A Compute-in-Memory Hardware Accelerator Design With Back-End-of-Line (BEOL) Transistor Based Reconfigurable Interconnect

Yandong Luo, Sourav Dutta, Ankit Kaul, *Graduate Student Member, IEEE*, Sung Kyu Lim, *Senior Member, IEEE*, Muhannad Bakir, *Senior Member, IEEE*, Suman Datta, *Fellow, IEEE*, and Shimeng Yu, *Senior Member, IEEE*

***Abstract*— Compute-in-memory (CIM) paradigm using ferroelectric field effect transistor (FeFET) as the weight element is projected to exhibit excellent energy efficiency for accelerating deep neural network (DNN) inference. However, two challenges exist. On the technology level, the chip area scaling is stalled due to the lack of logic voltage compatible FeFET at leading-edge technology node, e. g. 7nm. On the system level, CIM-based inference engine designs are usually customized for a specific DNN model, lacking the flexibility to support different DNN models. Besides, communication latency varies across different DNN models and can bound the total inference latency. Therefore, a reconfigurable interconnect is desired to be adaptive to different workloads, which can induce high area cost due to the reconfigurable circuit modules. To solve these issues, in this work, a system-technology co-design (STCO) of a monolithic 3D (M3D) reconfigurable CIM accelerator is performed, where back-end-of-line (BEOL) compatible oxide channel MOSFET and FeFET technologies are utilized. On the technology level, W-doped indium oxide (IWO) NMOS is utilized to design area-efficient M3D write circuit. On the system level, a reconfigurable interconnect design that inserts workload-specific express link is proposed, where the IWO-based NMOS and FeFET are adopted as the building element of the mux and crossbar switch in the router. The algorithm for interconnect configuration is also devised to achieve optimal latency for different workloads. From the system-level evaluation results, M3D IWO FeFET design (utilizing a hybrid 22nm/7nm M3D partition) shows 3.1× times higher energy efficiency than a 7nm 2D SRAM design with comparable chip area. With the proposed reconfigurable interconnect scheme, the interconnect latency is reduced by 9%∼32% compared to the baseline with a regular mesh network.**

***Index Terms*— Back-end-of-line transistor, compute-in-memory, deep neural network, monolithic 3D integration, system-technology co-design.**

Manuscript received December 18, 2021; revised February 19, 2022 and April 24, 2022; accepted April 26, 2022. Date of publication May 23, 2022; date of current version June 13, 2022. This work was supported in part by the Applications and Systems-Driven Center for Energy-Efficient Integrated NanoTechnologies (ASCENT), one of the Semiconductor Research Corporation (SRC)/Defense Advanced Research Projects Agency (DARPA) Joint University Microelectronics Program (JUMP) centers; and by the Innovative Materials and Processes for Accelerated Compute Technologies (IMPACT), one of the SRC Nanoelectronic Computing Research (nCORE) centers. This article was recommended by Guest Editor J. Kulkarni *(Corresponding author: Yandong Luo.)*

Yandong Luo, Ankit Kaul, Sung Kyu Lim, Muhannad Bakir, and Shimeng Yu are with the School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30332 USA (e-mail: yandongluo@gatech.edu; shimeng.yu@ece.gatech.edu).

Sourav Dutta and Suman Datta are with the Department of Electrical Engineering, University of Notre Dame, Notre Dame, IN 46556 USA.

Color versions of one or more figures in this article are available at https://doi.org/10.1109/JETCAS.2022.3177577.

Digital Object Identifier 10.1109/JETCAS.2022.3177577

I. Introduction and Motivation

COMPUTE-IN-MEMORY (CIM) paradigm is one of the most promising techniques for deep neural network (DNN) acceleration due to its superior energy efficiency [1]. As shown in Fig. 1(a), a typical CIM array design consists of three major components: a memory array with either SRAM (e.g. 8T bit cell) or emerging non-volatile memories (eNVMs) as the weight elements; a write circuit for weight programming, which requires I/O transistor due to the high write voltage (1.5V∼3V) of eNVM devices; and mixed-signal peripheral circuit for partial sum processing including analog-to-digital converter (ADC) and digital shift-and-add. For SRAM-based CIM, 8T-SRAM bit-cell is typically utilized due to the limited read margin of 6T-SRAM when multiple rows are activated.

Among eNVMs technologies, ferroelectric field effect transistor (FeFET) stands out due to its high on-state resistance ($R_{ON} > 100k\Omega$) and compatibility with advanced CMOS technology node, which has been demonstrated on 22nm industrial platform by Global Foundries [2]. According to the simulation results from DNN + NeuroSim [3], a 22nm FeFET CIM array with 2-bit per cell memory precision outperforms that of 7nm 8T-SRAM in terms of the energy consumption, which is plotted in Fig. 1(b). It can be attributed to the higher R_{ON} and higher bit-cell precision of FeFET than SRAM. However, in Fig. 1(c), the area cost for the FeFET CIM array is 9× higher than 7nm 8T-SRAM, which can potentially prevent it from being deployed to edge devices with limited chip area budget. The primary reason can be attributed to the unavailability of FeFET today at advanced technology node (e.g. 7nm), resulting in fabricating the peripheral circuit at relatively old technology node (e.g. 22nm). It is also noticed that the ADC area takes a large portion of the FeFET CIM array.

One solution to the area scaling issue is to harness the monolithic 3D (M3D) integration scheme with circuit block level partitioning [4]. In the M3D scheme, the memory array and write circuit are fabricated on the top tier utilizing design rules for a legacy node (e.g. 22nm), while the mixed signal and logic peripheral circuits are fabricated on the bottom tier

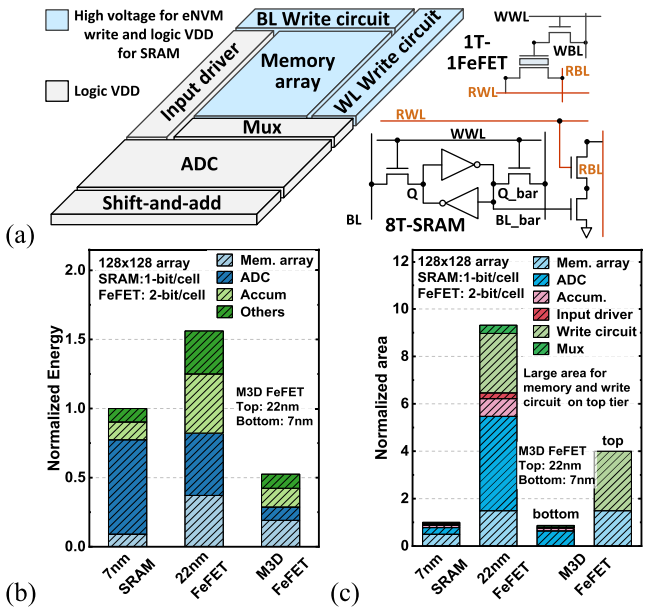

Fig. 1. (a) The schematic of a CIM array with either 8T-SRAM or FeFET as the weight memory cell. (b)-(c) Array level energy and area breakdown from DNN + NeuroSim [3]. 128 × 128 CIM array is assumed. For FeFET M3D design, it is assumed that regular Si CMOS and its I/O process are available on the top tier, which may not be feasible in practice. Thus, this option is listed to illustrate the existing challenges and motivations of this paper.

with a leading-edge node (e.g. 7nm). As plotted in Fig. 1(c) as the option "M3D," the area of ADC is reduced with such an M3D scheme. However, the chip area is still limited by the write circuit using Si I/O transistor, which is attributed to the difficulty to reduce the FeFET write voltage to be compatible with logic power supply (e. g. <1V). Therefore, seeking transistor technology beyond Si is one direction to reduce the area of the write circuit.

Besides the technological challenges mentioned above, another limitation of the architectural design arises from the rigidity of today's CIM accelerator design, which is usually customized for one specific DNN model [3]. Besides, the communication latency takes a large portion of the total inference latency [5]. Since different DNN workloads require different interconnect configurations to optimize the latency, a CIM-based inference engine design with reconfigurable interconnect is desired to provide the flexibility for a wide range of DNN models. Currently, the mainstream reconfigurable interconnect designs set up task-specific topology and/or add dedicated or by-pass/express links to reduce the number of hops between two communication pairs. However, significant hardware overhead can be induced due to the reconfigurable circuit modules. For example, in field programmable gate array (FPGA), the interconnect using NMOS + SRAM routing element consumes up to 70%-80% of the total chip area [6].

The recent research progresses in tungsten-doped indium oxide (IWO) transistor are providing the solution to the aforementioned technological and architectural challenges. The IWO transistor uses semiconducting oxide as the channel material thus it is BEOL compatible due to the low deposition temperature (∼250°C). Besides, it can endure high voltage even with small gate length because of the larger band gap of semiconducting oxide than Si.

Thus, in this paper, we propose to use IWO NMOS [7] and IWO FeFET [8] to design area-efficient M3D write circuit for CIM array and M3D router with crossbar switch at BEOL for reconfigurable CIM architecture, respectively. The contributions of this paper can be summarized as follows.

a) On the system level, a compile time reconfigurable CIM accelerator design is presented. To support the communication pattern of different DNN models, a reconfigurable interconnect with pre-inserted express link is proposed. The configuration algorithm is devised to provide model-optimal communication latency.
b) On the technology level, a practical M3D integration scheme using Si CMOS and IWO NMOS/FeFET is proposed and adopted to the system design. The area cost of CIM array and interconnect is reduced by using M3D write circuit and M3D router design, respectively.
c) A design automation flow for the technology-system co-design is proposed. It includes a mapper to partition and map the DNN model to the hardware, an algorithm to configure the express link based on task-specific communication pattern and the evaluation framework to estimate system level performance.

The rest of the paper is organized as follows. Background and related works are reviewed in Section II. The circuit level designs using IWO-based BEOL transistor are illustrated in Section III. Section IV presents the reconfigurable interconnect and the architecture design of the DNN inference engine. The methodology for the system level performance evaluation is illustrated in Section V and the results are discussed in Section VI. The conclusions are drawn in Section VII.

II. Background and Related Works

A. CIM-Based Inference Engine Design

The CIM paradigm conducts the vector-matrix multiplications (VMM) involved in DNN inference within the memory arrays. High energy efficiency is achieved by reducing the data movement between computing units and memories. In this paradigm, the weight matrix is mapped as different conductance levels of the memory cell. Multi-bit weights are stored into one or multiple memory cells, depending on the cell precision. The input vector can be encoded as either different voltage levels or multiple input cycles. When the input vectors are applied at the horizontal word line (WL), the output vector is obtained as the analog partial sum current along the vertical bit lines (BL). It is then converted to digital values by the analog-to-digital converter (ADC) at the periphery of the array (Fig. 1(a)). Typically, the low-precision ADC is implemented by multi-level sense amplifiers (ML-SA) with different reference current levels. Other sensing schemes such as voltage-mode sensing and charge-based sensing can also be applied. For multi-bit input or multi-bit weight, digital shift-and-add modules are needed to reconstruct the partial sum values.

FeFET is one of the technology options for the weight memory in a CIM array. In an FeFET device, a hafnium-zirconium

oxide (HZO)-based ferroelectric layer is fabricated into the gate stack of a conventional CMOS transistor [9]. Different partially switched polarization states in the ferroelectric layer effectively tune the threshold voltage of the transistor. The polarization state remains even after the gate voltage is removed, which leads to non-volatile storage. The FeFET is programmed and erased by applying positive or negative voltage to the gate. Multi-bit storage can be achieved with the write-and-verify scheme, which iteratively applies programming pulse and verifies the device conductance. FeFET can potentially be a replacement for the NMOS + SRAM bit-cell in the reconfigurable circuit, as it can store the configuration bit and act as a pass transistor concurrently.

Recently, various designs of CIM-based DNN inference engines have been prototyped. For example, S. Yin *et al.* proposed a 12T-SRAM based design for XNOR-net [10]. X. Si *et al.* reported a twin-8T SRAM macro [11] for DNN model with multi-bit weight. Non-volatile memory technologies such as resistive random access memory (RRAM) [12] and FeFET can also be utilized as the weight element for CIM [13], [14]. Those designs achieve lower leakage power than SRAM-based designs due to the non-volatility of the memory device, which is beneficial for edge devices. Besides, FeFET-based designs achieve higher energy efficiency than RRAM-based designs due to the higher R_{ON} of FeFET [14].

To further reduce the chip footprint, M3D integration scheme is adopted in the design. G. Murali *et al.* propose a design with 40nm RRAM array at the top tier and 16nm periphery circuit at the bottom tier [15]. AccuRed utilizes monolithically integrated RRAM-CIM layers and GPU layers for DNN training [16]. It optimizes the weight mapping to avoid hot-links between the RRAM-CIM layers and GPU layers. However, these designs simply assume the same Si technology is available at each tier, which may not be practical considering the fabrication challenges of growing single-crystalline Si on BEOL.

B. Tungsten-Doped Indium Oxide (IWO) Transistor Technology

In the sequential M3D integration scheme, the processing temperature when fabricating top tier transistors should be constrained below 400°C, in order not to degrade the performance of bottom CMOS transistors and interconnects. Silicon recrystallization by laser annealing [17] and amorphous semiconducting oxides are the two mainstream technology options that meet the thermal budget requirement. In laser recrystallization, an amorphous silicon (α-Si) film is deposited on top of the bottom Si transistor layer. A local high temperature is then generated on the α-Si layer by the laser, which recrystallizes the α-Si and forms poly-Si. The circuits are fabricated on the poly-Si. However, the performance of the top-tier transistor is noticeably degraded compared to the single-crystalline Si on the bottom tier.

For semiconducting oxide-based transistors, IWO is one of the promising channel materials as experimentally demonstrated in ref. [7]. In the IWO material, the tungsten acts as both electron donor and stabilizer by absorbing the oxygen vacancies. Fig. 2 shows the device structure of a double-gated IWO transistor on the top tier, where the IWO channel is sandwiched between the top and back gate electrodes. Thanks to the large bandgap of the IWO, IWO transistor can endure high voltage (2.5V) even with a short gate length of 30nm∼50nm. By integrating HZO into the gate stack, IWO FeFET is obtained. It can potentially be an alternative to the NMOS + SRAM bit-cell and thus saves chip area cost by BEOL fabrication. Compared to laser recrystallization, IWO transistor shows lower leakage (1fA/μm in IWO vs. 10pA/μm in recrystallized-Si) and the capability to be processed at even lower temperature (<250°C in IWO vs. 400°C in recrystallized-Si). More importantly, IWO offers higher endurance to high voltage operation, which is beneficial for the write circuit design.

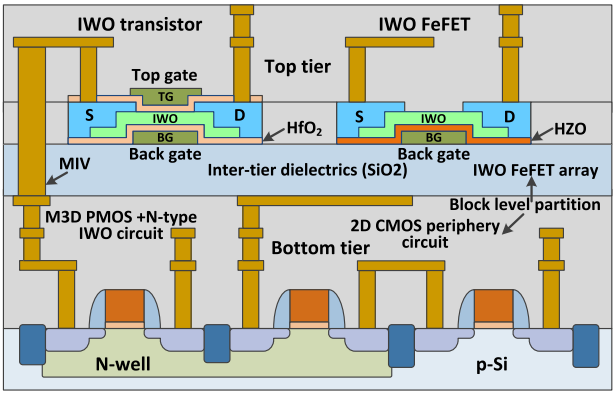

Fig. 2. An illustration of the device structure of a double-gated IWO transistor and a single-gated IWO FeFET. Transistor-level M3D partition can be achieved using IWO NMOS and Si PMOS. The M3D circuit with block level partition can be designed with an IWO FeFET memory array on the top tier and 2D CMOS circuit on the Si substrate.

It should be noted that only NMOS is available for IWO as p-type semiconducting oxide is still elusive [18]. Therefore, in this work, a combination of transistor-level partition and block-level partition is utilized to design the M3D circuit. In the transistor-level partition, IWO NMOS transistor is fabricated on the top tier and PMOS is on the bottom Si substrate, which is interconnected by the metallic inter-tier vias (MIV). The block level partition can be achieved with IWO FeFET memory array on the top tier and 2D CMOS periphery circuit on the bottom Si substrate.

Fig. 3 shows the I-V curve of the IWO transistor, CMOS logic and 2.5V I/O transistor from SPICE simulation. For the IWO transistor, the Verilog-A model in ref. [7] is used in the simulation. Under logic operating voltage (<1V), the performance of IWO transistor is noticeably inferior to CMOS logic due to its relatively low electron mobility (∼20cm^2/V·s). Nevertheless, under high gate voltage (Vgs = 2.5V), IWO transistor shows competitive performance with Si NMOS I/O transistor. Therefore, it can be regarded as a good replacement for NMOS I/O transistor on the top tier.

C. Reconfigurable Accelerator and Interconnect Design

Reconfigurable accelerators are designed to provide flexibility among various types of DNN models. Since different

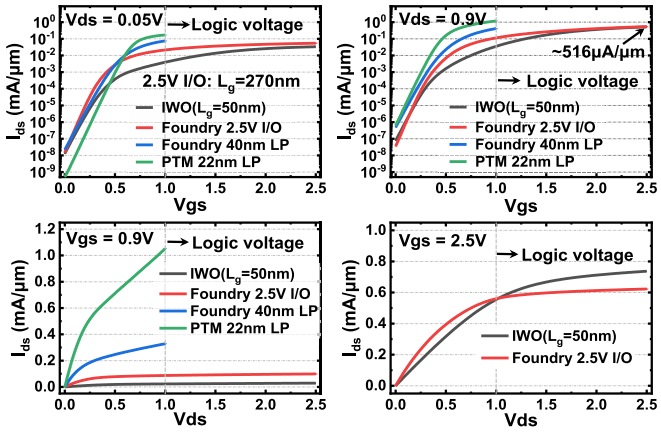

Fig. 3. The I-V curve of the IWO, Si CMOS 2.5V I/O and logic transistor. The IWO transistor shows comparable performance to CMOS 2.5V I/O transistor when driving high voltage (e.g. Vgs = 2.5V).

DNN models require different dataflow and communication patterns, typically a reconfigurable interconnect is adopted in a reconfigurable accelerator design.

The most common reconfigurable design is the FPGA, which utilizes configurable logic block to form circuit modules of different logic functionalities. The configurable logic blocks are connected through a reconfigurable interconnect, which consists of the connection box and switch box (crossbar switch). The connection box connects the configurable logic blocks to the links while the switch box is configured where a turn is made on the signal route. The configuration of both connection box and switch box are determined during compile time. It then remains fixed by storing the configuration bit into the SRAM cell attached to the gate of an NMOS pass transistor (NMOS + SRAM bit-cell). The interconnect area takes as high as 70%∼80% of FPGA chip area [6].

To reduce the area cost of NMOS + SRAM bit-cell, X. Chen *et al.* propose to use FeFET as a replacement [6]. The FeFET can be programmed to the high threshold voltage (V_T) state and low V_T state, which blocks or allows signal to pass through, respectively. However, due to the additional write access transistor in the 1T-1FeFET bit cell, the area benefit is not significant, which is reported to be 23% compared to SRAM at the same technology node [6]. In this paper, the bit-cell design using IWO FeFET and IWO NMOS is proposed. It can further reduce the chip area by fabricating the bit-cell at BEOL.

For digital DNN accelerator design, MAERI utilizes a reconfigurable adder tree to reduce the partial sum from arbitrary number of processing elements [19]. Eyeriss V2 utilizes a reconfigurable mesh network for the input delivery and partial sum collection [20]. For CIM-based reconfigurable accelerator design, A. Lu *et al.* propose a run-time reconfigurable design [21]. However, the interconnect is still based on a conventional H-tree and therefore no optimization is provided for different workloads. As a result, a CIM accelerator design with reconfigurable interconnect is needed to close the gap to its digital counterparts in terms of flexibility.

The reconfigurable interconnect sets up task-specific network topologies to reduce the communication latency. Based on the time when the configuration pattern is determined, it can be classified into compile-time and run-time reconfigurable designs. M. Modarressi *et al.* proposes a hybrid network with routers and routing switch [22], which builds customized topologies for different workloads during compile stage. To bypass the intermediate routers and reduce the hops along the route, links that enable multi-hop traversal (termed as express link in this paper) are adopted. A. Kumar *et al.* propose a design that allows packets in some virtual channels to traverse multiple hops within one cycle [23], which is termed as the express virtual channel (EVC). SMART follows a similar idea but even allows arbitrary multi-hop path with turning [24]. However, in both EVC and SMART, the express links are configured dynamically during run time, as they are targeted at the multi-core systems where the cache access is unpredictable. The run-time reconfiguration leads to latency overhead to set up the route and complicated flow control circuit. Considering the relatively static communication pattern for DNN inference, in this work, the express links are inserted during compile time. The insertion algorithm to achieve model-optimal communication latency is proposed. Besides, technology level innovations are adopted to alleviate the area overhead of the router with express link, which is achieved by partitioning the mux and crossbar switch on the top-tier using IWO transistor and IWO FeFET technology.

III. Circuit-Level Design With BEOL Technology

A. Level Shifter and Output Driver Design

In an FeFET CIM array, the write circuit consisting of level shifter and the output driver is needed to deliver the high programming voltage (2∼4V). As mentioned in Section II.B, IWO transistor shows good endurance to high voltage. Therefore, M3D write circuit can be designed with IWO NMOS and Si PMOS I/O transistor, as shown in Fig. 4(a). Transistor level partition is utilized for the M3D write circuit. The IWO NMOS are fabricated on the top tier with gate length L_g = 50nm while Si PMOS I/O transistor are on the bottom tier with gate length L_g = 270nm according to a 2.5V I/O transistor in TSMC 28nm PDK. The IWO NMOS and Si PMOS are connected using the MIVs between tiers. The parasitic capacitance and resistance of the 3D interconnect are estimated to be 0.18fF and 91Ω, respectively, assuming 6 metal layers at the bottom tier with M1∼M4 pitch = 40nm, M5∼M6 pitch = 64nm and MIV diameter of 30nm.

Fig. 4(b) shows the timing diagram of the write circuit, which is obtained from SPICE simulation using the IWO transistor Verilog-A model [7]. The design using Si CMOS I/O is included for comparison. Two design options are considered for IWO-based M3D design: one with the same transistor size as CMOS (*IWO-same*) and the other with IWO transistor sized up (*IWO-sized_up*). 1V (I/O voltage at 7nm node) and 2.5V is assumed as the VDDL and VDDH, respectively. For *IWO-same*, the delay of the input stage of the output driver is increased by about 0.52ns compared to the CMOS baseline. It is attributed to the degraded I_{ON} of IWO transistor when operating at VDDL, which increases the delay at the falling edge of the inverter. Therefore, when the IWO transistor width

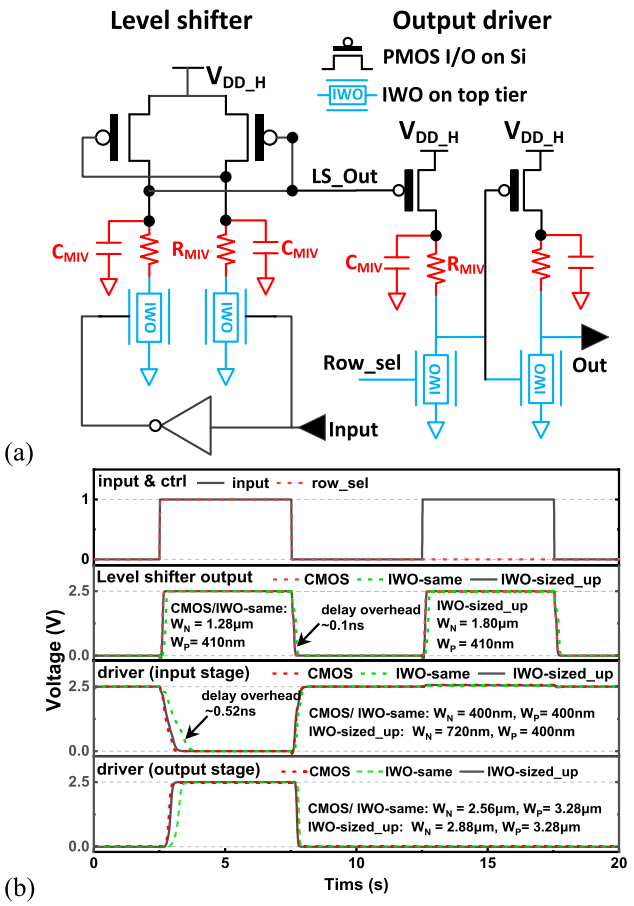

Fig. 4. (a) The schematic of the M3D level shifter with transistor level partition. (b) The timing diagram during the write circuit operation. For IWO-based M3D design, two schemes are considered: one with the same size as Si NMOS I/O transistor (IWO-same) and the other with increased IWO transistor width to match the delay of CMOS design (IWO-sized_up).

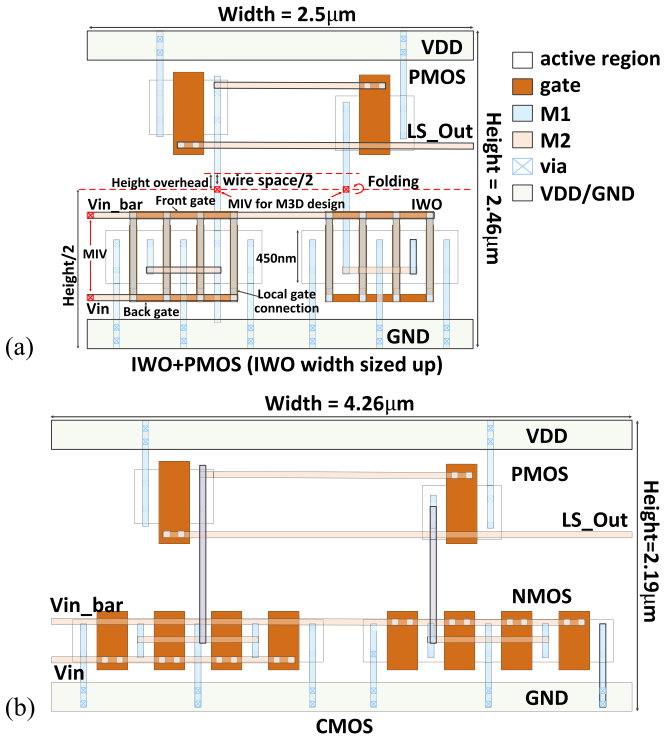

Fig. 5. (a) The layout of the unfolded M3D and (b) 2D Si CMOS level shifter design with 2.5V I/O transistor design rule. The M3D design will be folded at the half-height of the 2D design by inserting MIVs, which introduces height overhead.

of the input stage is increased to 720nm (*IWO-sized_up*), almost the same delay as Si CMOS design is achieved.

The layouts of the M3D and 2D Si CMOS write circuit are shown in Fig. 5. For the M3D design, an unfolded 2D layout is shown for the ease of illustration. It is observed that the width of the IWO-based level shifter is reduced by 41% compared with the CMOS design due to the reduced gate length (50nm in IWO vs. 270nm in Si I/O). The height of the IWO-based level shifter is slightly increased by 0.27μm due to the increased transistor width and the additional gate contact for double-gate device structure. For the M3D write circuit, the layout is partitioned at half-height and then folded, which follows the design strategy proposed by ref. [25]. MIVs are inserted at the cross-section, which induces height overhead of half metal pitch and therefore the area of the M3D design is about 51.2% of the 2D unfolded layout. Overall, the area of the IWO-based M3D level shifter design is only 34% compared to the Si CMOS baseline.

B. Crossbar Switch Design

Crossbar switch (also termed as switch block in FPGA) is a routing component that directs an input to its target output port. The schematic of a 5-port crossbar switch is shown in Fig. 6(a), which is implemented by a group of muxes. The crossbar switch can be configured during either compile time or run time. The former strategy is adopted in FPGA for the interconnect among configurable logical blocks, which remains fixed after configuration. To store the configuration bit, an NMOS + SRAM bit-cell is utilized in the mux, as shown in Fig. 6(b) [6]. The mux with run time reconfigurability is typically utilized in router to handle the varying traffic during run time. As shown in Fig. 6(c), the NMOS transistor is driven by external control signals to select the desired input. Besides, an additional cycle could be required to set up the connection pattern in the crossbar switch. For both cases, level restore circuit is required.

Crossbar switch typically consumes a large portion of chip area, especially for that using NMOS + SRAM due to the large cell size of SRAM. Replacing the NMOS + SRAM bit cell with more compact FeFET is a potential solution. However, as mentioned before, due to the additional write access transistor in 1T-1FeFET bit cell, the area reduction is not significant even compared to SRAM at the same technology node. To further improve the area efficiency, M3D design with IWO FeFET and IWO NMOS (write access transistor) can be utilized. In this scheme, the IWO FeFET and NMOS are fabricated on the top tier with 22nm design rule while the peripheral circuits (e.g. level restorer) are on the bottom tier with 7nm design rule.

The equivalent circuit schematic of an input-output pair is shown in Fig. 7(a). A minimum sized PMOS level restorer is utilized to restore the signal after FeFET and then the output

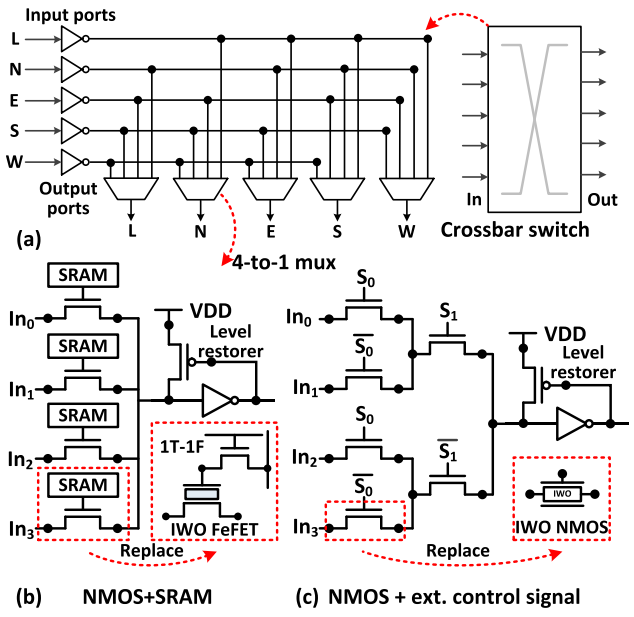

Fig. 6. (a) The schematic of a 5-port crossbar switch implemented by 4-to-1 mux. (b) the mux with NMOS + SRAM bit-cell for compile time reconfigurability. The bit-cell can be replaced by 1T-1 IWO FeFET with more compact bit-cell size and BEOL compatibility. (c) the mux implemented by NMOS for run-time reconfigurability, which is driven by external control signals. The NMOS can be replaced by BEOL compatible IWO NMOS.

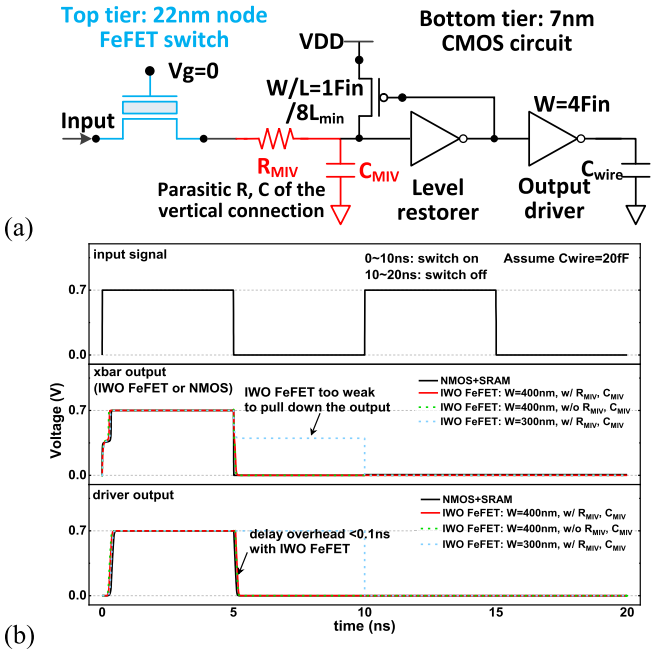

Fig. 7. (a) The equivalent circuit model for an input-output pair in the crossbar switch with 1T-1 IWO FeFET bit cell. (b) The timing diagram of the crossbar switch.

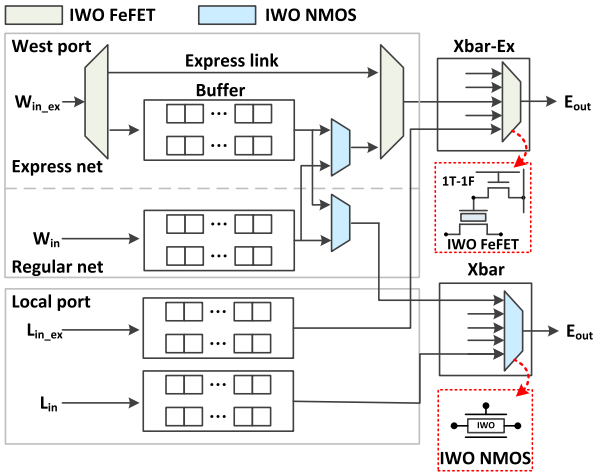

Fig. 8. The schematic of the router in the reconfigurable interconnect. Only the west, local input ports and east output port are shown for the ease of illustration.

drivers deliver the signal across the wires. The width of FeFET and PMOS level restorer should be carefully chosen so that the PMOS is not too strong to pull down the output of FeFET. Similarly, the Si NMOS in the run time reconfigurable mux can be replaced by IWO NMOS at BEOL.

Fig. 7(b) shows the timing diagram of the M3D crossbar switch with IWO FeFET (22nm design rule) and the 2D baseline with NMOS + SRAM (7nm design rule). For IWO FeFET, the parameters of 22nm PTM transistor model [26] are modified to match the I-V curve reported in ref. [27]. It should be noted that low R_{ON} (e.g. $<10k\Omega$) is required for the bit-cell in the crossbar switch, which is opposed to that of weight memory cell. Therefore, width = 400nm is assumed for the IWO FeFET. From the simulation results, only a 0.1ns increase of output delay is observed for IWO FeFET-based crossbar switch, compared to the 7nm NMOS + SRAM baseline. Besides, the impact of parasitic R, C from M3D interconnect is negligible by comparing the results with and without such parasitics. To further reduce the IWO FeFET width from 400nm to 100nm, it is required that electron mobility of the oxide channel be increased by 4× by material engineering. We will use this assumed mobility in the following analysis.

IV. Architecture Level Design With Reconfigurable Interconnect

A. Interconnect and Router Design With Express Link

In this work, a hybrid interconnect design is proposed to reduce the communication latency during DNN inference. It consists of two networks. One is the express network where express links that bypass the intermediate routers are inserted during the compile stage. It can be beneficial for the long-distance communication. During DNN inference, the express links are fixed due to the relatively static communication pattern. This design choice is also limited by the relatively long programming latency for FeFET-based crossbar switch (∼50ns). In this case, the route conflict is resolved during the compile stage. The other network is based on the regular one-hop links (regular link), which is termed as regular network. The route conflicts are resolved during run time and thus provides high throughput by sharing the same physical links in a time multiplexing manner. A packet can either traverse in only one of the networks or through a combination of the two networks before arriving at its destination.

The router micro-architecture in the hybrid network is shown in Fig. 8, where only the local (L) and west (W) input ports, the east (E) output port are shown for illustration

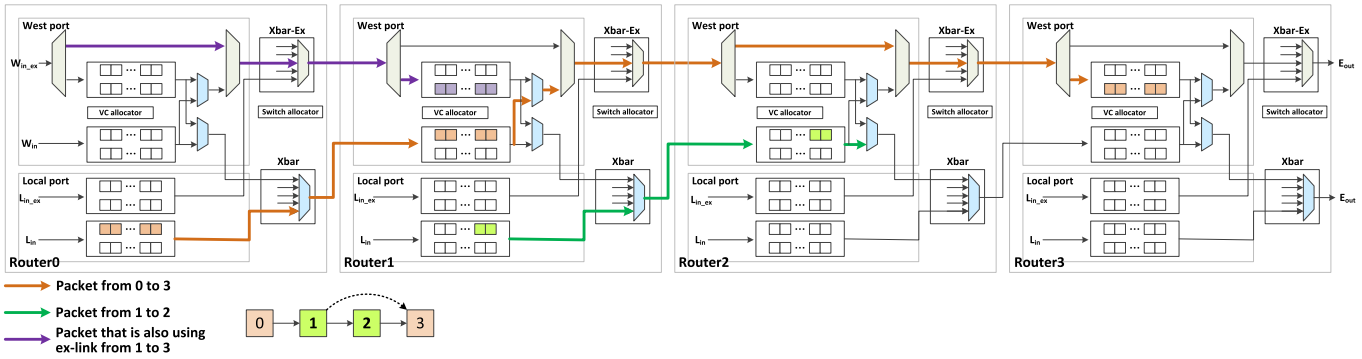

Fig. 9. An illustration of how the router works. Three packets of different source and destination are considered. There is an express link set up between router 1 and 3 to bypass router 2.

purpose. The actual router contains 5 input and output ports. At the non-local input port such as W, the input links are evenly divided as express link and regular link, respectively. Each group of links has its private input buffers. For the input port L, the packets injected by the local processing element can enter either the express network or regular network, depending on the pre-calculated route stored in a look-up table. The two networks have separate crossbar switches so that the packets in different networks can traverse concurrently. For the express network, the crossbar switch (xbar-ex) is based on IWO FeFET, which is configured during the compile stage and then the connection pattern remains fixed. For the regular network, the crossbar switch is based on IWO transistor, where the connection is dynamically determined by the switch allocator.

In the express network, each router can serve as the start node, end node or an intermediate node of an express link. When a router is an intermediate node, the input mux will be pre-configured so that the packets will traverse through the bypass link to skip any router pipeline stages. In this scenario, the path to the input buffer will be deactivated. If the router is an end node, the packet will be buffered and wait for its next hop or be received by the local output port. If the former occurs, the router serves as both the end node of the previous express link and also the start node of the next express link.

The packet can also switch between the express network and the regular network. For example, when switching from express network to the regular network, the packet is first buffered at the input port of the express network. It then undergoes a 2-stage arbitration for the output virtual channel (VC) and the output ports in the regular network. The winner is determined by the VC allocator and the switch allocator. The mux at the input buffer will be properly configured to allow the winner to traverse the crossbar switch in the regular network. Similarly, when switching from the regular link to the express link, the packet arbitrates for output VCs and the output ports in the express network.

Fig. 9 shows an example of packet traversal in the network and how the routers are connected. Assume there are four routers and three packets to traverse. The communication pattern is that router 0 sends data to router 3, router 1 to router 2. At the same time, one packet sent by other routers (not shown) is traversing through this region to its destination. An express link that bypasses router 2 is set up from router 1 to router 3. The packet from router 0 (orange) is first injected into its local port of the regular net and then traverse by 1-hop to arrive at the W input port of router 1. At router 1, it arbitrates for the express link with packet 3 (purple) and eventually wins. Then, it leaves router 1 using the express network, bypass router 2 and finally arrives at router 3. As a result, the router pipeline stages at router 2 are skipped. The local traffic packet 2 (green) is injected at router 1 and arrives at its destination router 2 using the regular network. At router 1, the traversal of packet 2 can start concurrently with packet 1 as they are using a separate network.

B. The Algorithm to Insert Express Link

The pseudo code to determine the optimal insertion scheme for the express links is shown in *Algotrithm1*, which is formatted in Python style. Initially, the weight blocks of the DNN models are mapped onto an array of processing elements (PE). Each PE communication pair is assumed to use the regular network with XY routing scheme. The low-contention latency t_{ij} for communication between PE i and j can be determined using the number of hops along the route H, the router latency t_R and the wire latency t_W, as Eq. (1). P/BW is the serialization delay to split a packet of size P into multiple flits to fit the link bandwidth BW. In this paper, the link bandwidth is assumed to be 128-bit for both the express network and the regular network. The router pipeline is assumed to be 5-stage and thus $t_R = 5$.

$$t_{ij} = H \cdot t_R + H \cdot t_W + P/BW \tag{1}$$

The weighted sum of latency for each PE communication pair is used as the criterion to guide the express link insertion, which is formulated as Eq. (2). w_{ij} is the amount of data to transfer for PE communication pair i, j.

$$T = \sum\nolimits_{(i,j)} w_{ij} \cdot t_{ij} \tag{2}$$

The PE communication pairs are sorted in a descending order by latency t_{ij}. In each iteration, the pair with the longest latency is selected. Then, the possible insertion schemes along the route are enumerated and examined recursively. The express link follows the same XY routing scheme as the regular network. For each possible insertion scheme, its

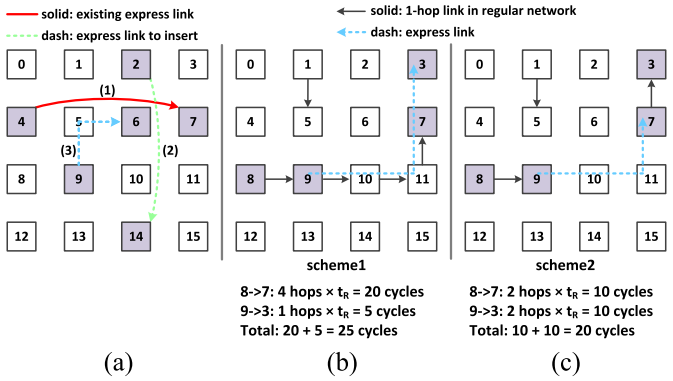

Fig. 10. (a) An illustration of the validity of express link. (b) and (c) two express link insertion schemes for the same communication pairs (9, 7) and (8, 3). An express link is inserted between router 9 and 7.

validity is first examined. In a valid insertion scheme, the input/output ports of each router along the express link should not have been configured. For example, in Fig. 10(a), express link (2) is valid while express link (3) is invalid since the output port E of router 5 is already connected to the input port W for express link (1). When a valid insertion scheme is found, the sum of latency is re-evaluated using Eq. (2). It should be noted that the express links inserted not only can be utilized by the selected route, but also other routes that overlap with the express links. Therefore, when re-evaluating the sum of latency, all the routes that are affected by the express link should be examined. For example, Fig. 10(b) and (c) shows two possible insertion schemes along the route from router 9 to router 3. In scheme 1, a dedicated express link is built from router 9 to router 3, which reduces the number of hops to 1 for communication pair (9, 3). In scheme 2, the express link is inserted between router 9 and router 7. Although the number of hops from router 9 and to router 3 is increased to 2, the number of hops from router 8 to router 7 is also reduced to 2 from 4, resulting in lower total latency than scheme 1. Eventually, the insertion scheme that leads to the minimal sum of latency is selected and the configurations are applied to the crossbar switches along the route. This process is repeated until no more express link can be inserted to any of the PE communication pairs. For each DNN model, the algorithm is executed only once before deployed to the hardware.

Algorithm 1 The Methodology to Insert Express Link

```
# Enumerate all the insertion schemes and find the optimal
def insert_ex_link(start, route, cur_ex_route):
    if start == len(route):
        # Evaluate the route latency when reaching the end
        min_latency = eval_total_latency(cur_ex_route)
        if min_latency < cur_min_latency:
            cur_min_latency = min_latency
            cur_opt_insertion = copy(cur_ex_route)
    else: # insert express link
        for end in range(start + 1, len(route)):
            if is_valid_ex_link(start, end):
            cur_ex_link.add(route[end])
            # Recursively call insert_ex_link until all the nodes along
the route are examined
            insert_ex_link(end, route, cur_ex_route)
            cur_ex_route.pop() # remove the latest node and go to the
next candidate node
last_update = 0
cur_min_latency = eval_total_latency()
while last_update < len(routes):
    # Each time pick the route with longest latency
    # Exit the while loop if no ex-link can be inserted
    max_route = get_route_with_longest_latency(routes)
    cur_opt_insertion = None
    # Start from the first node in route
    insert_ex_link(0, max_route, [ ])
    if cur_opt_insertion: # find an optimal insertion
        # apply the configuration to the xbar switch
        apply_config(cur_opt_insertion)
        last_update = 0
    else:
        last_update += 1
```

The meaning of variables in Algorithm1

start: the index of the starting node
route: a list of nodes along the route.
cur_ex_route: the nodes in the current route with express link
cur_min_latency: the minimal latency up to now. Assumed to be a global variable here.
cur_opt_insertion: the optimal express link insertion scheme up to now. Assumed to be a global variable here.

C. The Architecture of the Reconfigurable CIM Design

The architecture of the reconfigurable CIM design is shown in Fig. 11, which consists of a 24 × 24 processing elements (PE) array and the proposed reconfigurable interconnect. In each PE, there are 4 × 2 FeFET-based CIM arrays with array size to be 144 × 128. In the CIM array, the weights are stored into the BEOL FeFET memory array. The input vector is fetched from the input buffer and delivered to the read word line (RWL) by the input driver. The partial sum current is then sensed by the peripheral ADC. Each ADC is implemented by a multi-level current-mode sense amplifier (ML-CSA), which groups multiple sense amplifiers with different reference current levels. In this work, the ADC precision is assumed to be 5-bit for 2-bit per cell FeFET. The write operation is conducted through the BL and WL level shifter, which drives the write word line (WWL) and write bit line (WBL), respectively. The CIM array utilizes M3D scheme with 22nm and 7nm design rules for the top and bottom tier, respectively. The FeFET memory array is placed on the top tier using IWO FeFET technology. The peripheral circuits are fabricated on the bottom tier with 7nm process, which includes the input driver, ADC and shift-and-add. As aforementioned, the write circuit is partitioned at the transistor level with IWO NMOS on the top and Si I/O PMOS on the bottom tier.

To conduct partial sum reduction, activation and normalization, each PE consists of an SRAM-based input buffer, accumulation unit, special function unit for batch norm (BN), ReLU and pooling. All those modules are fabricated at the bottom tier with 7nm process.

For the router design, the crossbar switches and muxes are placed at the top tier using IWO NMOS or IWO FeFET, which is illustrated in Fig. 8 with different colors. The input buffer and routing logics are fabricated on the bottom tier using 7nm process. Fig. 11(b) and (c) show the layout of a 4-to-1 mux using NMOS + SRAM (7nm design rule) and 1T-1 IWO

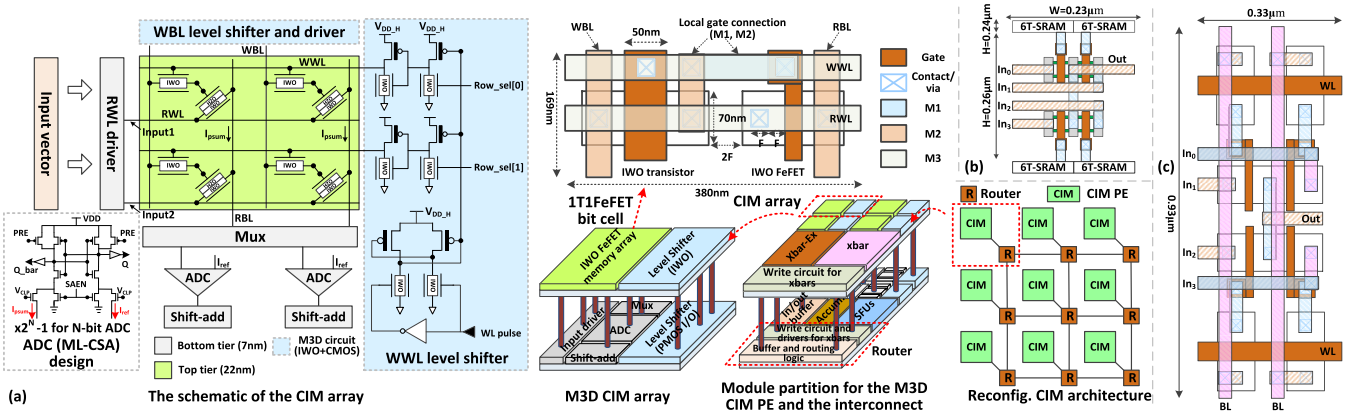

Fig. 11. (a) The array and architecture level design for the reconfigurable CIM accelerator. In each CIM PE, there are 4 × 2 arrays with 144 × 128 array size. The IWO FeFET as weight memory has bit-cell size ∼132F^2 (considering the IWO access transistor for 1T-1FeFET bit cell at 22nm process). The IWO access transistor is single-gate for a compact bit-cell design. (b) and (c) The layout of a 4-to-1 mux in the crossbar switch with 1T-FeFET and NMOS + SRAM bit-cell.

FeFET (22nm design), respectively, which are the building blocks for the crossbar switch. The dimension of a 6T-SRAM is estimated to be 0.24μm × 0.11μm [28]. Although the area of IWO FeFET mux (∼0.31μm^2) is larger than that of NMOS + SRAM (∼0.17μm^2), it should be noted that no Si area is consumed as it is fabricated at BEOL.

V. Methodology for System Level Performance Evaluation

A. The Flow for Reconfiguration and Performance Evaluation

Fig. 12 illustrates the design flow to achieve the compile-time reconfigurability and to evaluate the system level performance. The input is a dataflow graph (DFG) of the DNN model, where each node represents an operation such as vector-matrix multiplication (For Conv. or FC layers), BN, ReLU and pooling etc. Each edge shows the direction of the dataflow. Next, the input DFG is converted to a hardware DFG considering the weight mapping scheme and the architecture-specific information (e.g. PE size). In this work, the conventional CIM weight mapping scheme proposed in ref. [29] is utilized. In this scheme, each filter of a convolutional layer is first unrolled into a long bar and then mapped into a column of the memory array (illustrated in Fig. 12 between step1 and step2). Different filters are mapped into different columns. Similarly, the input feature map in the convolutional window is also unrolled and delivered to each row as the input. In this step, a large weight block is partitioned into multiple small blocks that fit the PE size. It results in additional edges in the hardware DFG for the partial sum reduction among multiple PEs. For each layer, the BN, ReLU and pooling operations are mapped into the PE that produces the final output. During the model mapping stage, simulated annealing is utilized to determine the PE that each weight block in the hardware DFG maps to. The object function is set to minimize the sum of low-contention latency for all the PE communication pairs. Finally, the express links are set up using the algorithm described in Section IV and the route for each PE communication pair is determined.

TABLE I
The Performance of the Circuit Modules in a CIM Array

	2D SRAM	2D FeFET	M3D FeFET
Technology	7nm	22nm	7nm/22nm
Area (μm^2)			
Mem. array	553	1661	1052
Write circuit	45	2790	1323[1]
ADC	314 (4-bit)	4429 (5-bit)	714 (5-bit)
Shift-add	108	832	120
Input driver	32	271	57
Others	61	386	77
Total	1113	10369	2351[2]
Energy (pJ) for inference			
Mem. array	1.9	7.9	4.0
Write circuit	--	--	--
ADC	14.5	9.6	2.0
Shift-add	2.7	9.1	2.9
Input driver	2.0	6.6	2.2
Others	0.1	0.0	0.0
Total	21.2	33.2	11.1

1. The area of the M3D write circuit is the maximum of its top tier and bottom tier area.
2. The area of the M3D CIM array is the maximum of its top tier (memory array and the NMOS of write circuit) and bottom tier area.

The performance of the CIM array and the special function units are evaluated using DNN + NeuroSim V1.3 [3]. The technology file of NeuroSim is first updated with the device parameters of the IWO devices, including the I_{ON}, I_{OFF}, gate capacitance, etc. New circuit modules such as the M3D level shifter are then built. The area and energy values of the circuit modules in a CIM array are listed in TABLE I. The energy consumption is obtained for 1-bit input.

The router design is synthesized using Synopsys Design-Vision at 7nm and 22nm nodes, respectively. For 7nm node, the standard cell library of ASAP7 is used [30]. For 22nm node, the circuit level area and performance are first obtained using a commercial 28nm standard cell library. Then, the values are projected to 22nm node considering the scaling

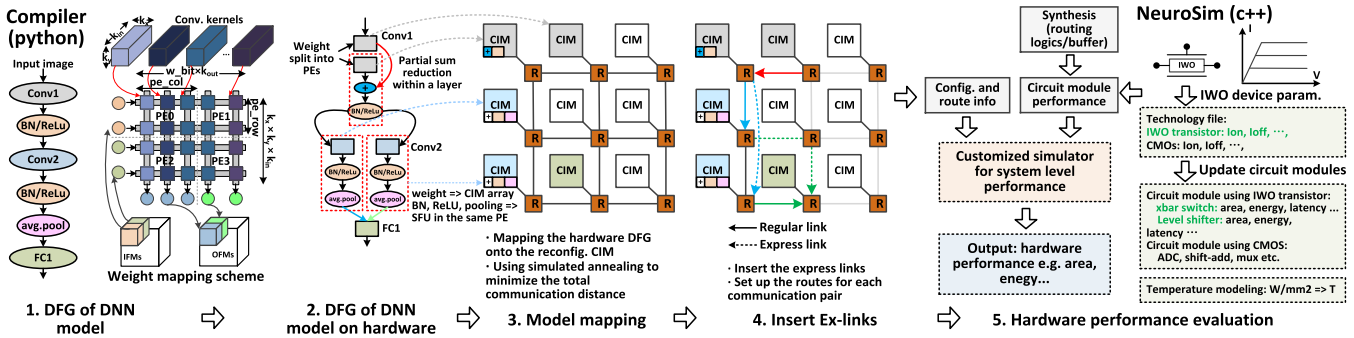

Fig. 12. The design flow to compile and map the DNN model onto the reconfigurable CIM accelerator. The input is the dataflow graph (DFG) of the DNN model. Then the weight of each layer is partitioned into blocks according to the weight mapping scheme and the PE size. The BN, ReLU and pooling operations are mapped onto the special function unit of the PE that produce the final OFM of a layer. The hardware performance is evaluated using M3D NeuroSim with the integrated thermal model [4] and synthesis.

of area and operating voltage. The M3D temperature profile is estimated using the integrated compact thermal model assuming air-cooling for the package [4].

The system level performance is evaluated using the configuration information from the previous design flow and the circuit level performance from the NeuroSim and synthesis. A customized simulator is built to simulate the computing latency.

B. Assumptions for the Technology

Three technology options are considered: 7nm 2D SRAM, 22nm 2D FeFET and hybrid M3D design (22nm top/7nm bottom), as listed in TABLE II. For all the designs, the ADC is implemented with multi-level current-mode sense amplifiers.

M3D SRAM that partitions the NMOS and PMOS in an SRAM bit-cell is not considered due to the inferior performance of IWO transistor technology under logic operating voltage. The cell size of a 7nm 8T-SRAM cell is estimated to be about $688F^2$ [31] assuming F = 7nm for normalization purpose. The precision of an SRAM cell is 1 bit. The crossbar switch in the router utilizes SRAM + NMOS bit-cell structure for the express network.

For the 22nm 2D FeFET option, both the FeFET array and the peripheral circuit are fabricated at 22nm node, which is limited by the availability of FeFET at advanced technology node. The R_{ON} is estimated to be about 110kΩ with 70nm transistor width [2]. A Si I/O transistor is assumed as the write access transistor for the 1T-1FeFET bit-cell. For the M3D FeFET option, the R_{ON} is estimated to be about 180 kΩ [27]. The cell size is estimated to be $132F^2$ assuming 1T-1FeFET bit-cell structure with IWO transistor as the access transistor (Fig. 11(a)). The cell-bit precision of both the 2D FeFET and BEOL FeFET is assumed to be 2-bit. Therefore, an 8-bit weight is split into 4 cells and the partial sums of different weight significance are added up with shift-and-add. Besides, for both options, the crossbar switch of the express network is implemented with 1T-1FeFET bit-cell.

C. Assumptions for DNN Models

To demonstrate the reconfigurability, a wide variety of DNN models are considered: ResNet-20 (w/ 0.27M weights) and ResNet-32 (w/0.46M weights), DenseNet-40 (w/1M weighs), VGG-8 (w/12M weights) for CIFAR-10 dataset and ResNet-18 (w/11M weights), DenseNet-121BC (w/7M weights) for ImageNet dataset. The weight and activations are assumed to be 8-bit. All the models are executed on the same 24 × 24 PE array. The PE utilization depends on the model size, which is about 59% when running ResNet-18 model.

TABLE II
THE DEVICE PARAMETERS USED IN THE SIMULATION

	2D SRAM	2D FeFET	M3D FeFET
Technology node	7nm	22nm	Bottom: 7nm Top: 22nm
R_{ON}	--	110kΩ (Si channel)	180kΩ (IWO channel)
Bit cell config.	8T-SRAM	1T (Si I/O)-1F	1T(IWO) -1F
Bit cell size	$0.0337\mu m^2$ ($688F^2$)	$0.101\mu m^2$ ($210F^2$)	$0.064\mu m^2$ ($132F^2$)
Cell precision	1-bit	2-bit	2-bit
ADC precision	4-bit	5-bit	5-bit

D. Baseline Selections

Two groups of baselines are selected to demonstrate the advantages of the proposed technology option and architecture design, respectively. First, to illustrate the supremacy of the proposed 22nm/7nm M3D scheme over 7nm 2D CMOS, the performance of the proposed architecture is evaluated and compared among the technology options listed in TABLE II, which includes 7nm 2D SRAM and 22nm 2D FeFET.

To demonstrate the efficacy of the proposed reconfigurable interconnect, the design using regular one-hop link with the same total bandwidth (256-bit) is selected as the baseline. In this case, both the baseline and the proposed design are using the 22nm/7nm M3D scheme.

VI. RESULTS AND DISCUSSIONS

A. Comparison Among Different Technology Options

The system level energy efficiency (in TOPS/W) is plotted in Fig. 13. On average, the proposed hybrid M3D FeFET design achieves 3.1× higher energy efficiency over 7nm 2D

TABLE III
A COMPARISON WITH STATE-OF-THE-ART DNN ACCELERATOR DESIGNS

	Google's edge TPU [32]	NVIDIA Jetson Xavier NX [33]	Xilinx ZCU102 [34]	TVLSI'21 [15][1]	This work
Platform	Digital ASIC	Edge GPU	FPGA	RRAM-CIM	FeFET-CIM
Implementation	Silicon	Silicon	Silicon	Post-layout	Simulation
Technology	N/A	N/A	16nm	M3D (40nm/16nm)	M3D (22nm/7nm)
Data format	INT8	INT8	INT16	INT8	INT8
Reconfigurability	Yes	Yes	Yes	No	Yes
Frequency (MHz)	500	1100	200	100	200
Power (W)	2	15	23.6	0.69	0.031 (peak)
Energy efficiency (TOPS/W)	2	1.4	0.012	10.6	87.6 (peak)

1. The specs. are reported from a macro with 3×3 PE.

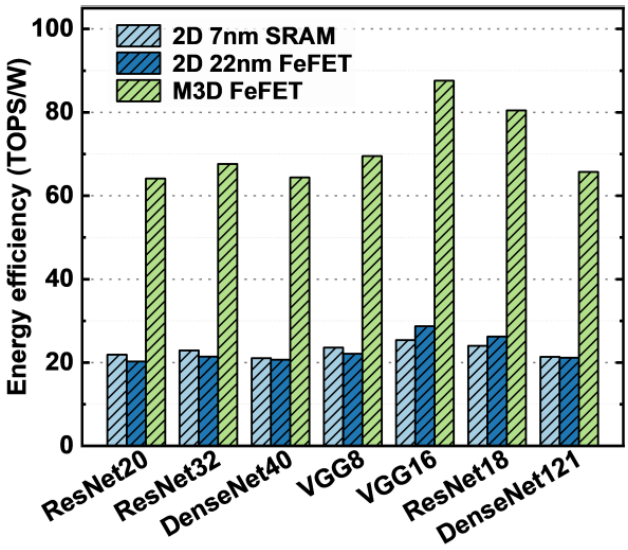

Fig. 13. The system level energy efficiency (TOPS/W) when running different DNN models. The technology options listed in TABLE II are considered.

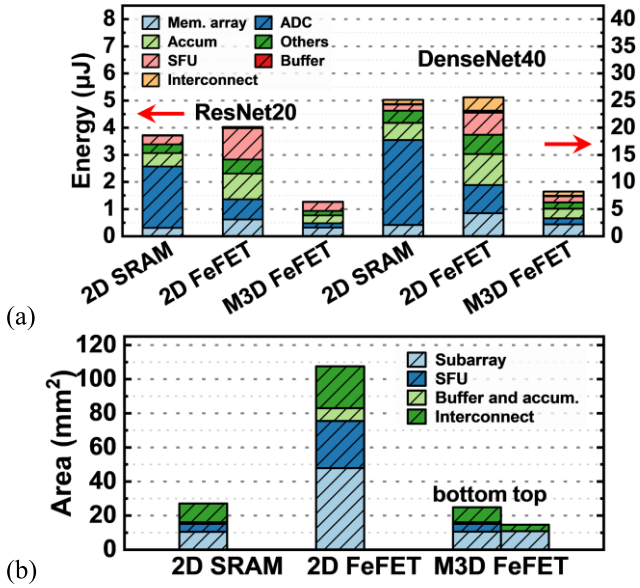

Fig. 14. (a) The system level energy breakdown for one image inference. (b) The chip area breakdown.

SRAM design. It can be attributed to the high R_{ON} of FeFET that reduces the partial sum current, which leads to lower ADC energy consumption as shown in the energy breakdown in Fig. 14(a). Besides, compared to 22nm 2D FeFET design, the proposed M3D design scheme reduces the energy consumption of periphery circuits (e.g. ADC) by enabling the use of 7nm fabrication process.

It is also observed that the energy efficiency varies for different DNN models, where DenseNet shows a relatively low TOPS/W. It can be explained by the low memory array utilization of DenseNet models.

The chip area breakdown of different technology options is plotted in Fig. 14(b). In the proposed M3D design, a balanced area partition of the top and bottom tier is achieved at subarray level by using the transistor-level partition of write circuit. Compared to 22nm 2D FeFET, the proposed M3D design reduces the chip area cost by 4.2× as the peripheral circuits are fabricated at 7nm process node. Besides, by fabricating part of the write circuit and router on the top-tier, the chip area of the proposed design is 7% smaller than the 7nm SRAM baseline.

A summary of the chip performance when running ResNet-20 and ResNet-18 are shown in Fig. 15. These two DNN models represent two extreme cases: one with small memory utilization and the other with high memory utilization. It is observed that the proposed M3D design and 7nm SRAM design show similar compute density (in GOPS/mm^2) due to their comparable chip area cost. However, the energy efficiency of the proposed M3D design is higher than 7nm SRAM, resulting in a lower chip power density (in mW/mm^2). According to the thermal model [4], the chip temperature of the proposed M3D design is as low as 40°C when running ResNet-18.

B. Comparison With Regular Mesh With One-Hop Links

Fig. 16 plots the inference latency for the proposed design with express link and the baseline mesh network with 1-hop link. For DNN models with large fully connected (FC) layers such as VGG8 and VGG16, a significant latency reduction is observed using express link. It can be attributed to the

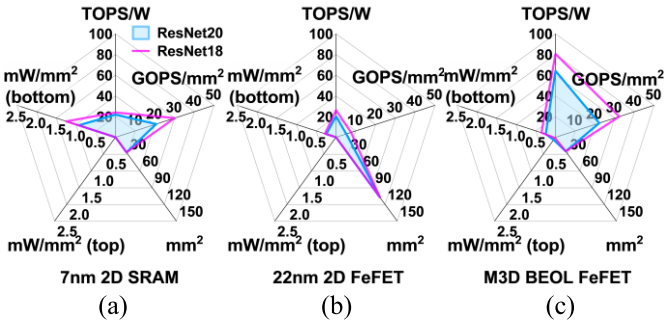

Fig. 15. (a)∼(c) The performance metrics of the proposed reconfigurable CIM accelerator with different technology options. The operating frequency is assumed to be 200MHz for all designs.

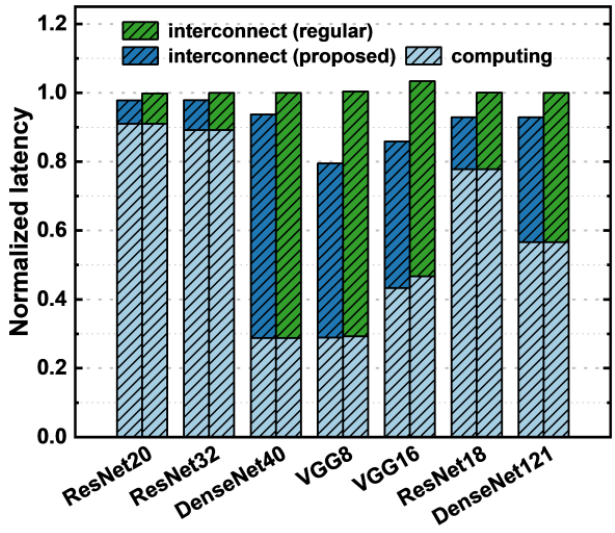

Fig. 16. The normalized DNN inference latency for the proposed reconfigurable interconnect with express link and the regular mesh network with 1-hop link. The same link bandwidth is assumed for both designs.

relatively fewer link conflicts for FC layers and thus dedicated express links can be built for each communication pair. For DenseNet models, the latency reduction becomes modest due to the high link utilization for the densely connected blocks. In this case, a significant portion of the latency originates from the contention delay and thus the benefits of the express link become less. Overall, the interconnect latency is reduced by 9%∼32%, resulting in 2%∼18.9% reduction of the overall inference latency. A lower inference latency reduction is observed for ResNet models since they are compute-bound.

Based on the results in Section A and B, for the proposed design, it can be concluded that the benefits of energy efficiency and chip area originates from the technology level innovation using IWO-based M3D integration scheme, while the latency reduction is attributed to the architectural level design of reconfigurable interconnect with express link.

C. Comparison With State-of-the-Art Inference Engine

A comparison with state-of-the-art inference engine designs is summarized in TABLE III. Google's edge TPU and Nvidia's edge GPU Jetson Xavier NX are selected as the representatives for digital ASIC and GPUs, respectively. Compared to them, the proposed design in this work achieves 43.8× and 62.6× higher energy efficiency, respectively. It can be attributed to the lower DRAM access energy in the CIM paradigm.

FPGA is a reconfigurable computing platform that provides gate-level reconfigurability during the compile time. However, such fine-grained reconfigurability leads to significantly degraded energy efficiency. Therefore, the proposed M3D design achieves ∼760× lower power consumption than Xilinx ZCU102, which is fabricated with 16nm process. Compared to the other M3D RRAM-CIM design [15], the proposed M3D design enables the scaling of both the mixed-signal periphery circuit and write circuit. Therefore, chip fabrication at more advanced technology node is enabled, which results in 8.2× higher energy efficiency.

It should be noted that the metrics of the other designs are either from silicon measurement or post-layout simulation. As an early-stage design space exploration of IWO technology, it is believed that the simulation methodology in this work based on realistic technology parameters is a practical way to project the performance. Developing process design kit (PDK) for IWO transistor technology is crucial to further validate the efficiency of the proposed design. It could be one of our future works.

VII. Conclusion

In this work, a M3D reconfigurable CIM accelerator is designed based on BEOL compatible device technologies. On the technology level, IWO NMOS is utilized to design area-efficient M3D write circuit, which reduces the write circuit area and thus enables further area scaling for FeFET-based CIM array. On the architecture level, to resolve the rigidity of today's CIM-based inference engine, a reconfigurable interconnect design with workload-specific express link is proposed, where the IWO NMOS and FeFET are adopted as the mux and crossbar switch in the router. The system-level evaluation results show that the M3D IWO FeFET design (utilizing a hybrid 22nm/7nm M3D partition) demonstrates 3.1× times higher energy efficiency than a 7nm 2D SRAM design while a comparable chip area is achieved. With the proposed reconfigurable interconnect, the interconnect latency is reduced by 9%∼32% compared to the baseline with regular mesh network.

References

[1] S. Yu, H. Jiang, S. Huang, X. Peng, and A. Lu, "Compute-in-memory chips for deep learning: Recent trends and prospects," *IEEE Circuits Syst. Mag.*, vol. 21, no. 3, pp. 31–56, Aug. 2021, doi: 10.1109/MCAS.2021.3092533.

[2] S. Dunkel *et al.*, "A FeFET based super-low-power ultra-fast embedded NVM technology for 22 nm FDSOI and beyond," in *IEDM Tech. Dig.*, Dec. 2017, pp. 19.7.1–19.7.4, doi: 10.1109/IEDM.2017.8268425.

[3] X. Peng, S. Huang, Y. Luo, X. Sun, and S. Yu, "DNN+NeuroSim: An end-to-end benchmarking framework for compute-in-memory accelerators with versatile device technologies," in *IEDM Tech. Dig.*, Dec. 2019, pp. 32.5.1–32.5.4, doi: 10.1109/IEDM19573.2019.8993491.

[4] X. Peng *et al.*, "Benchmarking monolithic 3D integration for compute-in-memory accelerators: Overcoming ADC bottlenecks and maintaining scalability to 7 nm or beyond," in *IEDM Tech. Dig.*, Dec. 2020, pp. 30.4.1–30.4.4, doi: 10.1109/IEDM13553.2020.9372091.

[5] S. K. Mandal, G. Krishnan, C. Chakrabarti, J.-S. Seo, Y. Cao, and U. Y. Ogras, "A latency-optimized reconfigurable NoC for in-memory acceleration of DNNs," *IEEE J. Emerg. Sel. Topics Circuits Syst.*, vol. 10, no. 3, pp. 362–375, Sep. 2020, doi: 10.1109/JETCAS.2020.3015509.

[6] X. Chen, K. Ni, M. T. Niemier, Y. Han, S. Datta, and X. S. Hu, "Power and area efficient FPGA building blocks based on ferroelectric FETs," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 66, no. 5, pp. 1780–1793, May 2019, doi: 10.1109/TCSI.2018.2874880.

[7] H. Ye *et al.*, "Double-gate W-doped amorphous indium oxide transistors for monolithic 3D capacitorless gain cell eDRAM," in *IEDM Tech. Dig.*, Dec. 2020, pp. 28.3.1-28.3.4, doi: 10.1109/IEDM13553.2020.9371981.

[8] S. Dutta *et al.*, "Monolithic 3D integration of high endurance multi-bit ferroelectric FET for accelerating compute-in-memory," in *IEDM Tech. Dig.*, Dec. 2020, pp. 36.4.1-36.4.4, doi: 10.1109/IEDM13553.2020.9371974.

[9] A. I. Khan, A. Keshavarzi, and S. Datta, "The future of ferroelectric field-effect transistor technology," *Nature Electron.*, vol. 2, no. 10, pp. 580–586, 2020, doi: 10.1038/s41928-020-00492-7.

[10] S. Yin, Z. Jiang, J.-S. Seo, and M. Seok, "XNOR-SRAM: In-memory computing SRAM macro for binary/ternary deep neural networks," *IEEE J. Solid-State Circuits*, vol. 55, no. 6, pp. 1733–1743, Jun. 2020, doi: 10.1109/JSSC.2019.2963616.

[11] X. Si *et al.*, "A twin-8T SRAM computation-in-memory macro for multiple-bit CNN-based machine learning," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2019, pp. 396–398, doi: 10.1109/ISSCC.2019.8662392.

[12] C.-X. Xue *et al.*, "16.1 A 22 nm 4Mb 8b-precision ReRAM computing-in-memory macro with 11.91 to 195.7TOPS/W for tiny AI edge devices," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2021, pp. 245–247, doi: 10.1109/ISSCC42613.2021.9365769.

[13] M. Jerry *et al.*, "Ferroelectric FET analog synapse for acceleration of deep neural network training," in *IEDM Tech. Dig.*, 2017, pp. 6.2.1–6.2.4, doi: 10.1109/IEDM.2017.8268338.

[14] Y. Long *et al.*, "A ferroelectric FET based power-efficient architecture for data-intensive computing," in *Proc. Int. Conf. Comput.-Aided Des.*, San Diego, CA, USA, Nov. 2018, pp. 1–8, doi: 10.1145/3240765.3240770.

[15] G. Murali, X. Sun, S. Yu, and S.-K. Lim, "Heterogeneous mixed-signal monolithic 3-D in-memory computing using resistive RAM," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 29, no. 2, pp. 386–396, Feb. 2021, doi: 10.1109/TVLSI.2020.3042411.

[16] B. K. Joardar, J. R. Doppa, P. P. Pande, H. Li, and K. Chakrabarty, "AccuReD: High accuracy training of CNNs on ReRAM/GPU heterogeneous 3-D architecture," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 40, no. 5, pp. 971–984, May 2021, doi: 10.1109/TCAD.2020.3013194.

[17] F.-K. Hsueh *et al.*, "TSV-free FinFET-based monolithic $3D^{+}$-IC with computing-in-memory SRAM cell for intelligent IoT devices," in *IEDM Tech. Dig.*, Dec. 2017, pp. 12.6.1–12.6.4, doi: 10.1109/IEDM.2017.8268380.

[18] W. Chakraborty, B. Grisafe, H. Ye, I. Lightcap, K. Ni, and S. Datta, "BEOL compatible dual-gate ultra thin-body W-doped indium-oxide transistor with I_{on}=370μA/μm, SS = 73mV/dec and I_{on} /I_{off} Ratio > 4×10^{9}," in *Proc. IEEE Symp. VLSI Technol.*, Jun. 2020, pp. 1–2, doi: 10.1109/VLSITechnology18217.2020.9265064.

[19] H. Kwon, A. Samajdar, and T. Krishna, "MAERI: Enabling flexible dataflow mapping over DNN accelerators via reconfigurable interconnects," in *Proc. 23rd Int. Conf. Architectural Support Program. Lang. Operating Syst.*, Mar. 2018, pp. 461–475, doi: 10.1145/3173162.3173176.

[20] 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 J. Emerging Sel. Topics Circuits Syst.*, vol. 9, no. 2, pp. 292–308, Jun. 2019, doi: 10.1109/JETCAS.2019.2910232.

[21] A. Lu, X. Peng, Y. Luo, S. Huang, and S. Yu, "A runtime reconfigurable design of compute-in-memory based hardware accelerator," in *Proc. Design, Autom. Test Eur. Conf. Exhib. (DATE)*, Feb. 2021, pp. 932–937, doi: 10.23919/DATE51398.2021.9474156.

[22] M. Modarressi, A. Tavakkol, and H. Sarbazi-Azad, "Application-aware topology reconfiguration for on-chip networks," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 19, no. 11, pp. 2010–2022, Nov. 2011, doi: 10.1109/TVLSI.2010.2066586.

[23] A. Kumar, L.-S. Peh, P. Kundu, and N. K. Jha, "Express virtual channels: Towards the ideal interconnection fabric," in *Proc. 34th Annu. Int. Symp. Comput. Archit. (ISCA)*, 2007, pp. 150–161, doi: 10.1145/1250662.1250681.

[24] T. Krishna, C.-H.-O. Chen, W. Cheol Kwon, and L.-S. Peh, "Breaking the on-chip latency barrier using SMART," in *Proc. IEEE 19th Int. Symp. High Perform. Comput. Archit. (HPCA)*, Feb. 2013, pp. 378–389, doi: 10.1109/HPCA.2013.6522334.

[25] Y.-J. Lee, P. Morrow, and S. K. Lim, "Ultra high density logic designs using transistor-level monolithic 3D integration," in *Proc. Int. Conf. Comput.-Aided Design (ICCAD)*, 2012, pp. 539–546.

[26] W. Zhao and Y. Cao, "New generation of predictive technology model for sub-45 nm design exploration," in *Proc. 7th Int. Symp. Quality Electron. Des. (ISQED)*, 2006, pp. 585–590. [Online]. Available: http://ptm.asu.edu/, doi: 10.1109/ISQED.2006.91.

[27] S. Dutta *et al.*, "Logic compatible high-performance ferroelectric transistor memory," 2021, *arXiv:2105.11078*.

[28] J. Chang *et al.*, "12.1 A 7 nm 256 Mb SRAM in high-k metal-gate FinFET technology with write-assist circuitry for low-VMIN applications," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2017, pp. 206–207, doi: 10.1109/ISSCC.2017.7870333.

[29] X. Peng, R. Liu, and S. Yu, "Optimizing weight mapping and data flow for convolutional neural networks on processing-in-memory architectures," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 67, no. 4, pp. 1333–1343, Apr. 2020, doi: 10.1109/TCSI.2019.2958568.

[30] L. T. Clark *et al.*, "ASAP7: A 7-nm FinFET predictive process design kit," *Microelectron. J.*, vol. 53, pp. 105–115, Jul. 2016, doi: 10.1016/j.mejo.2016.04.006.

[31] M. E. Sinangil *et al.*, "A 7-nm compute-in-memory SRAM macro supporting multi-bit input, weight and output and achieving 351 TOPS/W and 372.4 GOPS," *IEEE J. Solid-State Circuits*, vol. 56, no. 1, pp. 188–198, Jan. 2021, doi: 10.1109/JSSC.2020.3031290.

[32] M. Suryavansh. (2019). *Google Coral Edge TPU Board vs NVIDIA Jetson Nano Dev Board—Hardware Comparison*. [Online]. Available: https://towardsdatascience.com/google-coral-edge-tpu-board-vs-nvidia-jetson-nano-dev-board-hardware-comparison-31660a8bda88

[33] Nvidia. (2021). *Jetson Xavier NX: The Tech Specs*. [Online]. Available: https://developer.nvidia.com/embedded/jetson-xavier-nx

[34] L. Lu, J. Xie, R. Huang, J. Zhang, W. Lin, and Y. Liang, "An efficient hardware accelerator for sparse convolutional neural networks on FPGAs," in *Proc. IEEE 27th Annu. Int. Symp. Field-Program. Custom Comput. Mach. (FCCM)*, Apr. 2019, pp. 17–25, doi: 10.1109/FCCM.2019.00013.