS), weight stationary (-WS), row stationary (-RS), and multicast-for-wireless (-MW). Wireless based architectures achieve at least 59.7% latency reduction and 37.7% energy saving when compared to wired 2D mesh architecture on all dataflows. Both WiNN and e-WiNN demonstrate performance improvement (nearly 2x) for latency and energy consumption on MW dataflow. The results validate the efficiency of MW dataflow that maximizes the multicast opportunities in DNN workloads on wireless interconnection based accelerators. In e-WiNN, one additional wired interconnection hop is required when the sliding window moves across clusters in OS dataflow. That is the reason why we observe higher latency on e-WiNN

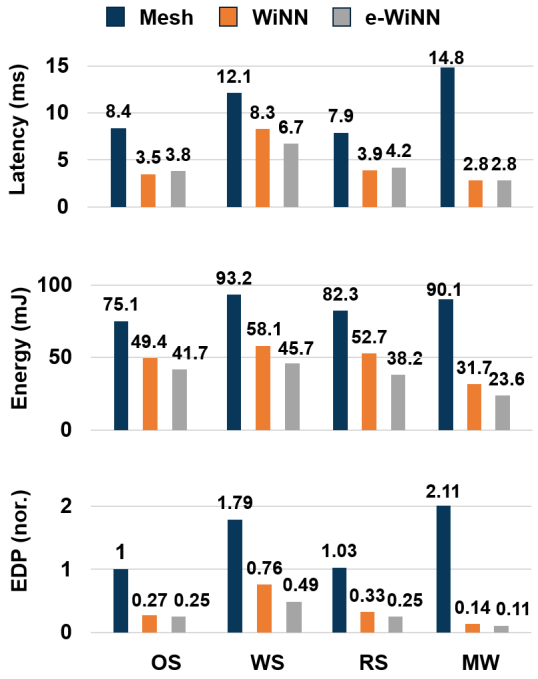

Figure 13: Execution Latency (top), energy cost (middle), and EDP (bottom) of ResNet-9 on various hardware configurations.

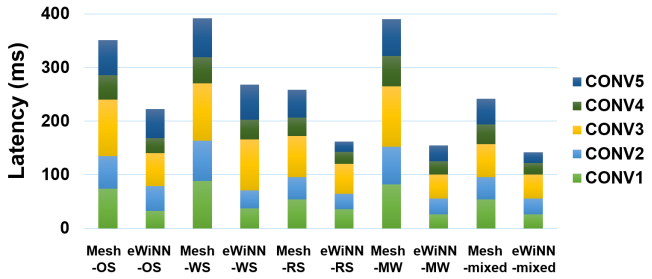

Figure 14: Execution latency of ResNet-50 with output stationary (-OS), weight stationary (-WS), row stationary (-RS), and multicast for wireless (-MW) dataflow on Mesh and e-WiNN architecture. We further demonstrate the flexible dataflow implementation for each layer type (-mixed)

when OS is compared to WiNN. However, MW dataflow shows identical latency on e-WiNN and WiNN, because most data distribution is via wireless channels. We further show the energy delay product (EDP) plot in the bottom, in which e-WiNN performs the best.

Fig. 14 shows the execution latency of ResNet-50 on the 2D Mesh and e-WiNN architecture. Four representative dataflows (OS, WS, RS, and MW) are all examined. Mesh architecture achieves the best latency performance with row stationary dataflow, approximately half of the latency with weight stationary, while e-WiNN performs best with multicast for wireless dataflow, which is specifically developed for wireless interconnections. As argued in Section III.B, although row stationary dataflow achieves the most data reuse, the reduced opportunities to multicast data would deteriorate the

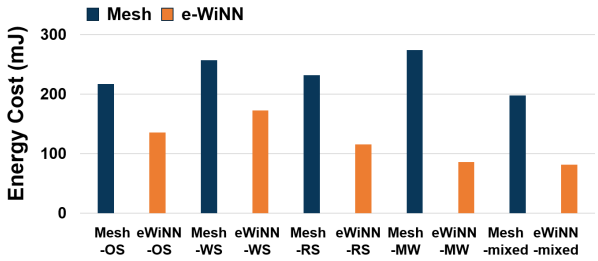

Figure 15: Energy cost of ResNet-50 with dataflows output stationary (-OS), weight stationary (-WS), row stationary (-RS), multicast for wireless (-MW), and mixed dataflows (-mixed) on the 2D Mesh and e-WiNN architectures.

Table II: Performance of the proposed MAC circuits with 14nm Technology ($f_t = 1GHz$).

Constraint	4-Bit MAC	8-Bit MAC	16-Bit MAC	16-Bit VP-MAC
Power (uW)	14.909	45.621	384.26	171.608
Area (um^2)	38.01	112.91	732.29	487.067
td$_{max}$ (ns)	0.03	0.04	0.04	0.04

performance. Another observation is that the parameters of the convolutional layer, for instance, number of channels and filter and input activation ratio, have a significant impact on the efficiency of dataflow. As shown in Fig. 14, MW is the optimal dataflow for the e-WiNN over the entire ResNet-50 workload. However, CONV2 layer type has the lowest execution latency with RS dataflow. Therefore, we re-implement the simulation on a layer-wise basis, where most suitable dataflow for each layer type is pre-configured. The mixed dataflow implementation further reduces 11.3% of the overall latency compared to MW dataflow on e-WiNN.

Fig. 15 shows the energy cost of ResNet-50 with various dataflows (OS,WS,RS, and MW) on the 2D Mesh and e-WiNN architectures. The 2D Mesh architecture achieves the highest energy efficiency with RS dataflow, while e-WiNN prefers MW dataflow. We can infer from Fig. 14 and Fig. 15, the impact of dataflow on latency and energy cost have a positive correlation. The layer-wise flexible dataflow implementation further reduces the energy cost by 13.8% and 19.7% respectively on the Mesh and e-WiNN architectures compared to its globally optimal dataflow. This is static dataflow flexibility only to explore the benefits of dataflow reconfigurability for multi-tenant applications. The runtime reconfiguration controller and the corresponding area and energy overheads are not considered in the simulation.

In order to improve the hardware efficiency, we explore the quantization of both weights and input activations to low precision circuits as shown in Table III. We start with the impact of precision on the accuracy of the DNN models. ResNet-50 achieves 79.26% top1 classification accuracy on ImageNet dataset [31]. Then a simple floating-point-to-integer conversion and retraining generates the 32-bit integer weights with negligible loss in accuracy. We implement ResNet-50 on ImageNet dataset using Pytorch 1.4. Various quantized models are derived from TorchVision domain library.

As for hardware performance, we simulated Dadda-

Table III: Comparison of hardware performance between e-WiNN and WiNN on ResNet-50 over varying precisions.

Precision	Inference Accuracy		WiNN			e-WiNN		
			latency	energy	Area	latency	energy	Area
q(bits)	(top1)	(top5)	(ms)	(mJ)	(mm^2)	(ms)	(mJ)	(mm^2)
32 floating point	79.26	94.75	/	/	/	/	/	/
32	79.24	94.69	92.2	30.1	16.5	162.5.5	30.1	4.9
16	77.13	91.37	83.3	15.9	4.1	115.6	18.6	1.15
16 (VP-MAC)	77.13	91.37	83.3	12.2	2.6	115.6	12.7	0.61
8	75.85	86.24	78.8	9.6	1.7	85.1	8.26	0.46
4	63.96	76.08	76.2	5.0	1.5	79.4	4.1	0.39

algorithm based MAC circuits (Fig. 7) with 4-bit, 8-bit and 16-bit inputs using 14 nm technology libraries as shown in Table II. The simulated power & area figures of the standard MAC units scale non-linearly up to $184\mu W$ and $732\mu m^2$, respectively, as would be expected from increasing complexity. The vimporved VP-MAC unit, which can perform all 4/8/16-bit operations, however, would save substantial (45%) power using only 55% of the area without compromising the speed. The interconnections, on-chip memory, and computation units are all designed to accommodate 32-bit integer operations in the baseline version. We apply the same design to lower precision with reduced duty cycle. Both WiNN and e-WiNN benefit from reduced precision in terms of area overhead, latency, and energy consumption at the cost of accuracy loss. In Table III we observe a convergence of latency reduction at 4bit precision as the bit width scales down. Memory accesses and data movement no longer cause extra delays. Computation operations are on the critical path. The table gives a quantitative insight of the trade-off between neural network accuracy and underlying hardware cost over various quantizations.

We compare the performance of e-WiNN with various state-of-art DNN accelerators in Table IV. Due to the difference of the verification simulators for MAC design and the wireless transceiver, separate semiconductor technology nodes are considered and listed in the second row of Table IV. All the performance metrics are evaluated under the designated semiconductor process. Eyeriss [14] makes a comprehensive study of dataflow in DNNs and implements the proposed the row stationary dataflow in the hardware architecture. It has been widely compared to by follow-up accelerator designs, as hardware architects pay more attention to the data movement. Eyeriss V2 [16] is the second generation of Eyeriss, incorporating a hierarchical mesh NoC to meet the upgrading requirement of throughput and energy efficiency. These two architectures are selected as the representative wired interconnected accelerator with optimized data movement and NoC design. Unlike WiNN, we do not scale these design parameters in the comparison. The original configurations and silicon technology are listed in the table to demonstrate the design choices and the performance of various architectures. e-WiNN achieves 36.3% latency reduction and 76.1% energy saving on the three DNN models compared to Eyeriss V2. Other wireless interconnection based accelerators also demonstrate distinct advantages against Eyeriss series on latency and energy consumption. However, for example WiNN, has a lower performance-per-watt than Eyeriss. It means, although WiNN can beat Eyeriss in absolute value of latency and energy consumption, the relative energy cost of computation operation is still higher. e-WiNN solves this problem by further reducing the energy consumption with clustering at the cost of increased latency. Therefore, e-WiNN performs the best in performance-per-watt except for WIENNA. WiNoC [7] and WIENNA [8] are two accelerators with the wireless hybrid interconnection. WiNoC employs one wireless channel for low latency weight broadcast, following the WS dataflow as shown in Fig. 5. WIENNA is a 2.5D chiplets-based architecture, using wireless interconnections in the interposer. It has orders of magnitude more number of PEs and therefore largest area overheads. Although WIENNA achieves the highest performance-per-watt, it incurs 43.5% more energy cost in average on the three DNN models compared to e-WiNN. As shown in the last two columns, e-WiNN reduces 70.3% area overheads and 41.6% energy consumption, but increases 11% latency increase when compared to WiNN. Furthermore, e-WiNN achieves $2.3\times$ more TOPS in average than WiNN.

V. Related Work

DNN Accelerators and Dataflows. Shidiannao [34] is a fixed dataflow accelerator, which employs output stationary dataflow via mesh-based electrical interconnects. Dadiannao [35] is a tile-based multi-chip accelerator, relying on a fat tree for balanced data transfer between the global buffer and PE. Cambricon-X [36] uses adder tree as the computation engine and exploits sparsity in both weights and input activations. Eyeriss [14] is a state-of-the-art low-energy DNN accelerator that proposed row stationary dataflow and developed the computation-centric dataflow taxonomy. Eyeriss V2 [16] identifies the bottleneck of bus-based NoC as hindering the accelerator performance and proposes a hierarchical mesh to support high throughput and data reuse. Flexflow [37] is a reconfigurable accelerator that supports three distinct dataflows. TPU [38] is a systolic array-based DNN accelerator designed for cloud workloads in data centers. MAERI [18] proposes a chubby-tree for efficient multicast and constructed the PE array with an adder tree to support flexible dataflows. Maelstrom [39] identifies reconfigurable hardware to accommodate multiple DNN workloads with flexible dataflows. All the prior accelerators utilize electrical links for data transfer between the PE and global buffers. However, the inherent high latency and energy consumption of multi-hop traversal by metallic links hinder the performance improvements. Dataflow optimizations allow reducing data movement leading to design trade-offs in energy-efficiency and performance. WAX [40]

Table IV: Hardware configuration and performance comparison of e-WiNN with Eyeriss V2, Eyeriss, WINOC, WIENNA, AND WiNN on AlexNet, VGG16, and ResNet-50.

	Eyeriss v2	**Eyeriss**	**WiNoC**	**WIENNA**	**WiNN**	**e-WiNN**
Technology	65nm	65nm	65nm	65nm	45nm	14nm
Core area (mm^2)	28.1	12.25	14.2	1699	16.5	4.9
PEs	192	168	256	16384	256	256
Throughput (GMACs)	153.6	42	128	7680	113	121
Onchip SRAM (Kb)	246	181.5	64	13312	192	192
Core Frequency (MHz)	200	200	500	500	500	500
Local SRAM	400	512	256	512	128	128
Arithmetic Precision (bits)	8	16	32	\	32	4-16
Wireless Bandwidth (Gbps)	\	\	16	8-16	8	8
Wireless Frequency	\	\	60	60	40-160	40-160
AlexNet: 666M MACs (Conv) + 2.3M (Conv) + 58.6M weights						
Execution Latency (ms)	9.8	30.1	1.7	\	1.5	1.6
Energy (mJ)	5.9	8.4	1.14	1.91-2.35	1.02	0.39
TOPS/W	0.92	1.99	0.83	2.37-3.15	0.82	2.28
VGG16: 15.3G MACs (Conv) + 14.7M (Conv) + 124M weights						
Execution Latency (ms)	215.7	1316	149.3	\	118.3	131.9
Energy (mJ)	85.1	312.5	48.6	39.5-58.3	36.2	23.1
TOPS/W	7.60	12.63	9.22	12.9-16.3	9.80	17.13
ResNet-50: 3.86G MACs (Conv) + 23.5M (Conv) + 2M(FC) weights						
Execution Latency (ms)	239.1	1592	132.4	\	92.2	162.5
Energy (mJ)	72.6	185.3	46.2	31.3-49.0	30.1	15.8
TOPS/W	3.29	8.59	2.87	5.72-6.61	3.06	10.25

is a wire-aware CNN accelerator and employs a deep and distributed memory hierarchy to enable flexible dataflows over short wires. dMazeRunner [41] is a DNN accelerator design search framework, tailoring dataflows for a given DNN model for efficient execution. However, eWiNN evaluates all representative dataflows and explores efficient variants by maximizing multicast capabilities of wireless interconnects.

Wireless interconnections and accelerators. Wireless NoC architectures have been applied to many-core systems to overcome the bandwidth, scalability and power limitations of electrical networks [42], [43], [6], [44], [45], [46]. Several of these prior work were hybrid designs that effectively combined both electrical and wireless technology to improve energy-efficiency and performance. Wireless channels are constructed for long-distance communication as the second tier highway to reduce the latency when traversing multiple hops. Few prior work have incorporated wireless interconnections in DNN accelerators. WiNoC [7] develops a hybrid wireless and wired NoC for the accelerator to efficiently broadcast weights. WIENNA [8] employs wireless interconnects for high bandwidth inter-chiplet communication in the interposer. The inter-chiplet wireless interconnects design is extended by using memory footprint compression to reduce the memory and communication traffic in [47]. A comprehensive study of introducing Wireless Networks-on-Chip as a novel interconnect paradigm to accelerate multicast traffic is conducted in [48]. WiNN [11] develops an X-Y dimension wireless interconnection and a new dataflow to maximize the multicast traffic. This enables simultaneous communication of weights and input activations in x and y dimensions. None of these accelerators consider multiple dataflows to efficiently accommodate various DNN applications through wireless interconnects. e-WiNN further explores PE clustering to reduce the wireless area overheads and flexible precision MAC units to improve the energy efficiency of edge applications.

VI. Conclusions

This work explores the energy-efficiency and performance trade-offs by exploiting emerging wireless technology for implementing scalable DNN accelerators. In conjunction with accommodating various representative dataflows, we also explore the data reuse and data multicast with wireless technology on our proposed e-WiNN architecture. Furthermore, we leverage novel circuit design by utilizing Dadda-algorithm based Multiply-and-Accumulate and explored the impact of multiple bit precisions (4-bit, 8bit and 16-bit) on hardware performance. To reduce the area overhead, we cluster wireless transceivers into groups of four such that both weights and input features can be effectively multicast to reduce the data movement. We considered the link budget and the required RF power for the center frequency at maximum inter-PE distance. Our detailed RTL modeling and cycle-accurate simulation results show that e-WiNN achieves 36.3% latency reduction and 76.1% energy saving when compared to state-of-art wire interconnected accelerators; 70.3% area reduction and 41.6% energy saving at the cost of 11% latency increase when compared to prior wireless accelerators on various neural networks (AlexNet, VGG16, and ResNet-9/50). In the future, we will extend e-WiNN to adapt to emerging applications and multi-DNN workloads which require flexible dataflows. Further, we will evaluate flexible and reconfigurable wireless connections to improve power and performance for various DNN applications.

Acknowledgment

This research was partially supported by National Science Foundation grants CCF-1513606, CCF-1703013, and CCF-

1901192.

REFERENCES

[1] V. Sze, Y.-H. Chen, T.-J. Yang, and J. S. Emer, "Efficient processing of deep neural networks: A tutorial and survey," *Proceedings of the IEEE*, vol. 105, no. 12, pp. 2295–2329, 2017.

[2] C. Latotzke and T. Gemmeke, "Efficiency versus accuracy: A review of design techniques for dnn hardware accelerators," *IEEE Access*, vol. 9, pp. 9785–9799, 2021.

[3] W. J. Dally, Y. Turakhia, and S. Han, "Domain-specific hardware accelerators," *Communications of the ACM*, vol. 63, no. 7, pp. 48–57, Jun. 2020. [Online]. Available: https://doi.org/10.1145/3361682

[4] T. Norrie, N. Patil, D. H. Yoon, G. Kurian, S. Li, J. Laudon, C. Young, N. P. Jouppi, and D. Patterson, "Google's Training Chips Revealed: TPUv2 and TPUv3," in *2020 IEEE Hot Chips 32 Symposium (HCS)*, Aug. 2020, pp. 1–70, iSSN: 2573-2048.

[5] N. P. Jouppi, D. Hyun Yoon, M. Ashcraft, M. Gottscho, T. B. Jablin, G. Kurian, J. Laudon, S. Li, P. Ma, X. Ma, T. Norrie, N. Patil, S. Prasad, C. Young, Z. Zhou, and D. Patterson, "Ten Lessons From Three Generations Shaped Google's TPUv4i : Industrial Product," in *2021 ACM/IEEE 48th Annual International Symposium on Computer Architecture (ISCA)*, Jun. 2021, pp. 1–14, iSSN: 2575-713X.

[6] D. DiTomaso, A. Kodi, D. Matolak, S. Kaya, S. Laha, and W. Rayess, "A-winoc: Adaptive wireless network-on-chip architecture for chip multiprocessors," *IEEE Transactions on Parallel and Distributed Systems*, vol. 26, no. 12, pp. 3289–3302, 2014.

[7] M. Sinha, S. H. Gade, W. Singh, and S. Deb, "Data-flow aware cnn accelerator with hybrid wireless interconnection," in *2018 IEEE 29th International Conference on Application-specific Systems, Architectures and Processors (ASAP)*. IEEE, 2018, pp. 1–4.

[8] R. Guirado, H. Kwon, S. Abadal, E. Alarcón, and T. Krishna, "Dataflow-architecture co-design for 2.5 d dnn accelerators using wireless network-on-package," in *2021 26th Asia and South Pacific Design Automation Conference (ASP-DAC)*. IEEE, 2021, pp. 806–812.

[9] A. Gholami, S. Kim, Z. Dong, Z. Yao, M. W. Mahoney, and K. Keutzer, "A Survey of Quantization Methods for Efficient Neural Network Inference," *arXiv:2103.13630 [cs]*, Jun. 2021, arXiv: 2103.13630. [Online]. Available: http://arxiv.org/abs/2103.13630

[10] K. Wang, Z. Liu, Y. Lin, J. Lin, and S. Han, "Haq: Hardware-aware automated quantization with mixed precision," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 8612–8620.

[11] S. Liu, S. Karmunchi, S. Laha, S. Kaya, and A. Karanth, "Winn: Wireless interconnect based neural network accelerator," in *2021 IEEE 39th International Conference on Computer Design (ICCD)*. IEEE, 2021.

[12] T.-J. Yang, Y.-H. Chen, and V. Sze, "Designing energy-efficient convolutional neural networks using energy-aware pruning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 5687–5695.

[13] R. Guirado, H. Kwon, E. Alarcón, S. Abadal, and T. Krishna, "Understanding the impact of on-chip communication on dnn accelerator performance," in *2019 26th IEEE International Conference on Electronics, Circuits and Systems (ICECS)*. IEEE, 2019, pp. 85–88.

[14] Y.-H. Chen, T. Krishna, J. S. Emer, and V. Sze, "Eyeriss: An energy-efficient reconfigurable accelerator for deep convolutional neural networks," *IEEE journal of solid-state circuits*, vol. 52, no. 1, pp. 127–138, 2016.

[15] B. Tiwari, M. Yang, X. Wang, and Y. Jiang, "Data streaming and traffic gathering in mesh-based noc for deep neural network acceleration," *arXiv preprint arXiv:2108.02569*, 2021.

[16] Y.-H. Chen, T.-J. Yang, J. Emer, and V. Sze, "Eyeriss v2: A flexible accelerator for emerging deep neural networks on mobile devices," *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, vol. 9, no. 2, pp. 292–308, 2019.

[17] H. Kwon, A. Samajdar, and T. Krishna, "Rethinking nocs for spatial neural network accelerators," in *2017 Eleventh IEEE/ACM International Symposium on Networks-on-Chip (NOCS)*. IEEE, 2017, pp. 1–8.

[18] ——, "Maeri: Enabling flexible dataflow mapping over dnn accelerators via reconfigurable interconnects," *ACM SIGPLAN Notices*, vol. 53, no. 2, pp. 461–475, 2018.

[19] H. Kwon, P. Chatarasi, M. Pellauer, A. Parashar, V. Sarkar, and T. Krishna, "Understanding reuse, performance, and hardware cost of dnn dataflow: A data-centric approach," in *Proceedings of the 52nd Annual IEEE/ACM International Symposium on Microarchitecture*, 2019, pp. 754–768.

[20] I. Hubara, M. Courbariaux, D. Soudry, R. El-Yaniv, and Y. Bengio, "Binarized neural networks," *Advances in neural information processing systems*, vol. 29, 2016.

[21] W. J. Townsend, E. E. S. Jr., and J. A. Abraham, "A comparison of Dadda and Wallace multiplier delays," in *Advanced Signal Processing Algorithms, Architectures, and Implementations XIII*, F. T. Luk, Ed., vol. 5205, International Society for Optics and Photonics. SPIE, 2003, pp. 552 – 560. [Online]. Available: https://doi.org/10.1117/12.507012

[22] H. Zhang, D. Chen, and S.-B. Ko, "New flexible multiple-precision multiply-accumulate unit for deep neural network training and inference," *IEEE Transactions on Computers*, vol. 69, no. 1, pp. 26–38, 2020.

[23] A. Abdelgawad and M. Bayoumi, "High speed and area-efficient multiply accumulate (mac) unit for digital signal prossing applications," in *2007 IEEE International Symposium on Circuits and Systems*, 2007, pp. 3199–3202.

[24] J. Wu, A. K. Kodi, S. Kaya, A. Louri, and H. Xin, "Monopoles loaded with 3-d-printed dielectrics for future wireless intrachip communications," *IEEE Transactions on Antennas and Propagation*, vol. 65, no. 12, pp. 6838–6846, 2017.

[25] Y. Song, Y. Wu, J. Yang, and K. Kang, "The design of a high gain on-chip antenna for soc application," in *2015 IEEE MTT-S International Microwave Workshop Series on Advanced Materials and Processes for RF and THz Applications (IMWS-AMP)*. IEEE, 2015, pp. 1–3.

[26] M. Alibakhshikenari, B. S. Virdee, C. H. See, R. A. Abd-Alhameed, F. Falcone, and E. Limiti, "High-gain metasurface in polyimide on-chip antenna based on crlh-tl for sub-terahertz integrated circuits," *Scientific reports*, vol. 10, no. 1, pp. 1–9, 2020.

[27] Y. S. Shao, J. Clemons, R. Venkatesan, B. Zimmer, M. Fojtik, N. Jiang, B. Keller, A. Klinefelter, N. Pinckney, P. Raina *et al.*, "Simba: Scaling deep-learning inference with multi-chip-module-based architecture," in *Proceedings of the 52nd Annual IEEE/ACM International Symposium on Microarchitecture*, 2019, pp. 14–27.

[28] X. Timoneda *et al.*, "Engineer the channel and adapt to it: Enabling wireless intra-chip communication," *IEEE Transactions on Communications*, vol. 68, no. 5, pp. 3247 – 3258, 2020.

[29] M. Awad, P. Benech, and J. Duchamp, "Design of dickson rectifier for rf energy harvesting in 28 nm fd-soi technology," in *2018 Joint International EUROSOI Workshop and International Conference on Ultimate Integration on Silicon (EUROSOI-ULIS)*. IEEE, 2018, pp. 1–4.

[30] Bsim.berkeley.edu. BSIM-CMG – BSIM Group. (2021, April 6). [Online]. Available: https://bsim.berkeley.edu/models/bsimcmg/

[31] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[32] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[33] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[34] Z. Du, R. Fasthuber, T. Chen, P. Ienne, L. Li, T. Luo, X. Feng, Y. Chen, and O. Temam, "Shidiannao: Shifting vision processing closer to the sensor," in *2015 ACM/IEEE 42nd Annual International Symposium on Computer Architecture (ISCA)*, 2015, pp. 92–104.

[35] T. Luo, S. Liu, L. Li, Y. Wang, S. Zhang, T. Chen, Z. Xu, O. Temam, and Y. Chen, "Dadiannao: A neural network supercomputer," *IEEE Transactions on Computers*, vol. 66, no. 1, pp. 73–88, 2017.

[36] S. Zhang, Z. Du, L. Zhang, H. Lan, S. Liu, L. Li, Q. Guo, T. Chen, and Y. Chen, "Cambricon-x: An accelerator for sparse neural networks," in *2016 49th Annual IEEE/ACM International Symposium on Microarchitecture (MICRO)*, 2016, pp. 1–12.

[37] W. Lu, G. Yan, J. Li, S. Gong, Y. Han, and X. Li, "Flexflow: A flexible dataflow accelerator architecture for convolutional neural networks," in *2017 IEEE International Symposium on High Performance Computer Architecture (HPCA)*. IEEE, 2017, pp. 553–564.

[38] N. P. Jouppi, C. Young, N. Patil, D. Patterson, G. Agrawal, R. Bajwa, S. Bates, S. Bhatia, N. Boden, A. Borchers *et al.*, "In-datacenter performance analysis of a tensor processing unit," in *Proceedings of the 44th annual international symposium on computer architecture*, 2017, pp. 1–12.

[39] H. Kwon, L. Lai, M. Pellauer, T. Krishna, Y.-H. Chen, and V. Chandra, "Heterogeneous dataflow accelerators for multi-dnn workloads," in *2021 IEEE International Symposium on High-Performance Computer Architecture (HPCA)*. IEEE, 2021, pp. 71–83.

[40] S. Gudaparthi, S. Narayanan, R. Balasubramonian, E. Giacomin, H. Kambalasubramanyam, and P.-E. Gaillardon, "Wire-aware architecture and dataflow for cnn accelerators," in *Proceedings of the 52nd*

annual ieee/acm international symposium on microarchitecture, 2019, pp. 1–13.
[41] S. Dave, A. Shrivastava, Y. Kim, S. Avancha, and K. Lee, "Dmazerunner: Optimizing convolutions on dataflow accelerators," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 1544–1548.
[42] S. Deb, K. Chang, X. Yu, S. P. Sah, M. Cosic, A. Ganguly, P. P. Pande, B. Belzer, and D. Heo, "Design of an energy-efficient cmos-compatible noc architecture with millimeter-wave wireless interconnects," *IEEE Transactions on Computers*, vol. 62, no. 12, pp. 2382–2396, 2012.
[43] B. Bahrami, M. A. J. Jamali, and S. Saeidi, "A novel hierarchical architecture for wireless network-on-chip," *Journal of Parallel and Distributed Computing*, vol. 120, pp. 307–321, 2018.
[44] A. Kodi, K. Shifflet, S. Kaya, S. Laha, and A. Louri, "Scalable power-efficient kilo-core photonic-wireless noc architectures," in *2018 IEEE International Parallel and Distributed Processing Symposium (IPDPS)*. IEEE, 2018, pp. 1010–1019.
[45] M. M. Rahaman and P. Ghosal, "Whms: An efficient wireless noc design for better communication efficiency," in *Computational Intelligence in Pattern Recognition*. Springer, 2020, pp. 325–337.
[46] N. Ashokkumar, P. Nagarajan, and P. Venkatramana, "3d (dimensional)—wired and wireless network-on-chip (noc)," in *Inventive Communication and Computational Technologies*. Springer, 2020, pp. 113–119.
[47] G. Ascia, V. Catania, A. Mineo, S. Monteleone, M. Palesi, and D. Patti, "Improving inference latency and energy of dnns through wireless enabled multi-chip-module-based architectures and model parameters compression," in *2020 14th IEEE/ACM International Symposium on Networks-on-Chip (NOCS)*. IEEE, 2020, pp. 1–6.
[48] R. Guirado Liñan, "Wireless chip-scale communications for neural network accelerators," B.S. thesis, Universitat Politècnica de Catalunya, 2019.

Siqin Liu (S'19) received the B.S. degree in optical information science and technology from the Wuhan University, Wuhan, China in 2011. During the undergraduate study, he was rewarded the first prize of the national electrical design contest in China. After graduation, he joined Lingcom Ltd., a digital signal processing platform provider, and was employed as a electrical hardware engineer for 8 years. He is currently a Ph.D student in Electrical Engineering and Computer Science department at Ohio University, Athens, OH, USA. His current research interests include computer architecture, Deep Neural Networks hardware accelerators and network-on-chips (NoCs).

Talha F. Canan (S'16) received the B.S. degree from Department of Electrical and Electronics Engineering, Bilkent University, Ankara, Turkey, in 2015. Following his work as research engineer in CAD modelling of GaN HEMT devices at the Nanotechnology Research Center in Bilkent University, he joined Ohio University in 2016 to commence his Ph.D. work on nano-scale CMOS integrated systems, logic devices and interconnects for kilo-core computing architectures. His other research interests include tunneling transistors, III-V and III-N semiconductor devices, avalanche photodiodes, thin-film transistors and flexible electronic integration. He was a recipient of G.E. Smith and G.V. Smith Memorial Engineering Award for Electrical Engineering and Computer Science Department at Ohio University at 2021.

Harsha Chenji (M'07) received the B.Tech. degree in Electrical and Electronics Engineering from NITK Surathkal, India, in 2007, and the M.S. and Ph.D. degrees in Computer Engineering from Texas A&M University, in 2009 and 2014, respectively. In 2014, he joined as a Post-Doctoral Researcher with the Wireless Networks Laboratory at The University of Texas at Dallas. He is currently an Associate Professor with the School of Electrical Engineering and Computer Science, Ohio University, where he also leads the Wireless Systems Research Group since 2015. In 2016, he was selected for the 2016 Visiting Faculty Research Program by the Air Force Research Laboratory (Information Directorate), Rome, NY, USA. His general area of research is computer networking. His current research interests include free-space optical networks and wireless networks.

Soumyasanta Laha (M'14) received MS degree in embedded digital systems with distinction from the University of Sussex, U.K, in 2007 and the Ph.D. degree in electrical engineering from Ohio University, Athens, OH, USA, in 2014. He was a visiting researcher at Intel Inc. in Austria in 2016. He is currently an Assistant Professor at the Department of Electrical and Computer Engineering of California State University, Fresno, CA, USA. He was the recipient of the Stocker Research Fellowship from Ohio University and the New Investigator Research Grant Award from CSUPERB of the California State University. His current research interests include biomedical instrumentation and energy-efficient mm-wave/sub-THz transceiver design in advanced CMOS technology for on-chip wireless communications.

Savas Kaya (SM'07) received the M.Phil. degree from the University of Cambridge, Cambridge, U.K., in 1994, with a focus on polarization insensitive liquid crystal switches and the Ph.D. degree from the Imperial College of Science, Technology and Medicine, London, U.K., in 1998, with a focus on strained Si quantum wells on vicinal substrates. He was a post-doctoral researcher at the University of Glasgow between 1998 and 2001, carrying out research in transport and scaling of Si/SiGe MOSFETs, and fluctuation phenomena in decanano MOSFETs. He is currently a professor with the Russ College of Engineering at Ohio University, Athens. His other interests include transport theory, device modeling and process integration, nanofabrication, nanostructures and nanosensors for flexible electronics integration.

Avinash Karanth (SM'12) received the M.S. and Ph.D. degrees in electrical and computer engineering from the University of Arizona, Tucson, AZ, USA, in 2003 and 2006, respectively. He is currently the Chair of the School of Electrical Engineering and Computer Science in Ohio University, Athens, OH, USA. His current research interests include computer architecture, optical interconnects, chip multiprocessors (CMPs), and network-on-chips (NoCs). He was a recipient of the National Science Foundation CAREER Award in 2011, the Best Paper Award at the ICCD 2013 conference and his papers have been nominated for best paper at the HiPC-2021, DATE-2019, NoCs-2010 and ASP-DAC-2009. He is the Associate Editor for IEEE Transactions on Computers and IEEE Transactions on Cloud Computing journals. He is a member of IEEE and ACM.