CICC

IEEE Custom Integrated Circuits Conference

An 11.16µJ/token Edge SLM Decoder Accelerator with Scalable Ring-based Configuration for Token-level Pipelining in 16nm FinFET

Guanchen Tao[*1], Junyi Luo*[*2], Shiwei Liu[1], Anhang Li[1], Gregory Kielian[3], Kauna Lei[1], Qirui Zhang[1], Dennis Sylvester[1] and Mehdi Saligane[2,3]*

[1]University of Michigan, Ann Arbor, MI, [2]Brown University, Providence, RI, [3]Google Research, Mountain View, CA

**Equally Credited Authors (ECAs)*

April 21st, 2026

IEEE SOLID-STATE CIRCUITS SOCIETY™ | IEEE

Outline

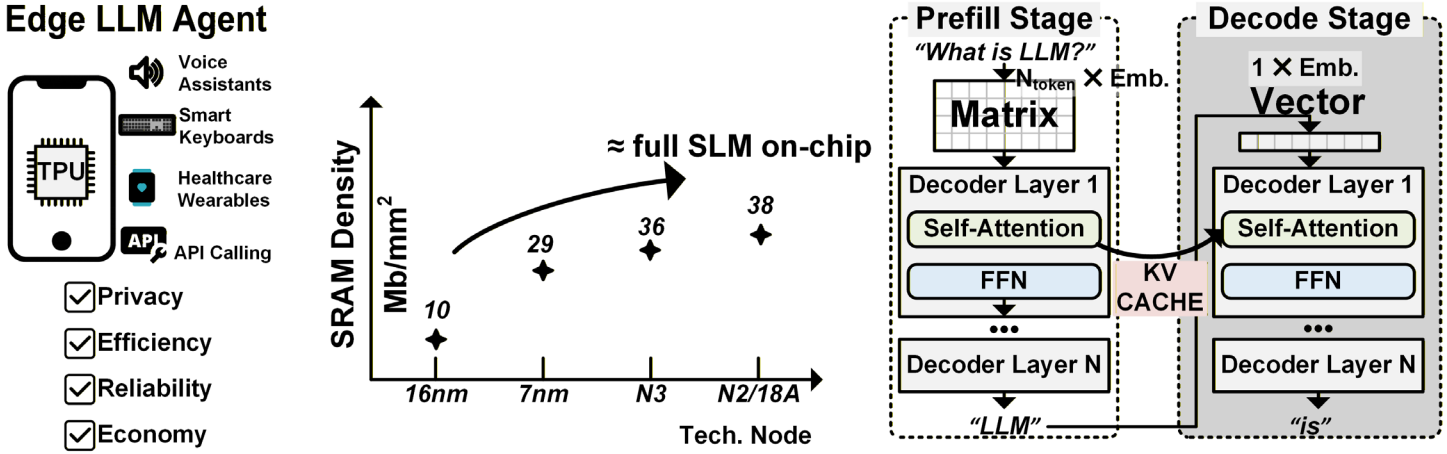
- Introduction
- Overall Architecture
- Key Features
  - Ring-based decoder execution engine (rDXE) with token-level pipelining
  - Vector Recompute (VRC) fusing nonlinear normalization with linear GEMV
  - On-Chip Databuffer Optimizations
- Experiment Results
- Summary

Outline

- Introduction
- Overall Architecture
- Key Features
  - Ring-based decoder execution engine (rDXE) with token-level pipelining
  - Vector Recompute (VRC) fusing nonlinear normalization with linear GEMV
  - On-Chip Databuffer Optimizations
- Experiment Results
- Summary

Sub-billion-parameter Models at the Edge: On-Device AI Is Feasible

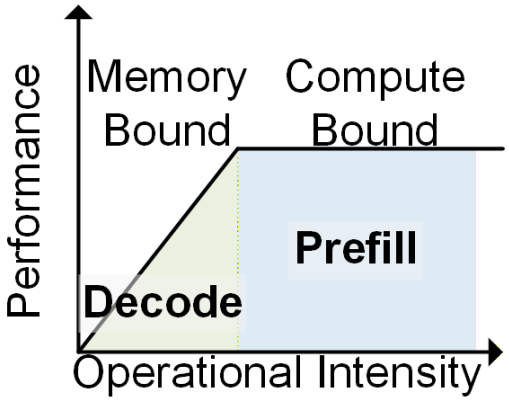
- **SLM**: Privacy, Efficiency, Reliability, Economy.
- **Feasibility**: At 38.1 Mb/mm² SRAM density, a 50 mm² budget fits ~250M INT8 parameters.
- **Challenge**: Compute-bounded Prefill stage and Memory-bounded Decode stage.

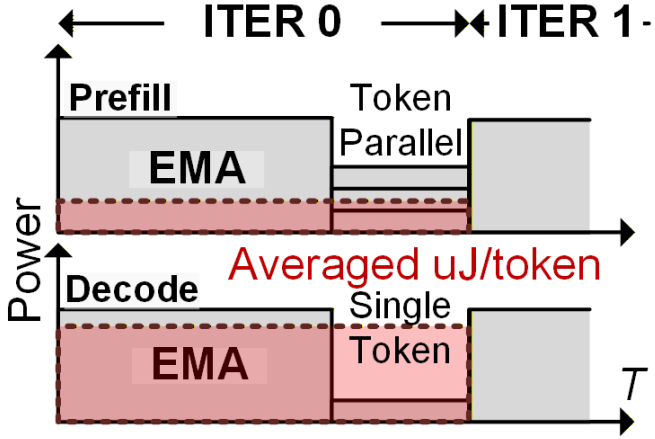


EMA-Limited Prefill and Decode Efficiency

Challenge 1

- Prefill stage benefits from token level parallelism but is compute bound.
- Decode stage is memory-bound and suffers from repeated EMA for each token.
- EMA dominates energy and limits battery life for edge devices.

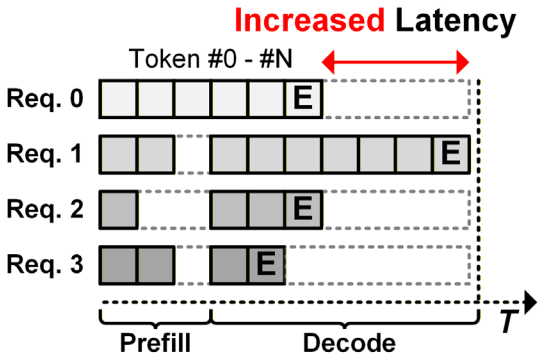




Bottlenecks in Multi-Request Inference

Challenge 2

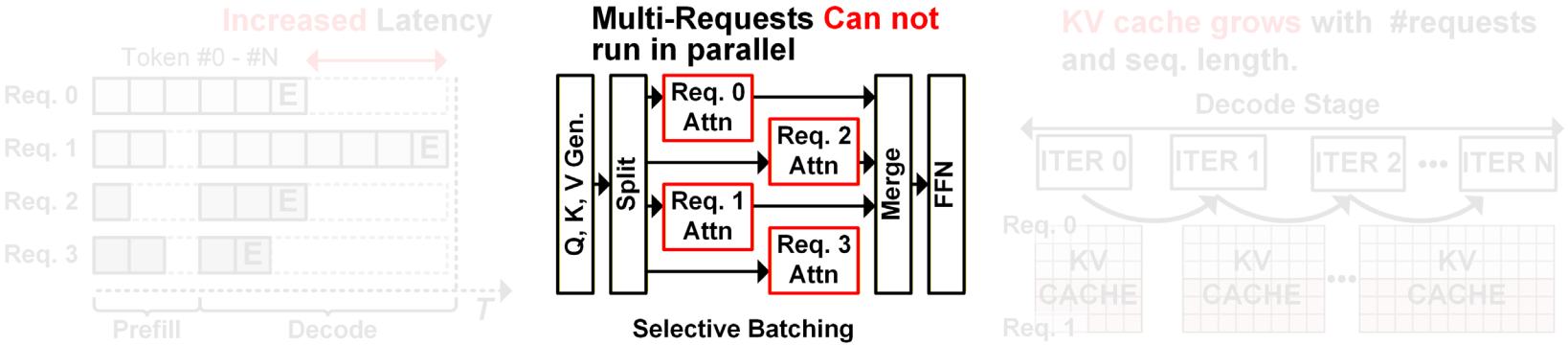
- Prefill: requests with differing sequence lengths must be padded to a uniform length.
- Decode: overall latency is limited by the request with the longest decode sequence.



Bottlenecks in Multi-Request Inference

Challenge 2

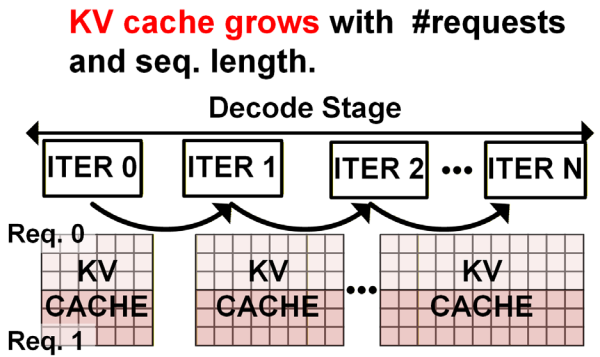
- Selective batching adds complexity to software stack.
- Attention requires per-request QK^T and PV matrix multiplication with its own K/V cache, limiting batch-level parallelism.



Bottlenecks in Multi-Request Inference

Challenge 2

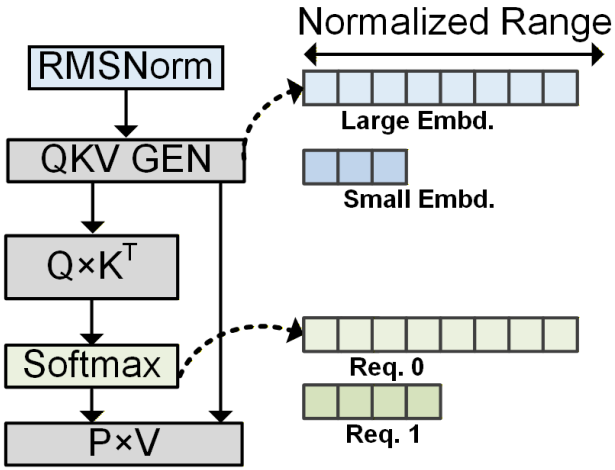
- KV-cache size grows with inference time and scales linearly with the number of concurrent requests.



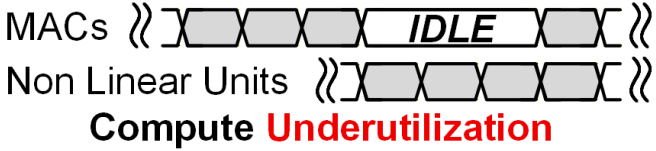
Underutilization from Non-MAC Operations

Challenge 3

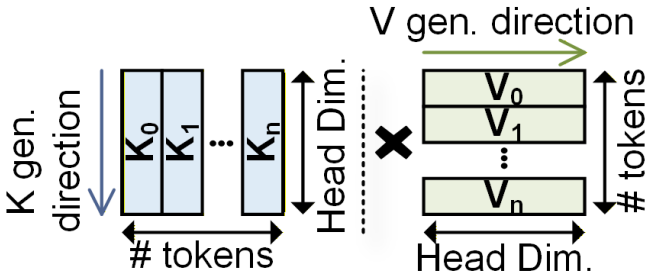
- Edge accelerators suffer from underutilization during non-MAC operations (Softmax, RMSNorm, matrix transpositions).



Latency Varies



Compute Underutilization

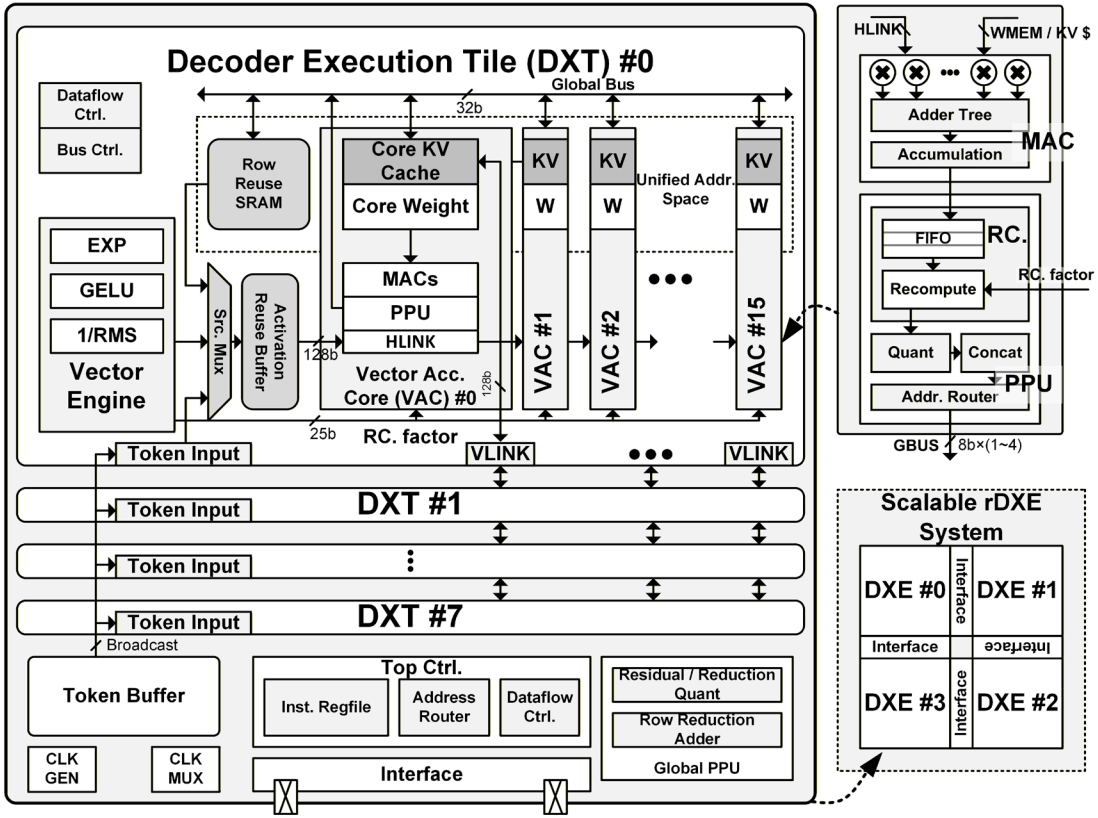


Reshape overhead from K/V dataflow mismatch

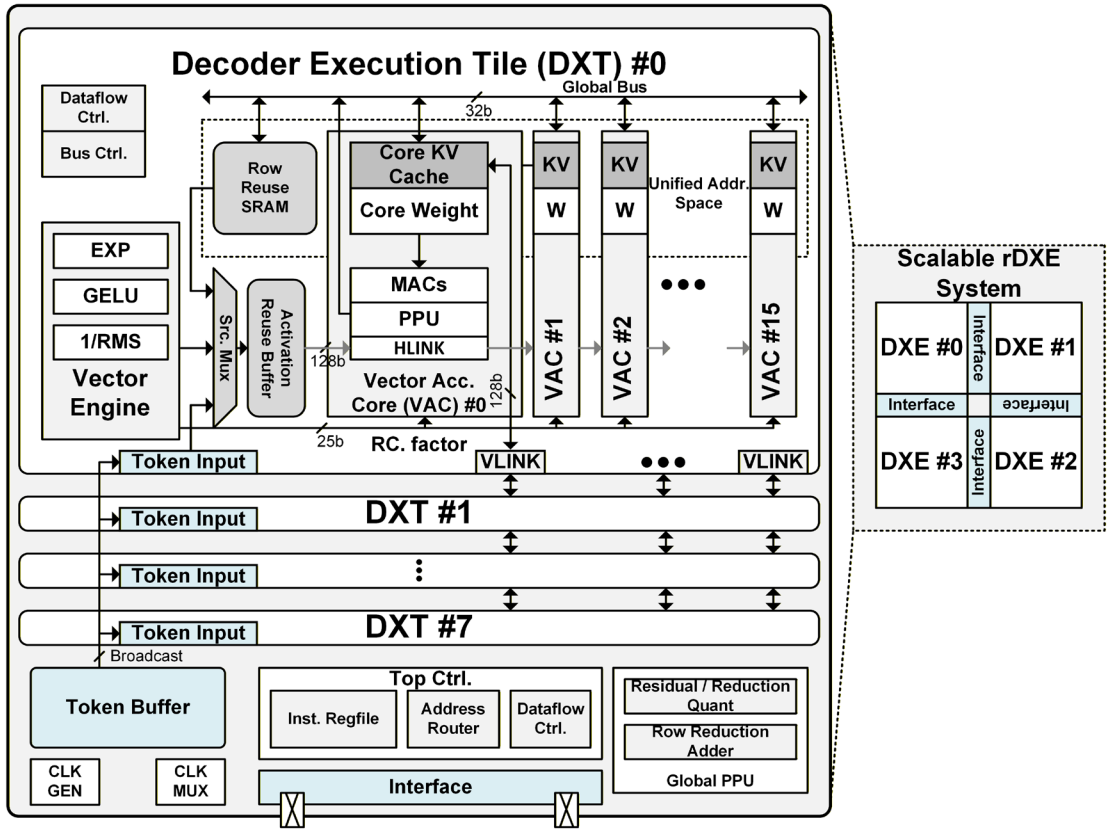
Outline

- Introduction
- Overall Architecture
- Key Features
  - Ring-based decoder execution engine (rDXE) with token-level pipelining
  - Multi-Requests-Multi-Tokens processing in rDXE and layer sharing
  - Vector Recompute (VRC) fusing nonlinear normalization with linear GEMV
  - On-Chip Databuffer Optimizations
- Experiment Results
- Summary

Overall Architecture of DXE



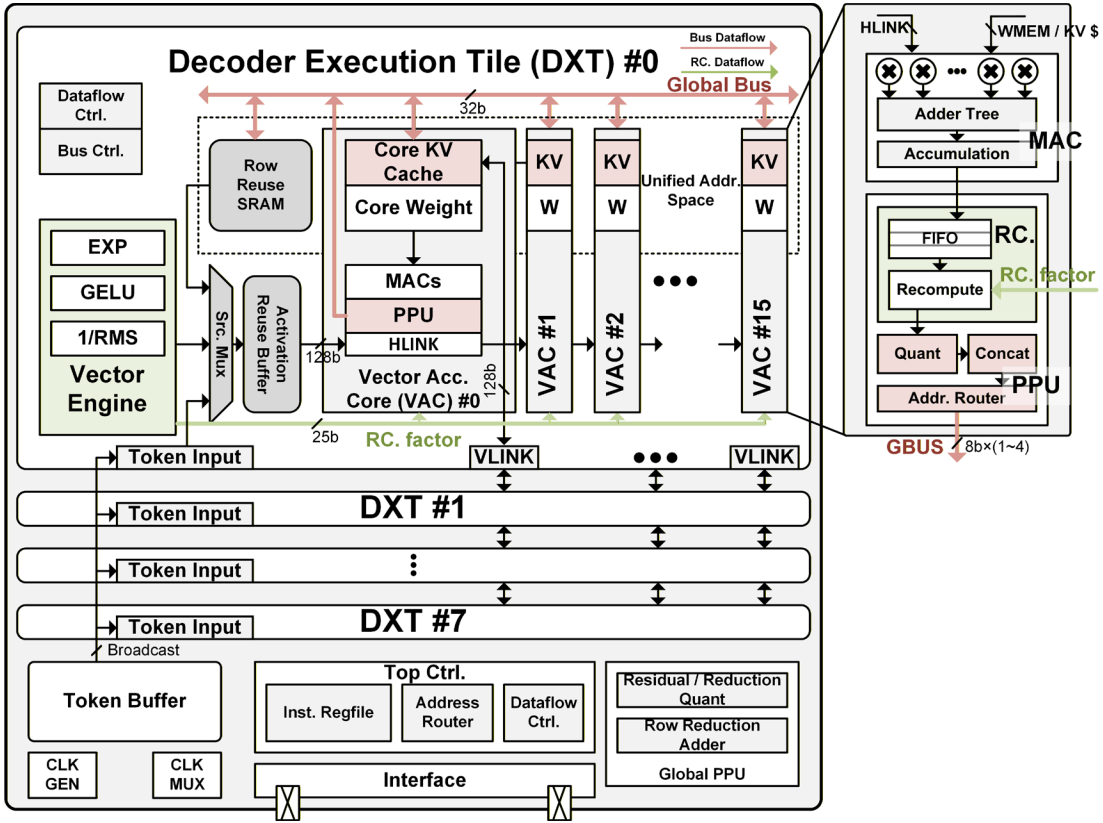
Scalability of DXE



Key Specification

- Single DXE can be extended to an rDXE for larger model support.
- Token buffer receives input tokens from the I/O interface and flexibly delivers them via multicast to all DXTs.

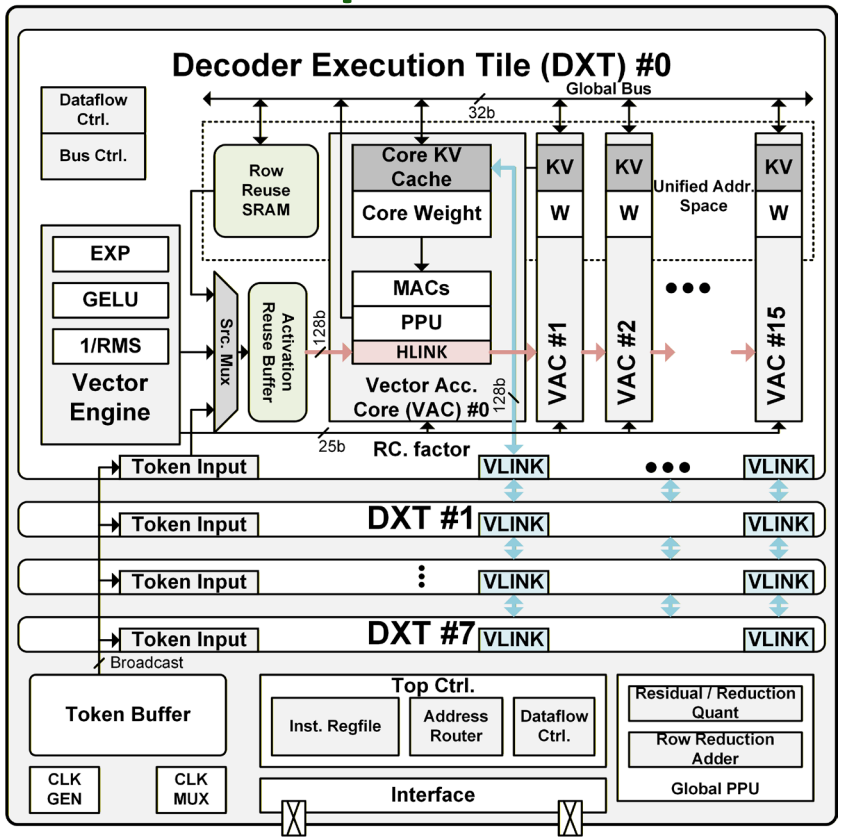
GEMV and Recompute in DXT



Key Specification

- Each DXT integrates 16 vector–accumulate cores (VACs) connected through a 32-bit global bus.
- A dedicated vector engine within each DXT computes the RC factors, which will be broadcast to the recompute unit (RCU) of each VAC.

Data flow optimization in DXE



Key Specifications

- Row-reuse SRAM for reusing recent outputs; Activation-reuse buffer for repeated activation access.
- Horizontal links (HLINK), forming a local systolic input datapath.
- Vertical links (VLINK) for global weight sharing.

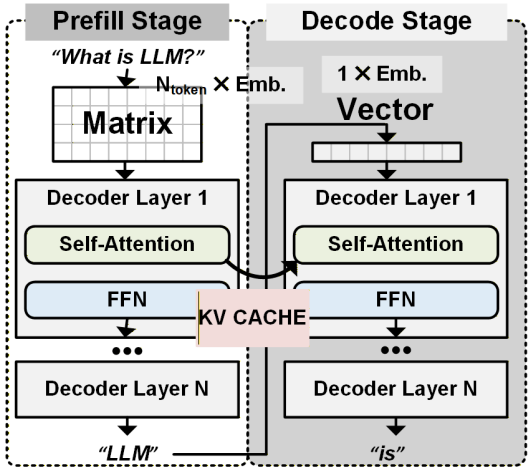
Outline

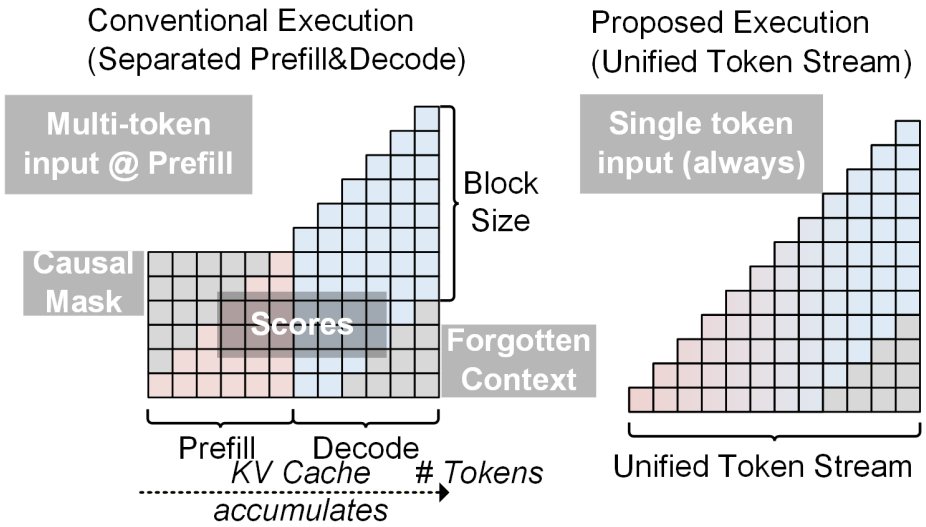
- Introduction
- Overall Architecture
- Key Features
  - Ring-based decoder execution engine (rDXE) with token-level pipelining
  - Vector Recompute (VRC) fusing nonlinear normalization with linear GEMV
  - On-Chip Databuffer Optimizations
- Experiment Results
- Summary

Unified Token Streaming in Prefill & Decode Stage

Key Feature 1

- Each DXE is designed to process a single token, thereby enabling a unified data flow, maintaining high hardware utilization across both stages.

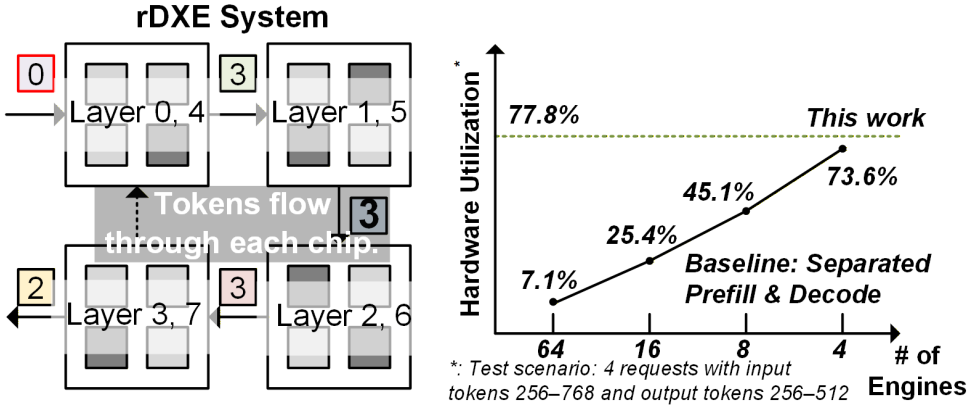




Multi-Requests-Multi-Tokens processing

Key Feature 1

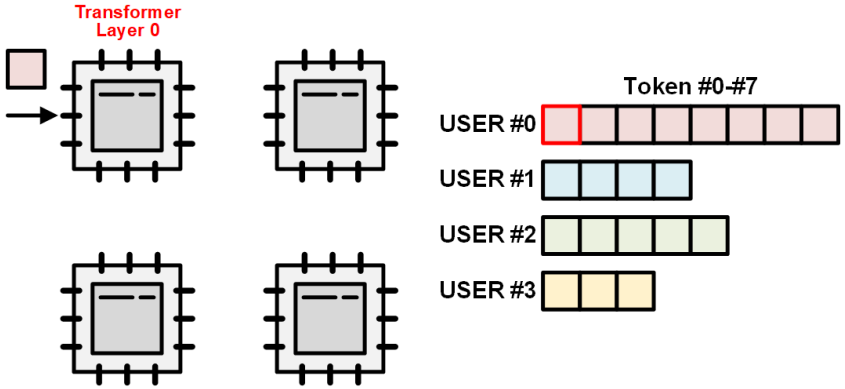
- Each DXE process a single token, thereby enabling a unified data flow, maintaining high hardware utilization.
- 16-channel quad-SPI I/O delivers 800 Mb/s bandwidth; minimum transmission overhead (one processed token).
- Relative to a baseline, rDXE delivers 32.7% and 52.4% higher utilization, respectively.



Multi-Requests-Multi-Tokens processing

Key Feature 1

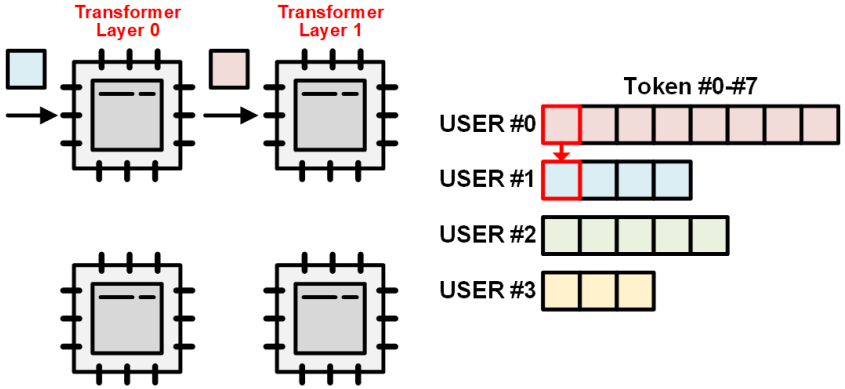
- **Step 1:** Generate the first token for USER0. TPU 0 process the first transformer layer (including QKV generation, Q×K and P×V, and feed-forward network.)



Multi-Requests-Multi-Tokens processing

Key Feature 1

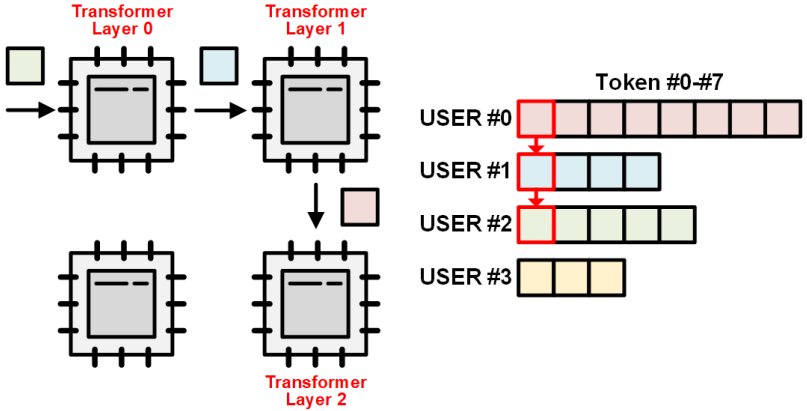
- **Step 2:** Generate the first token for USER1. TPU 0 process the transformer layer #0 for (USER1, Token0). TPU 1 process the transformer layer #1 for (USER0, Token0).



Multi-Requests-Multi-Tokens processing

Key Feature 1

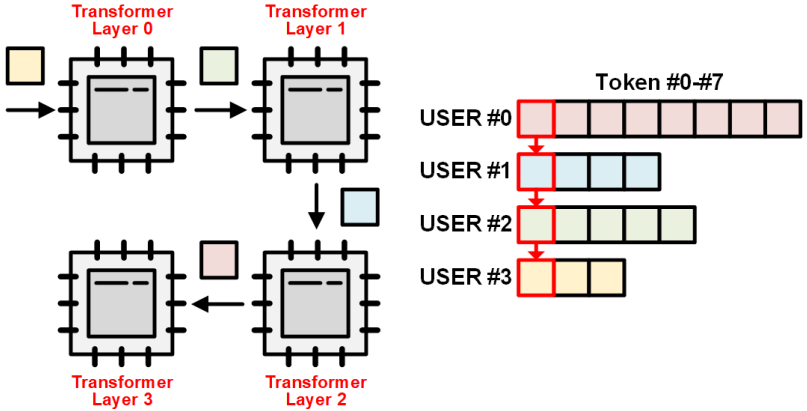
- **Step 3:** Generate the first token for USER2. TPU 0 process the transformer layer #0 for (USER2, Token0). TPU 1 process the transformer layer #1 for (USER1, Token0). TPU 2 process the layer #2 for (USER0, Token0)



Multi-Requests-Multi-Tokens processing

Key Feature 1

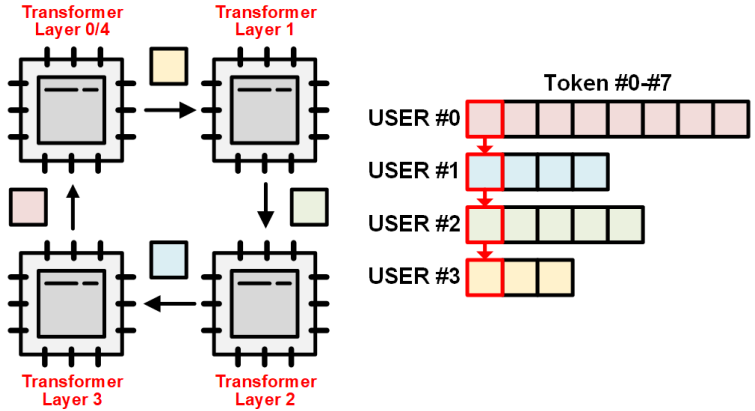
- **Step 4:** Generate the first token for USER3. TPU 0 process the transformer layer #0 for (USER3, Token0). TPU 1 process the transformer layer #1 for (USER2, Token0). TPU 2 process the layer #2 for (USER1, Token0). TPU 3 process the layer #3 for (USER0, Token0).



Multi-Requests-Multi-Tokens processing

Key Feature 1

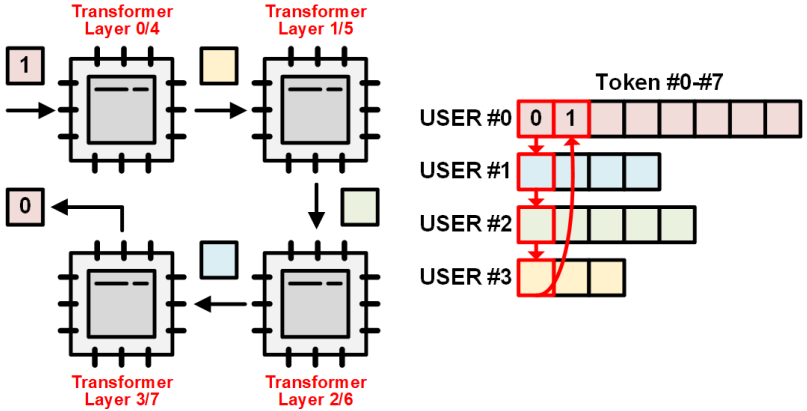
- **Step 5:** Ring Rotation. TPU 0 process the transformer layer #4 for (USER0, Token0). TPU 1 process the transformer layer #1 for (USER3, Token0). TPU 2 process the layer #2 for (USER2, Token0). TPU 3 process the layer #3 for (USER1, Token0).



Multi-Requests-Multi-Tokens processing

Key Feature 1

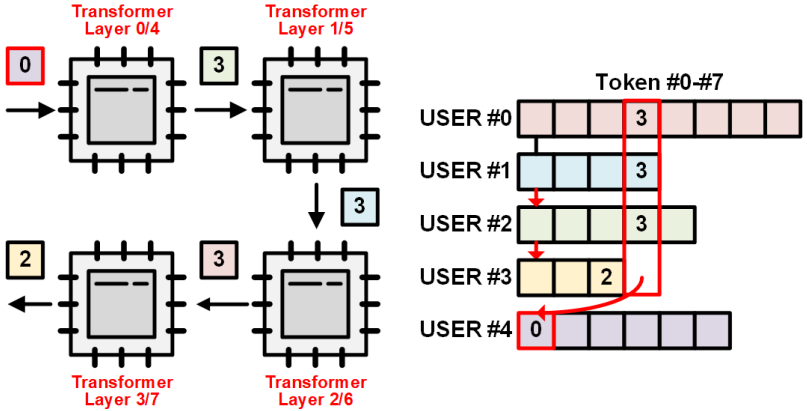
- **Step 6:** Finish the inference for (USER0, Token0). Feed (USER0, Token1) to TPU 0. This way won't stop the pipeline. In other words, it won't lead to pipeline bubble.



Multi-Requests-Multi-Tokens processing

Key Feature 1

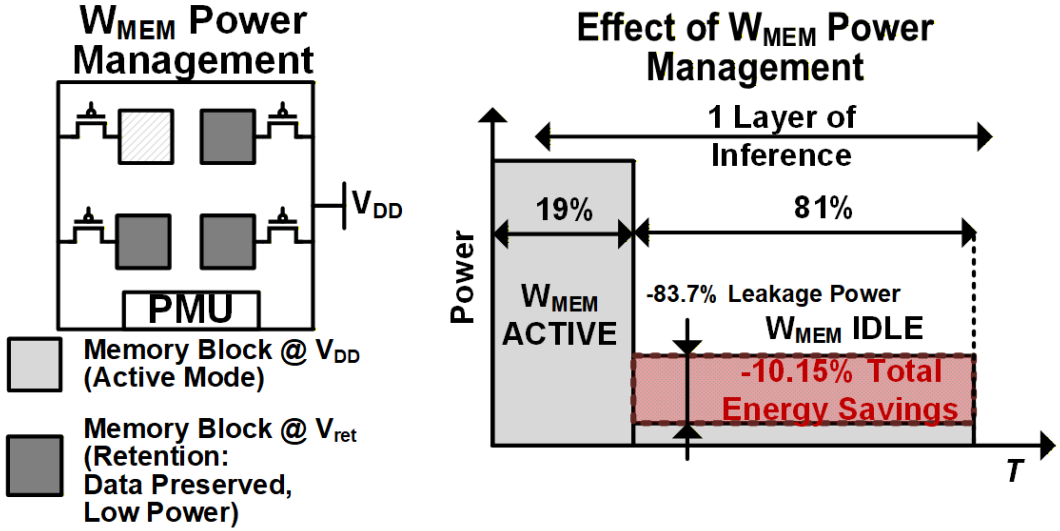
- **Step 7:** Finish the inference for all tokens from USER3. Feed (USER4, Token0) (A New USER) to TPU 0 and erase the KV cache for USER3. This way won't stop the pipeline. Additionally, it avoids padding. The tokens are processed in a coarse-grained layer-pipeline manner.



On-chip Memory Power Management

Key Feature 1

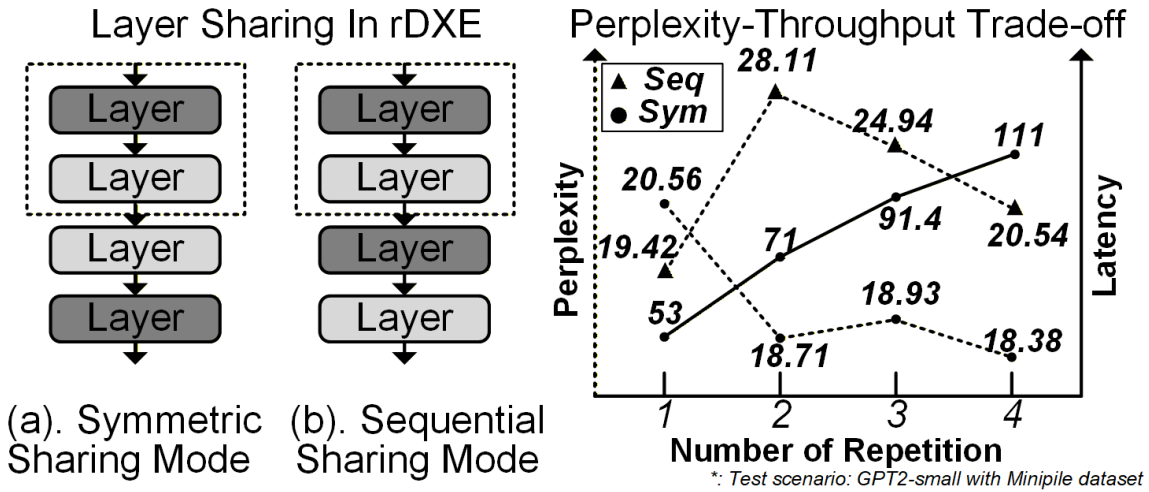
- Weight Memory is partitioned into banks, with only the necessary banks accessed at any given time.
- Power gating is applied to non-active banks to further reduce leakage energy, resulting in an overall DXE energy reduction of 10.15%.



Layer Sharing in rDXE

Key Feature 1

- DXEs enables flexible layer-sharing patterns, the ring topology keeps weights stationary and circulates tokens.
- Supports symmetric and sequential sharing, reducing perplexity by 25% without increasing on-chip memory.



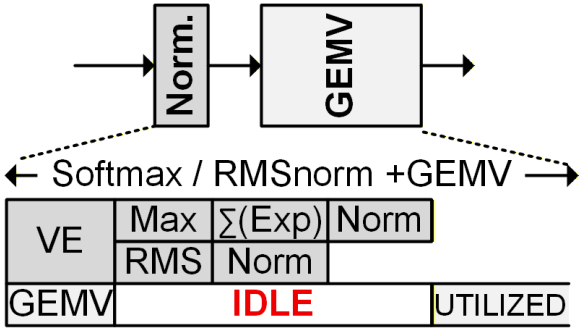
Outline

- Introduction
- Overall Architecture
- Key Features
  - Ring-based decoder execution engine (rDXE) with token-level pipelining
  - Vector recompute (VRC) fusing nonlinear normalization with linear GEMV
  - On-chip data buffer optimizations
- Experiment Results
- Summary

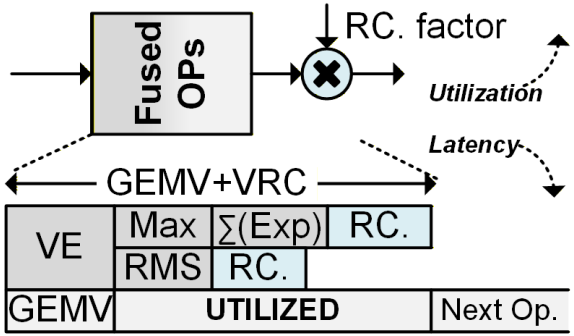
VRC fusing nonlinear normalization with linear GEMV

Key Feature 2

- Nonlinear stages (Softmax/RMSnorm) lowers MAC utilization, and increases overall latency.
- Polynomial approximations to nonlinear stages will degrade accuracy and additional rollback mechanism increases latency.
- Fuses nonlinear stages directly into the GEMV pipeline without compromising precision.



Conventional: Reduced PE utilization

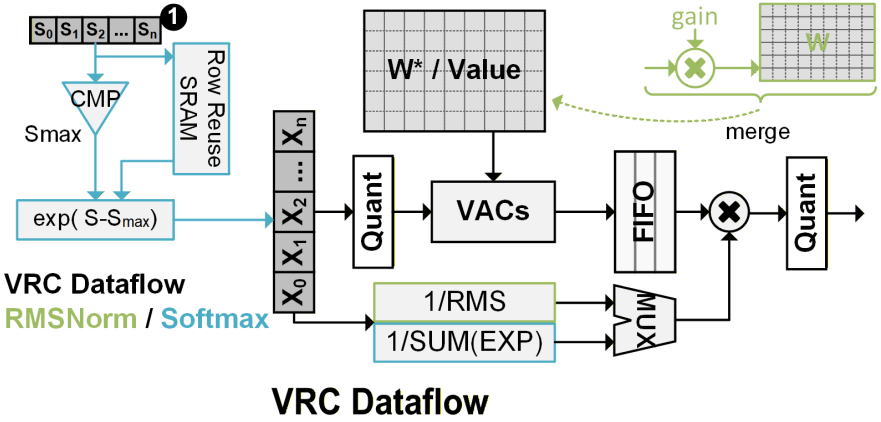


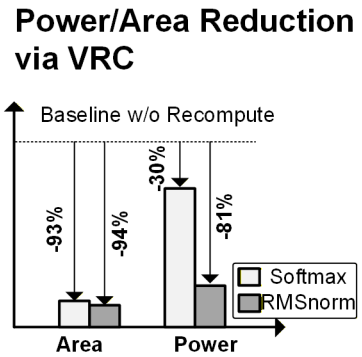
Proposed: Pipelined Vector Recompute Engine

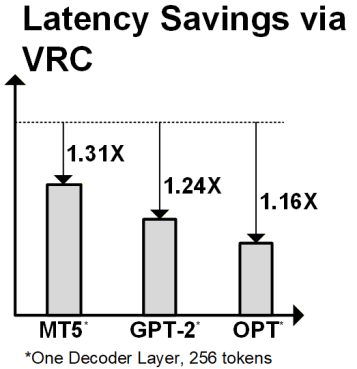
VRC fusing nonlinear normalization with linear GEMV

Key Feature 2

- Input token is simultaneously fed to both the MAC unit and the vector engine (VE).
- VE computes the reciprocal RMS and reciprocal exponential sum.
- Latency of VE hidden under linear MAC unit.
- Area reduction: Softmax 93%, RMSNorm 94%; Power reduction: Softmax 30%, RMSNorm 81%
- Up to 1.31× end-to-end latency improvement.



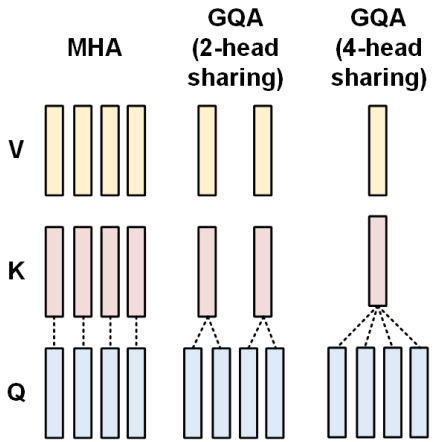


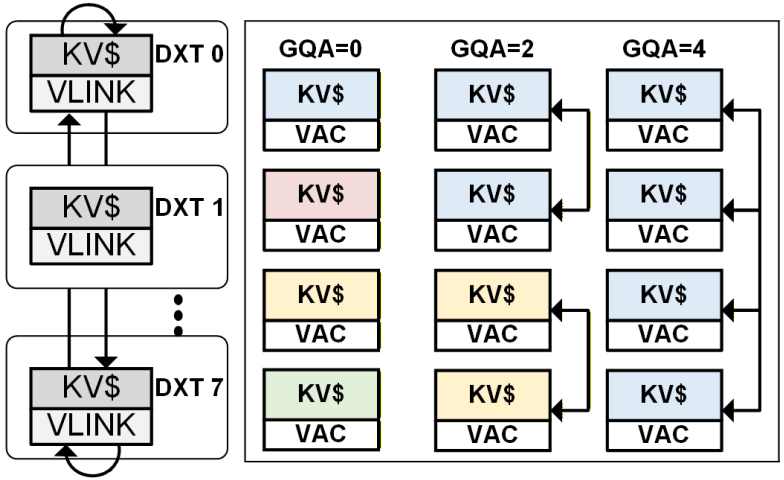


Group Query Attention (GQA) Hardware Support

Key Feature 3

- Splits queries into groups to share attention computation, boosting efficiency.
- Reduces computation and memory overhead, enabling scalable transformer models .

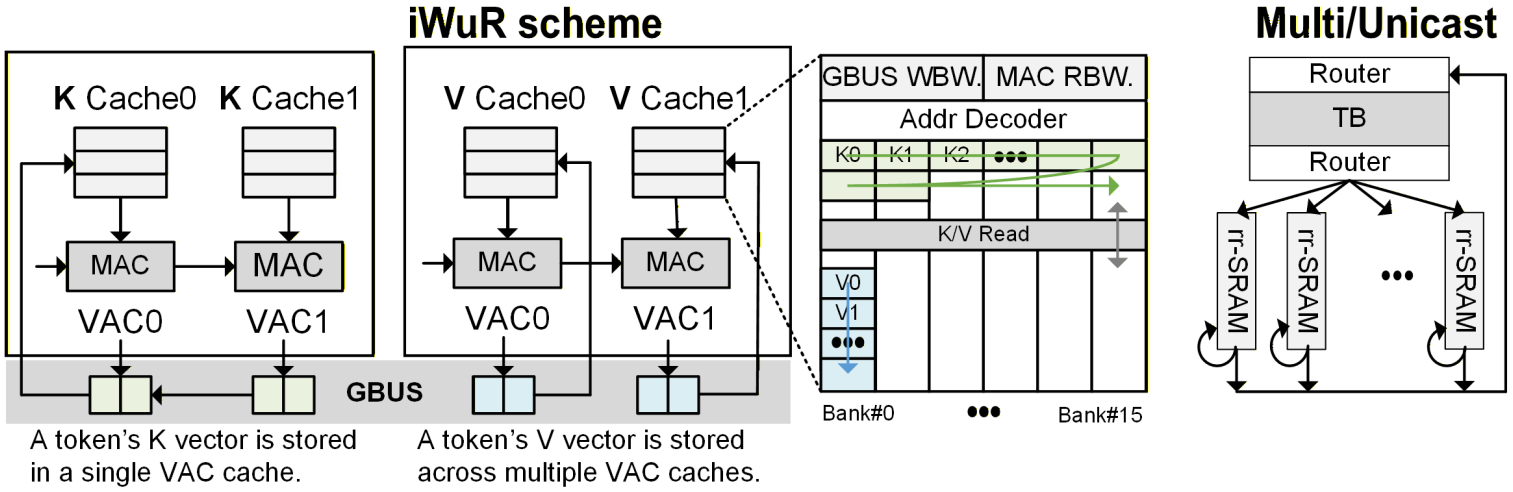




On-Chip Databuffer Optimization

Key Feature 3

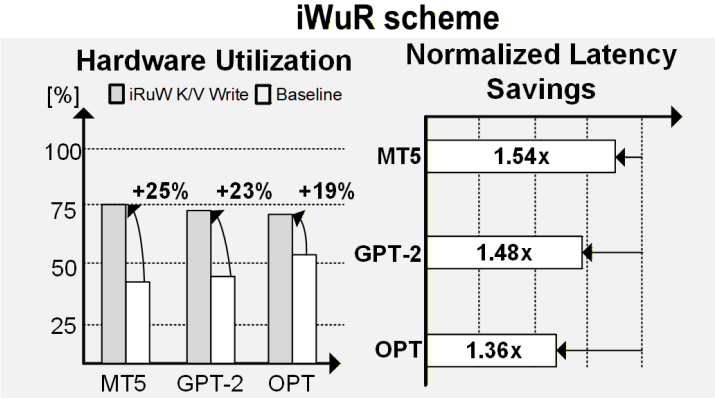
- GBUS and VAC caches support the irregular-write-uniform-read (iWuR) scheme for efficient matrix transposition.
- Flexible broadcast/unicast modes (TB, rr-SRAM) adapt to attention and projection layers.
- Increases data reuse and lowers inter-buffer traffic throughout attention and MLP layers.

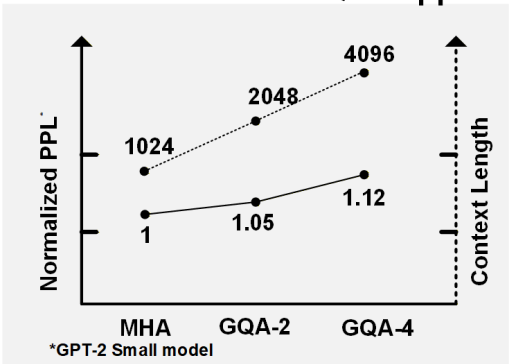


Performance Improvement

Key Feature 3

- iWuR scheme boosts utilization by up to 25% and cuts latency 1.36×–1.54×.
- GQA enabled by VLINK results in up to 83% buffer savings, minimal accuracy loss.





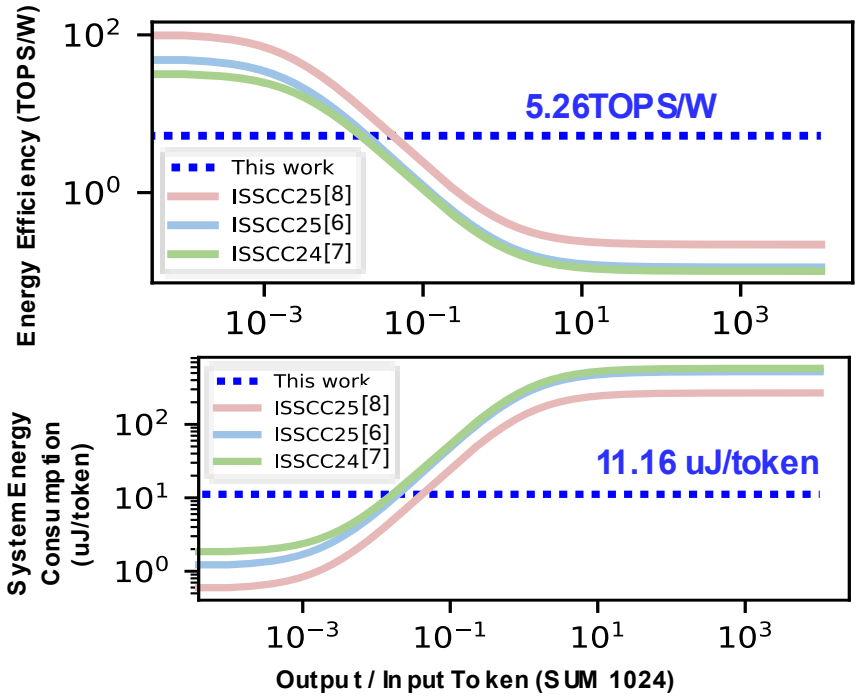
Outline

- Introduction
- Overall Architecture
- Key Features
  - Ring-based decoder execution engine (rDXE) with token-level pipelining
  - Vector recompute (VRC) fusing nonlinear normalization with linear GEMV
  - On-chip data buffer optimizations
- Experiment Results
- Summary

Benchmark Analysis

	Specifications			
Technology	16nm			
Die Area	14.21 mm^2			
Voltage	0.56V-1.0 V			
Frequency	12.5 MHz-200 MHz			
SRAM	4MB			
Data Precision	VAC: INT8 VE:BF16;			
Power	18.95 mW - 281.01 mW			
Peak Performance	1.63 TOPS **(1)**			
Peak Energy Efficiency	6.47 TOPS/W **(2)**			
Model Type	MT5	Mobile LLM	GPT-2	GPT-2
Dataset	Minipile	PIQA	Wikitext	Wikitext
Parameters	350M	125M	125M	125M
Accuracy	39.83 (BLEU)	60 (ACC)	18.38 (PPL)	20.56 (PPL)
Layer Sharing	1	4	1	4
Required #DXEs in rDXE	2	8	27	7
Inference Throughput (Token/s) **(4)**	5103 **(3)**	3921	14577	4412

(1) @ 1.0 V, 200MHz (2) @ 0.67V, 25MHz (3) Only the decoder of MT5 is evaluated

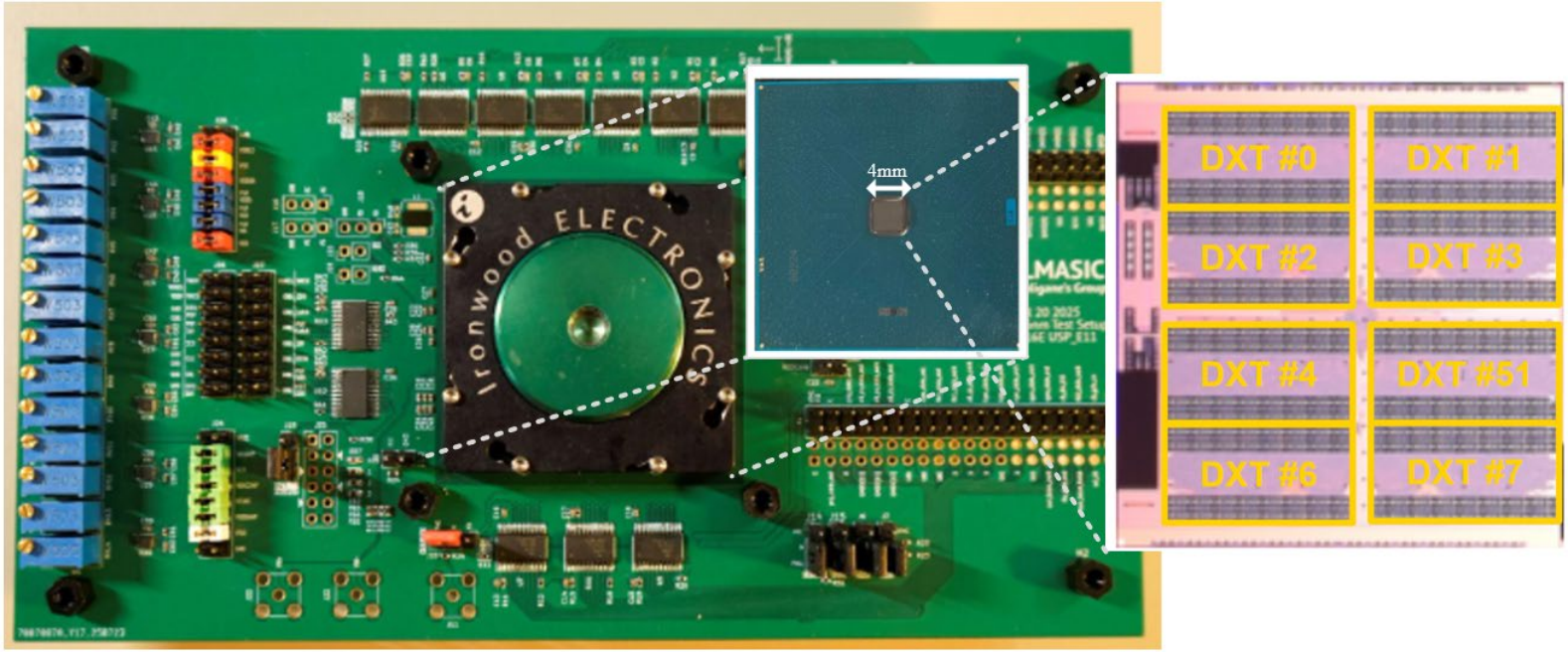


Comparison Table

	This Work	VLSI25 [9]	JSSC25 [11]	ISSCC25 [8]	ISSCC24 [7]
LLM Acceleration	**Yes**	Yes	No	Yes	Yes
Multi-Requests Support	**Yes**	Yes	No	No	No
Edge Application	**Yes**	No	Yes	No	No
Tech. (nm)	**16**	4	40	28	28
Supply Voltage (V)	**0.56-1.0**	0.65-0.9	0.9	0.58-1.0	0.7-1.1
Frequency (MHz)	**12.5-200**	25-1000	50-100	25-200	50-200
Die Area (mm2)	**14.21**	5.28	65.6	20.25	20.25
Precision	**INT8**	FP16	Posit8	A:INT4/8/16 W:INT1-16/1.58	INT8
On-Chip Memory (kB)	**4096**	2160	12000	500	500
Power (mW)	**18.95-281.01**	11-735	N/A	4.69-82.07	47.6-469.2
Performance(1) (TOPS or TFLOPS)	**0.81-1.63**	8.19	0.0256-0.0512	4.92-13.1	3.41
Energy Efficiency(2) (TOPS/W or TFLOPS/W)	**2.51-6.47**	25.3	0.43-0.50	77.6-189.8	22.9-47.8
Max EMA Reduction Percentage (%)	**100%**	N/A	100%	80.25%	73%
Normalized Energy Consumption(2)(3) (uJ/token)	**11.16**	273.89	119.53	17.28	37.89
Normalized Energy Efficiency(2)(3) (TOPS/W or TFLOPS/W)	**5.26**	0.21	0.49	3.40	1.55

(1)Numbers at the fastest voltage and frequency. (2) Numbers at the most energy-efficient voltage and frequency. (3) Estimated with LPDDR3 SDRAM and including EMA of 85pJ/B [1], the MT5 model was evaluated with 960 input tokens and 64 output tokens.

Die Photos, and Testing System



Outline

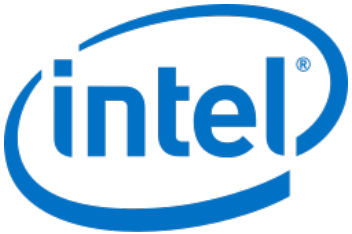
- Introduction
- Overall Architecture
- Key Features
  - Ring-based decoder execution engine (rDXE) with token-level pipelining
  - Vector recompute (VRC) fusing nonlinear normalization with linear GEMV
  - On-chip data buffer optimizations
- Experiment Results
- Summary

Summary

- Token-level pipelining enables seamless layer-wise processing and high utilization for multiple concurrent requests across prefill and generation.
- Vector Recompute (VRC):
  - Hardware fusion of Softmax/RMSNorm with MAC operations saves up to 1.31× latency.
  - Minimizes required feature storage and sharing compute resources with MAC unit enables power and area savings.
- HW/SW co-design features of on-chip memory optimization:
  - VLINK enables group query attention reduce KV$ capacity demand.
  - Irregular write uniform read scheme boosts KV$ bandwidth usage.
  - Multi-cast and unicast adaptive dataflow increases data reuse and minimizes memory traffic.

Acknowledgements

We thank Intel Corp. for the silicon donation and Google Research for support!



Thanks!