An LLM-Friendly CIM-Based Transformer Accelerator with Multi-Head Adaptable Computation Scheme

Yi-Hsuan Pan[1, 2], Hsin-Yu Liu[1], Ping-Hsuan Huang[1], Wei-Cheng Chen[1], Yue-Long Yu[1], Jyun-Jhe Chou[1], Xin-You Liu[1], Chieh-Fang Teng[2], Chih-Wei Chen[2], Pei-Kuei Chung[2], Chi-Sheng Shih[1], Tsung-Te Liu[1]
[1]National Taiwan University, Taipei, Taiwan, [2]MediaTek, Hsinchu, Taiwan,
[1]Email: yhpan@eecs.ee.ntu.edu.tw, hyliu@eecs.ee.ntu.edu.tw, phhuang@eecs.ee.ntu.edu.tw, ttliu@ntu.edu.tw

***Abstract*—This work presents a Transformer accelerator with a large-language-model-friendly (LLM-friendly) architecture, supporting a multi-head adaptable computation scheme. The accelerator features (1) the first token-length-independently fully-pipelined computing-in-memory (CIM) network with 27.1-42.3% energy reduction, (2) an embedding-dimension-adaptable CIM array mapping for truly parallel multi-head computation, and (3) a multi-ping-pong CIM macro with rescheduled attention-head dataflow to enhance 2.05× inference-level area efficiency. Implemented in 28-nm CMOS process, the proposed accelerator consumes 24.62 µJ/token and 2.03 inference/mm² for BERT-base, realizing 2.12× and 2.03× energy and area efficiency improvements over state-of-the-art works.**

***Keywords*—Large Language Model, Computing-In-Memory, Transformer, Multi-Head Attention, Energy-Efficient**

I. Introduction

LLMs excel in natural language processing, but demand increased memory access. Previous CIM-based Transformer accelerators [1-4] face challenges when considered from the practical perspective of LLM, as shown in Fig. 1. (1) Off-chip memory access accounts for over 85% of inference energy consumption. Partially-pipelined [1-2] and token-length-dependent techniques [3-4] all suffer from limited energy efficiency. Moreover, hardware area cost significantly increases as the token sequence length becomes longer. (2) Repetitive token access and low utilization arise in the MHA mechanism across various LLMs, further degrading the processing efficiency. (3) Intensive off-chip memory access dominates the Transformer inference process and causes serious resource idling, severely lowering the overall area efficiency. To address the above challenges, we propose a full-digital CIM-based accelerator featuring (1) a fully-pipelined CIM network, which thoroughly eliminates the intermediate data reloading and enables segmented processing of unconstrained token length, (2) a spatiotemporal-unfolded CIM array realizing fully parallel multi-head computation scheme, (3) a multi-ping-pong CIM (MPP-CIM) macro supporting concurrent writing and computing during runtime, with a rescheduled attention-head dataflow to enhance inference-level area efficiency. Compared to the state-of-the-art works, the proposed design achieves 2.12× and 2.03× improvements in inference-level energy and area efficiency, respectively.

II. Proposed LLM-Friendly Fully-Pipelined CIM Scheme

Fig. 2 shows the overall architecture, comprising a fully-pipelined engine (FPE), a system controller, a customized

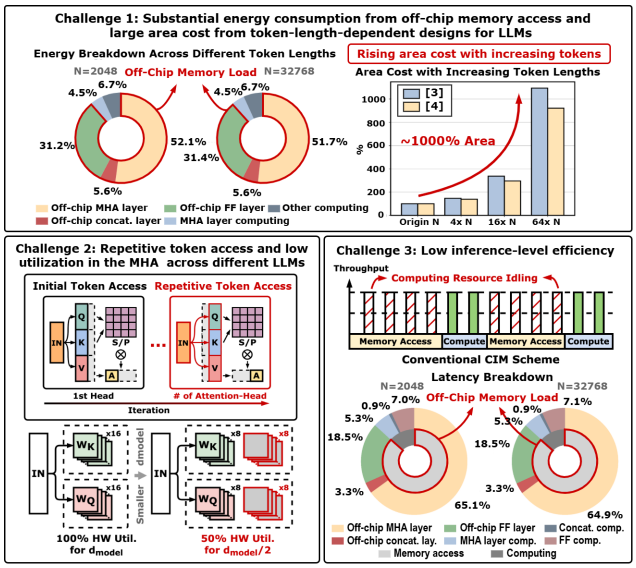

Fig. 1. Challenges of designing a CIM-based accelerator for LLMs.

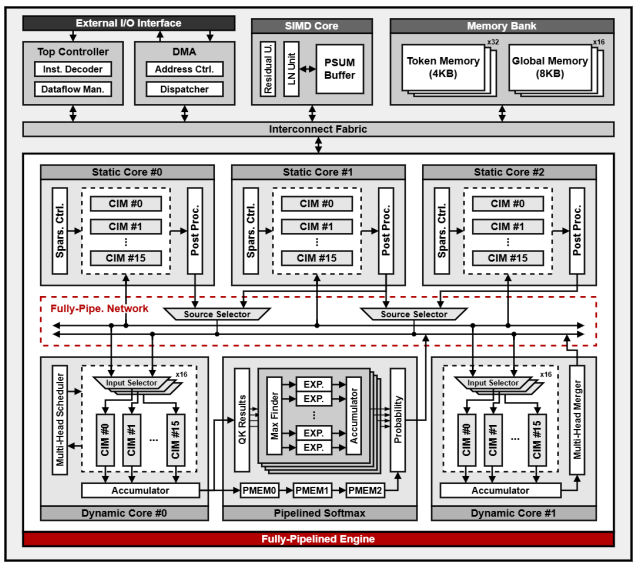

Fig. 2. The overall architecture of proposed CIM-based Transformer accelerator.

DMA, a SIMD core and a 256KB memory bank. The Transformer model inference process is primarily resolved within the FPE, which consists of 3 static cores, 2 dynamic cores, 1 pipelined softmax core (PSC), and a communication network enabling fully-pipelined operation (FPN). The sparse attention computation employs a sliding-window attention scheme with configurable window sizes [6].

Fig. 3 illustrates the data mapping of the entire attention layer onto the FPE. Static cores handle static (matrix-vector-multiplications) MVMs in Q, K and V generation,

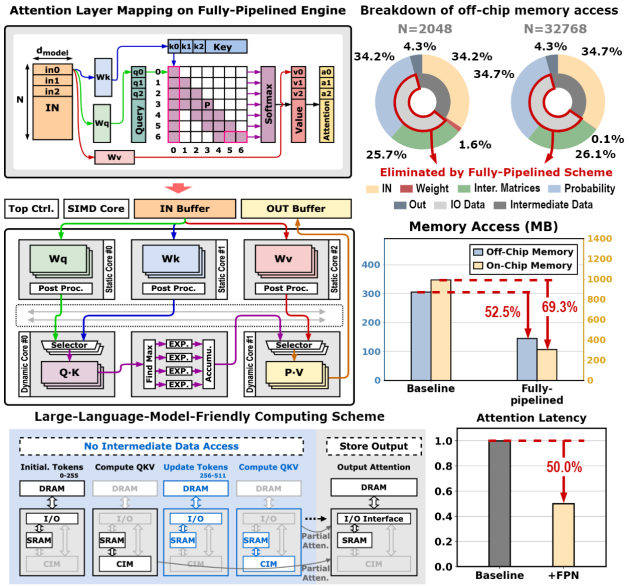

Fig. 3. Proposed LLM-friendly fully-pipelined CIM scheme.

while dynamic cores perform dynamic MVMs of QK^T and PV. The intermediate matrices (Q, K and V) and attention probabilities (P) are directly stored in the FPE after generation. Hence, the proposed fully-pipelined scheme eliminates the intermediate data reloading to maximize the processing efficiency. Besides, to support long sequence lengths of LLMs, the input tokens are divided into multiple segments and sequentially processed within the FPE. To ensure the streaming computation, this work supports continuously updating the used CIM, also preserving the previous token segment during the updating of the next token segment. Consequently, the proposed FPE achieves token-length-independent operation to support LLMs. For the attention layer with token lengths of 2,048-32,768, the FPE reduces off-chip memory access by 50.6-52.5% and achieves energy reductions of 27.1-42.3% when compared to the conventional full-parallel CIM design.

III. Proposed Multi-Head Adaptable CIM Array Mapping

Conventional CIM macros [1,2,4] are typically specialized only for a single model configuration. Therefore, if the operating model is smaller than expected, the hardware utilization would seriously decrease, causing efficiency degradation. In the proposed multi-head adaptable scheme shown in Fig. 4, static-CIM macros within each static core can be arranged by a 2-D flexible mapping strategy, referred to as spatial-unfolded mapping. For models with varying embedding dimensions, the macros are divided into multiple groups, each group exhibiting different attention-head information, allowing for parallel-head computation. Furthermore, the temporal-unfolded mapping leverages CIM blocks to accommodate different attention-head parameters, as shown in Fig. 5. Over time, each CIM block, formed by the same-position rows in CIM banks, is activated in sequence to generate partial intermediate matrices along the hidden dimension of each head. As a result, input tokens stored on-chip can be reused, eliminating the need for repetitive off-chip token access during each head computation. The proposed mixed

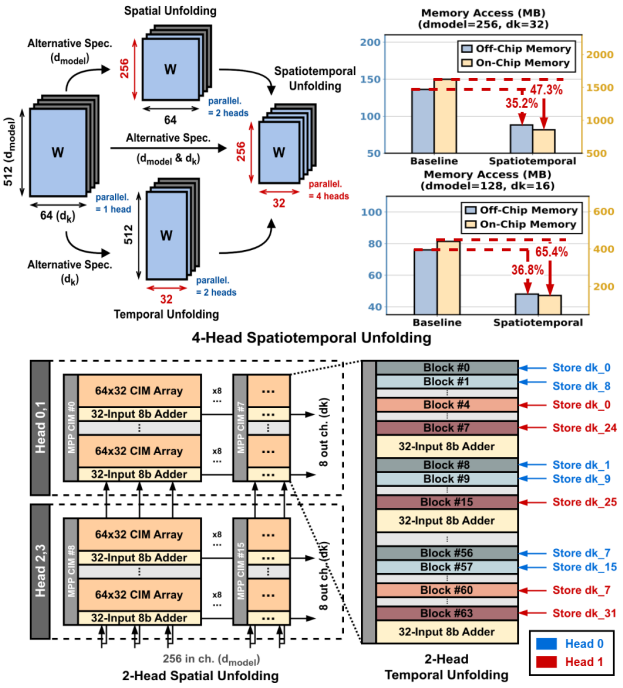

Fig. 4. Proposed multi-head adaptable spatiotemporal-unfolded CIM array mapping scheme.

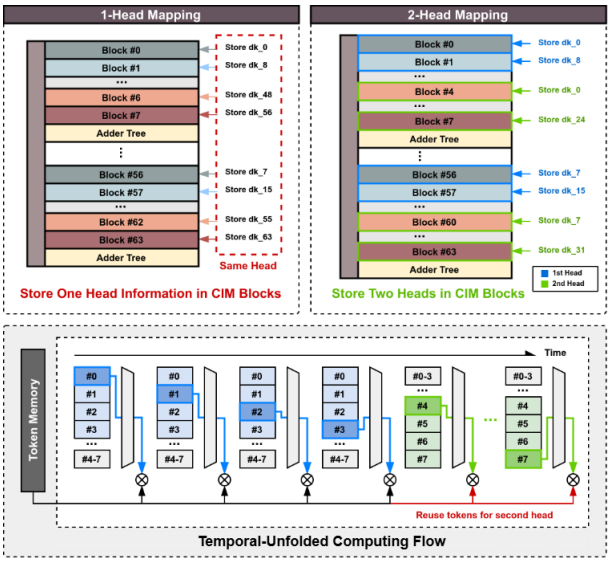

Fig. 5. Detailed dataflow of proposed temporal-unfolded CIM mapping.

spatiotemporal-unfolded mapping simultaneously adjusts to the embedding and hidden dimensions of the Transformer, reducing 35.2-36.8% off-chip memory access and 31.6-58.4% energy consumption while maintaining high hardware utilization across varying LLMs when compared to the same design without spatiotemporal-unfolded mapping, as shown in Fig. 6.

IV. Proposed Rescheduled Attention-Head Dataflow and Multi-Ping-Pong CIM Macro

Conventional CIM designs suffer from low area efficiency for LLM inference due to (1) substantial computing logic demanded by the simultaneous generation of all head channels and (2) prolonged hardware resource idling resulting from extensive off-chip memory access. Hence, we propose a rescheduled attention-head dataflow that supports sequentially producing portions of hidden

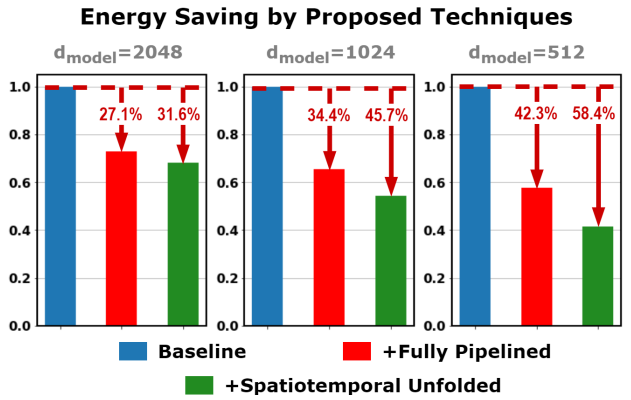

Fig. 6. Energy saving by the proposed LLM-friendly fully pipelined and multi-head adaptable spatiotemporal-unfolded CIM schemes.

dimension in segments by a rescheduling factor R to compute only R × dk channels at one time, as shown in Fig. 7. With a smaller R, the hardware required for static MVM can be reduced. However, this activates only partial CIM in the dynamic core at a time, causing efficiency reduction and an increase in computation latency. To address the efficiency degradation issue, a multi-ping-pong CIM (MPP-CIM) architecture is further proposed to allow different token blocks to be written and computed within the same CIM bank. As shown in Fig. 8, by consolidating computations within the active CIM, the proposed MPP-CIM significantly enhances the hardware utilization in dynamic cores and reduces the required computing logic by R^{-1}. Our analysis shows that R of 1/8 leads to the optimal inference efficiency by exploiting the trade-off between latency and area with different R, as shown in Fig. 7. Fig. 9 shows the corresponding MPP-CIM array design with one-eighth activation for R=1/8, where each CIM bank comprises eight bitcells using a time-sharing 4T-NAND for 1-bit operations. The proposed MPP-CIM bitcell employs the TG-based MUX to enable reliable concurrent writing and computing within a bank above 0.63V. The proposed rescheduled attention-head dataflow and MPP-CIM realizes a 2.05× improvement in area efficiency.

V. Implementation Results

The proposed LLM-friendly CIM-based Transformer accelerator was implemented in 28-nm CMOS technology. Fig. 10 shows the die photo with an area of 6.02 mm^2. Fig. 11 shows the voltage-frequency scaling characteristic of the accelerator. The highest energy efficiency of 23.70 TOPS/W is measured at 0.63V and 50MHz. Table. I shows the performance summary of the proposed accelerator and comparison with the state-of-the-art CIM-based designs. Compared to previous works, this design demonstrates competitive energy and area efficiency at the CIM macro level. At the system level, the proposed LLM-friendly design consumes only 24.62µJ/token and 2.03 inference/mm^2 for BERT-base tasks. This represents the highest inference-level energy and area efficiency with 2.12-2.83× and 2.03-13.27× improvement, demonstrating both superior energy and area efficiency for full LLM model inference.

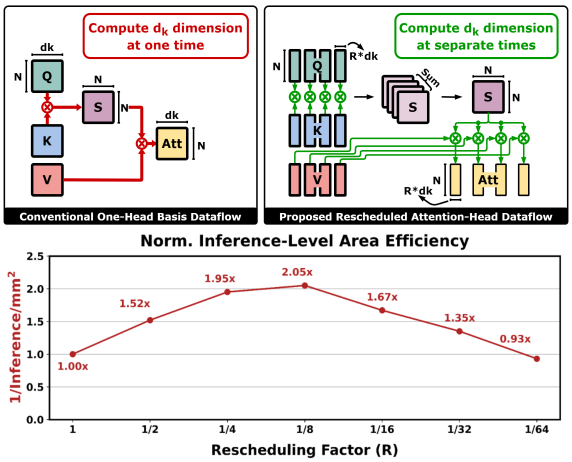

Fig. 7. Proposed rescheduled attention-head dataflow and the corresponding optimal rescheduling factor.

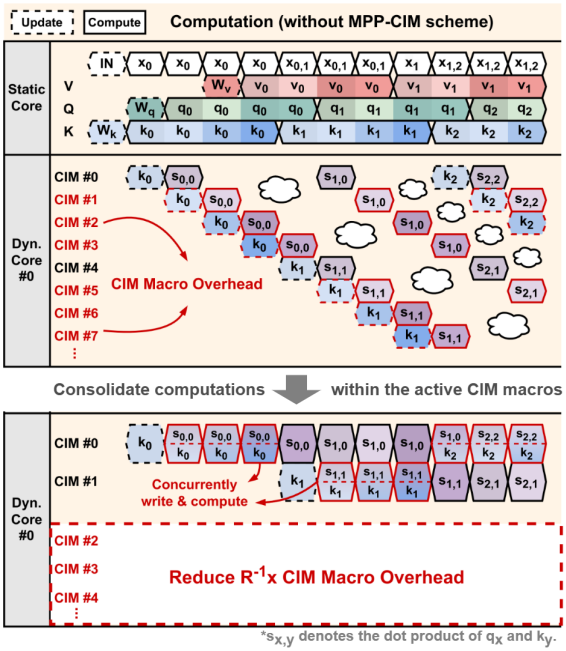

Fig. 8. The reduction in computing logic achieved through the proposed multi-ping-pong CIM scheme for an example case R = ¼.

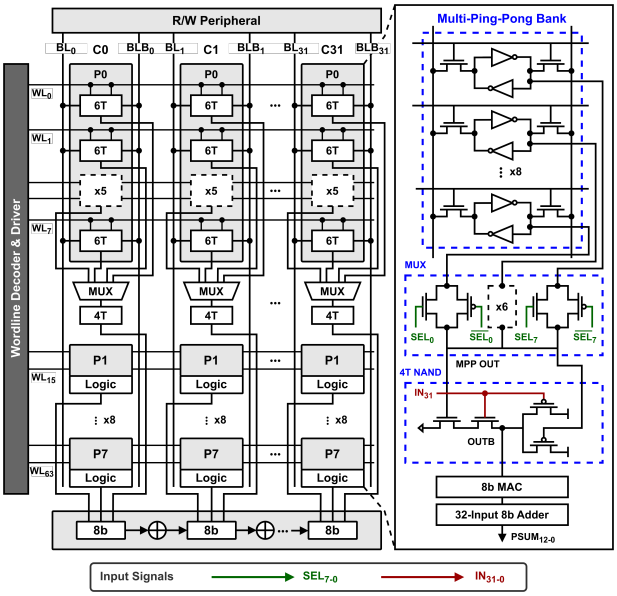

Fig. 9. Proposed multi-ping-pong CIM (MPP-CIM) architecture.

TABLE I. Performance Summary and Comparison.

	[1]	[2]	[3]	[4]	[5]	This Work
Technology (nm)	28	28	28	28	40	**28**
Precision	INT8/16	INT8	INT8	INT4/8/16	Posit8	**INT8**
Supply Voltage (V)	0.6-1.0	0.64-1.03	0.60-1.00	0.90	0.90-1.10	**0.63-1.40**
Frequency (MHz)	80-240	20-320	80-275	50-200	50-95	**50-310**
Die Area (mm^2)	6.83	3.93	6.98	2.00	65.61	**6.02**
CIM Size (KB)	24	80	24	9	N/A	**113**
Multi-Head Scheme	Single-head	Single-head	Sequential-head	Single-head	Single-head	**Parallel-head**
Throughput (TOPS)[a,b]	1.48 (INT8) 0.37 (INT16)	3.33	2.66	0.80 (INT8) 0.40 (INT16)	0.026-0.051	**4.44**
Energy Efficiency (TOPS/W)[a,c]	20.5 (INT8) 5.1 (INT16)	25.22	38.90	23.24 (INT8) 11.62 (INT16)	0.43-0.50	**23.70**
Area Efficiency ($TOPS/mm^2$)[a,b]	0.22 (INT8)	0.85	0.38[d]	0.40 (INT8)[d]	0.008[d]	**0.74**
Energy Consumption (μJ/Token)[c,e]	59.50[d] (2.42×)	69.77[d] (2.83×)	52.17[d] (2.12×)	55.52[d] (2.25×)	N/A	**24.62 (1×)**
Inference-Level Area Efficiency (inference/s/mm^2)[b,e]	0.31[d] (6.94×)	1.00[d] (2.03×)	0.17[d] (13.27×)	0.61[d] (3.37×)	N/A	**2.03 (1×)**

[a.] One operation (OP) represents one multiplication or one addition. [b.] Measured at the highest performance point. [c.] Measured at the highest efficiency point.
[d.] Estimated from the published data. [e.] Off-chip and on-chip memory access are included for analysis.

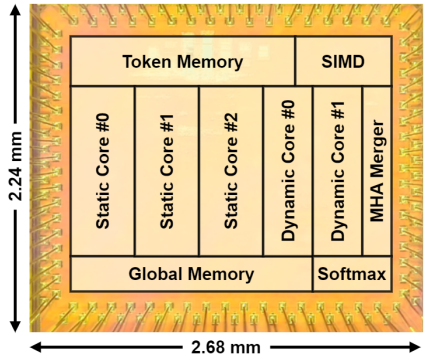

Fig. 10. Die photo of the Transformer accelerator.

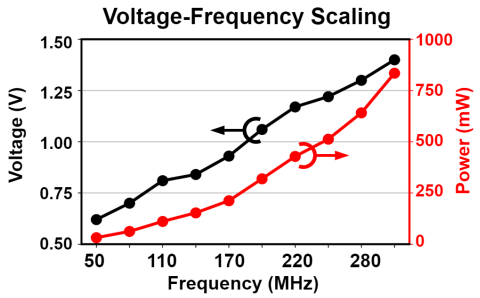

Fig. 11. Measured voltage-frequency scaling characteristic.

VI. Conclusion

This work presents a CIM-based Transformer accelerator featuring LLM-friendly architecture. The proposed fully-pipelined scheme enables the computation of the entire MHA layer on the chip without additional intermediate data access while supporting the arbitrary sequence lengths. In addition, the spatiotemporal-unfolded CIM array mapping methodology realizes fully parallel multi-head computation and accommodates the various model configurations, leading to an overall 31.6%-58.4% improvement in energy reduction. Together with the proposed rescheduled attention-head dataflow, the proposed MPP-CIM achieves a 2.05× improvement in inference-level area efficiency. Implemented in 28-nm CMOS process, the proposed CIM-based accelerator reaches high energy and area efficiencies of 23.70 TOPS/W and 0.74 $TOPS/mm^2$, respectively. When executing the BERT-base model, the proposed LLM-friendly Transformer accelerator consumes 24.62 μJ/token and 2.03 $inference/mm^2$, achieving 2.12-2.83× and 2.03-13.27× improvement in inference-level energy and area efficiency, respectively, when compared to the state-of-the-art works.

Acknowledgment

The authors would like to thank TSMC and TSRI for chip fabrication and technical support. This work was supported by MTK and NSTC.

References

[1] F. Tu et al., "TranCIM: Full-Digital Bitline-Transpose CIM-based Sparse Transformer Accelerator With Pipeline/Parallel Reconfigurable Modes," in IEEE Journal of Solid-State Circuits, vol. 58, no. 6, pp. 1798-1809, June 2023.

[2] S. Liu et al., "16.2 A 28nm 53.8TOPS/W 8b Sparse Transformer Accelerator with In-Memory Butterfly Zero Skipper for Unstructured-Pruned NN and CIM-Based Local-Attention-Reusable Engine," 2023 IEEE International Solid-State Circuits Conference (ISSCC), San Francisco, CA, USA, 2023, pp. 250-252.

[3] R. Guo et al., "CIMFormer: A 38.9TOPS/W-8b Systolic CIM-Array Based Transformer Processor with Token-Slimmed Attention Reformulating and Principal Possibility Gathering," 2023 IEEE Asian Solid-State Circuits Conference (A-SSCC), Haikou, China, 2023, pp. 1-3.

[4] X. Fu et al., "P^3 ViT: A CIM-Based High-Utilization Architecture With Dynamic Pruning and Two-Way Ping-Pong Macro for Vision Transformer," in IEEE Transactions on Circuits and Systems I: Regular Papers, vol. 70, no. 12, pp. 4938-4948, Dec. 2023.

[5] K. Prabhu et al., "MINOTAUR: An Edge Transformer Inference and Training Accelerator with 12 MBytes On-Chip Resistive RAM and Fine-Grained Spatiotemporal Power Gating," 2024 IEEE Symposium on VLSI Technology and Circuits (VLSI Technology and Circuits), Honolulu, HI, USA, 2024, pp. 1-2.

[6] I. Beltagy et al., "Longformer: The Long-Document Transformer," arXiv preprint arXiv:2004.05150 (2020).