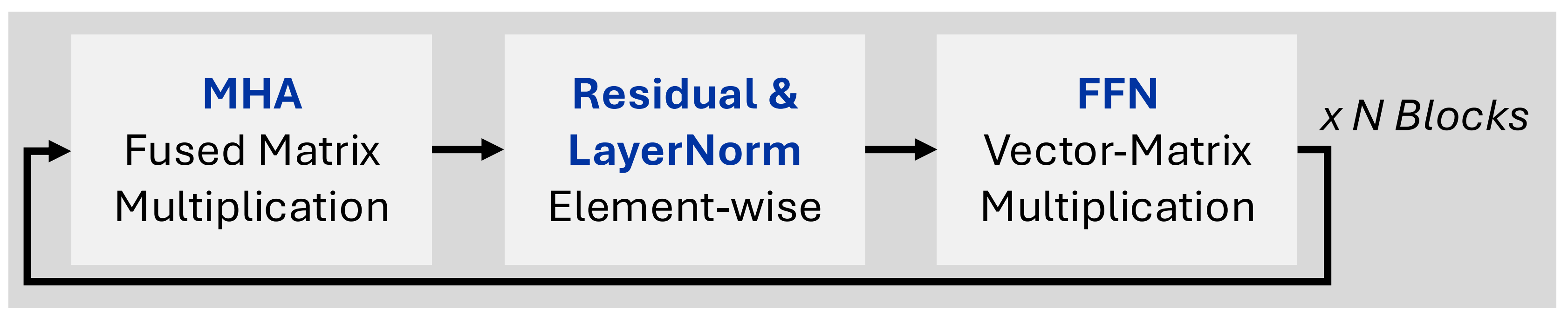
Design- and Run-Time Reconfigurable System-on-Chip Architectures for Language Model Inference on the Edge

Je Yang, Gabriele Tombesi, Joseph Zuckerman, and Luca P. Carloni

sld.cs.columbia.edu
esp.cs.columbia.edu

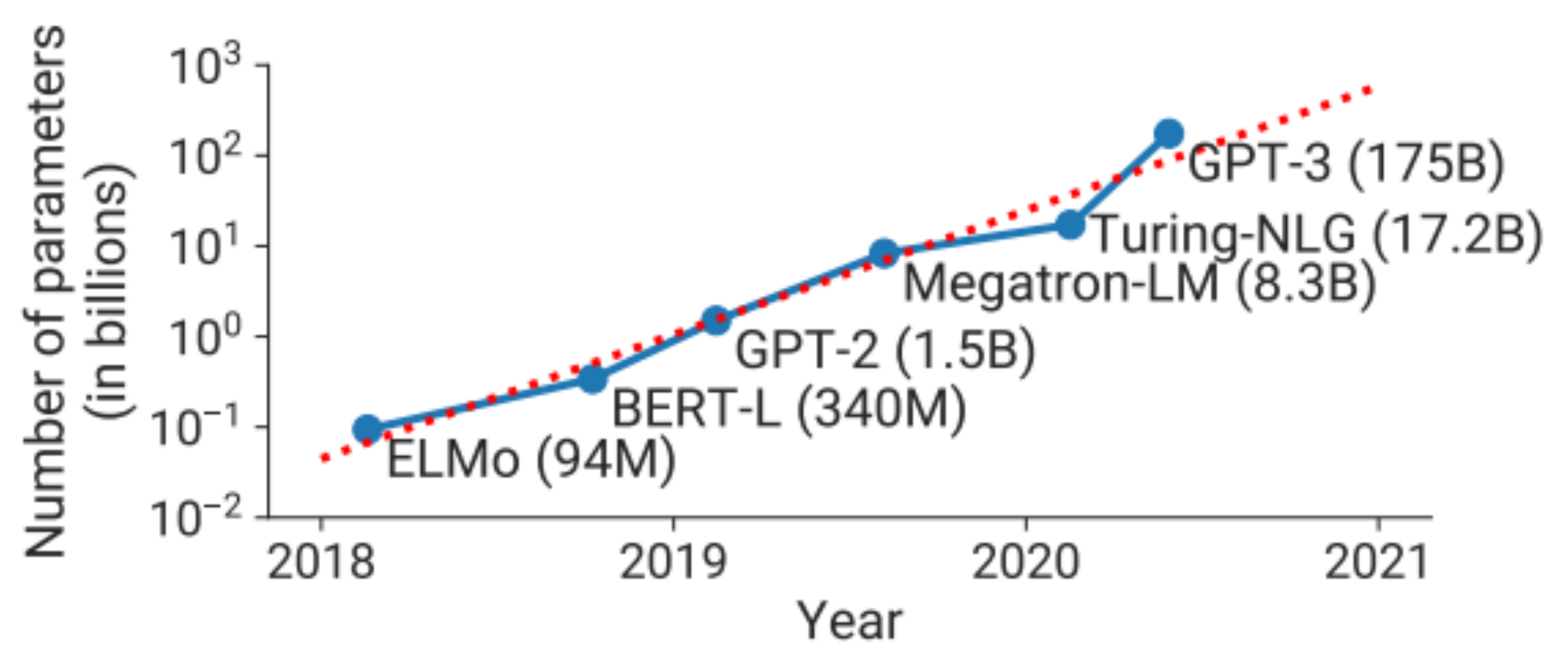
Introduction

Recent advancements in neural network design have emphasized **attention-based Transformer models**, which deliver state-of-the-art accuracy across applications in natural language processing and computer vision.
The encoder of Transformer performs a multi-head attention (MHA) module and a feed-forward network (FFN) module, each followed by a layer normalization operation and a residual connection.



SystemC Design for Fast-Growing Model

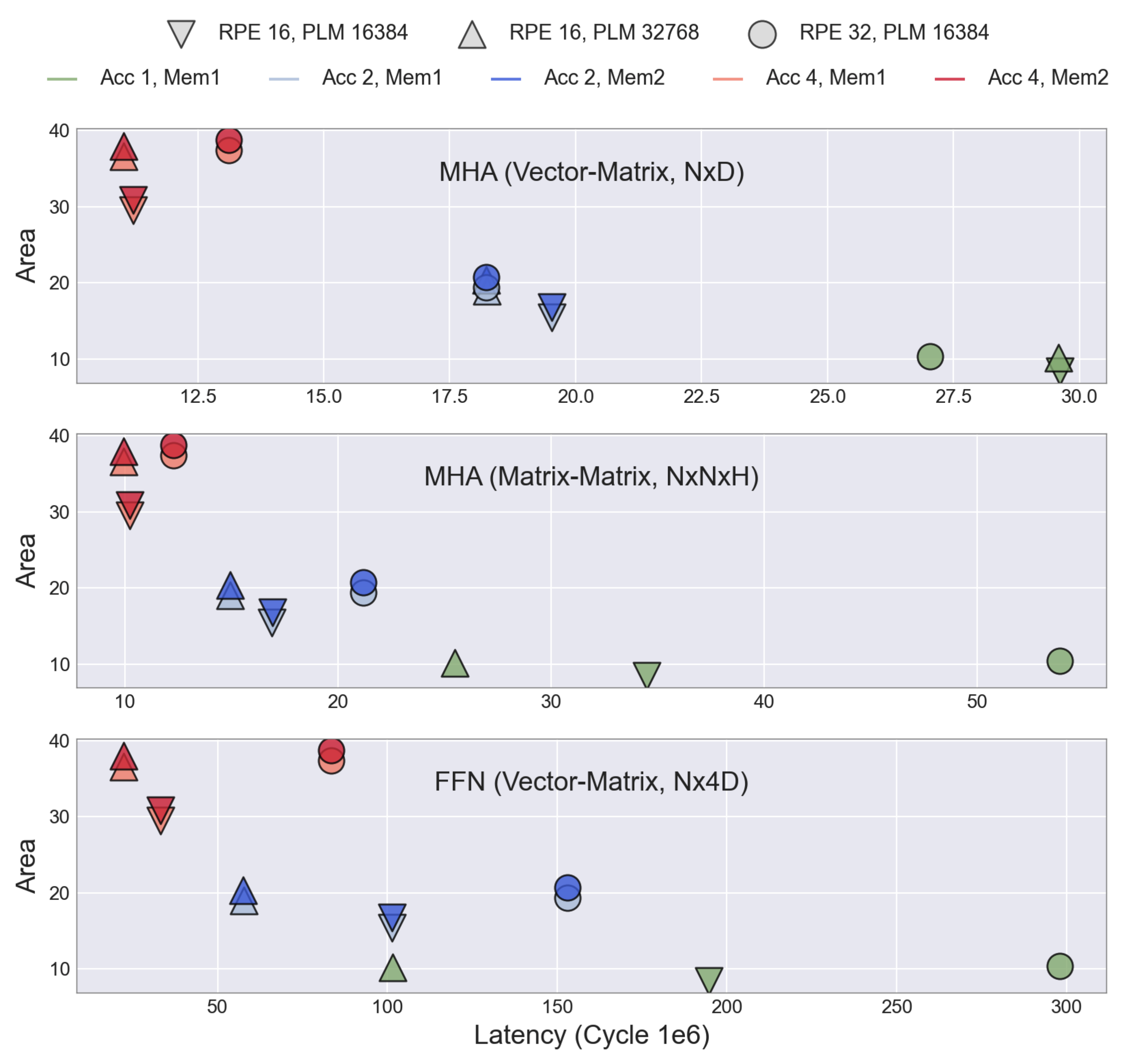
- Breakthrough performance of Transformer-based model
  - Market valued 4.1B$ in 2023 surge to 53.9B by 2032
  - Urgency for efficient and scalable hardware solutions
- **High-level-Synthesis-based Design to accelerate time-to-market**
  - Reduce simulation time by nearly an order of magnitude compared to traditional VHDL simulations



** Efficient Large-Scale Language Model Training on GPU Clusters using Megatron-LM*

Design-Time Reconfiguration for Resources

- **Layer-wise heterogeneity in Transformers** demand large computation and memory resources
- **Design-time reconfigurability enables optimal system choice on computing parallelism (RPE) and on-chip memory capacity (PLM)**

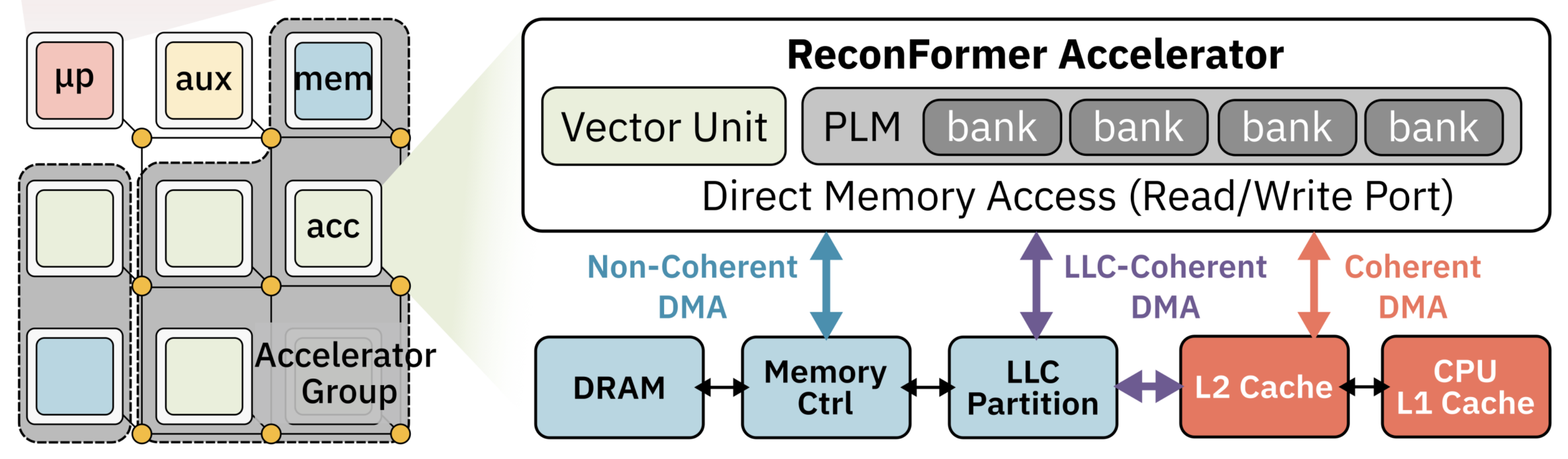


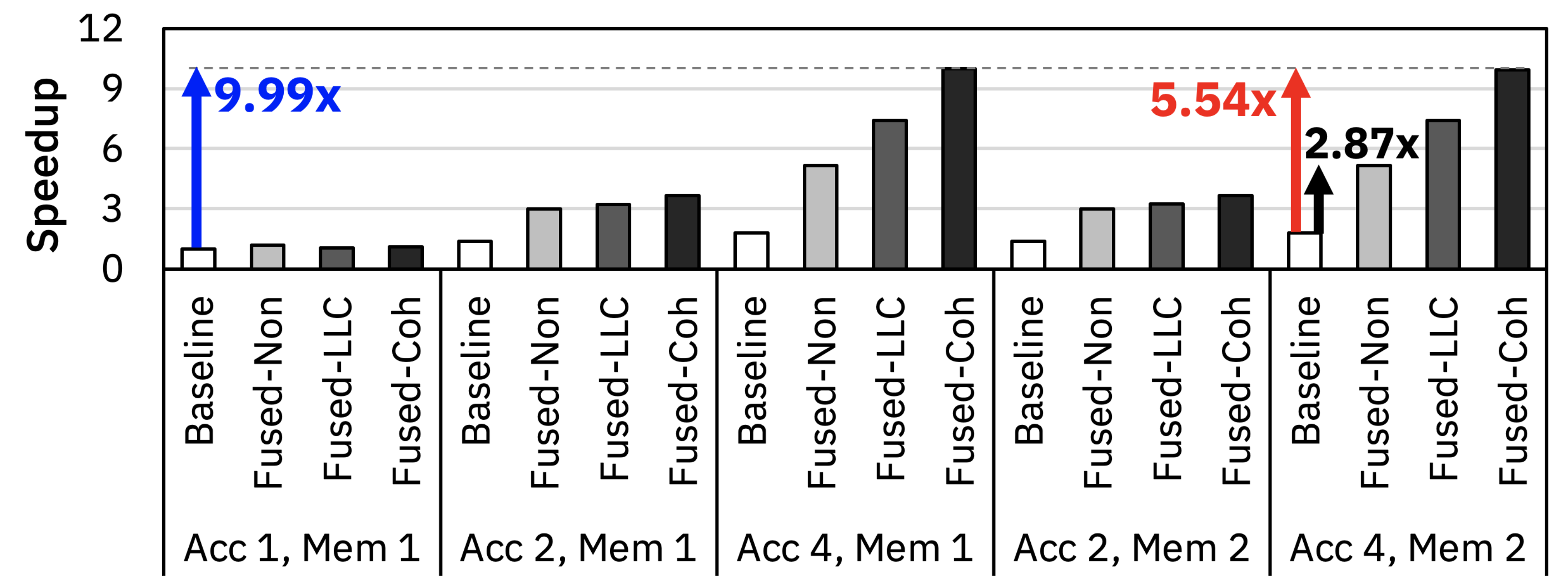
Run-Time Reconfigurable System Strategies

On top of optimal accelerator design, we incorporate adaptive strategies for **layer-wise cache coherence and operation fusion**.
We leverage the cache hierarchy to reduce energy and latency associated with external memory accesses. This dynamic configuration selection yields a **5.54x speedup in multi-head attention and a 3.61x end-to-end performance improvement** over static systems, even in long-sequence scenarios.

```
for kernel in attention {
  // Head Parallelism
  write(acc, coherent);
  run(acc, N_ACC);
}
...
for kernel in FFN {
  // Sequence Parallelism
  write(acc, non_coherent);
  run(acc, N_ACC);
}
```

Architecture-level Parameter	
Parallelism Degree (RPEs)	16, 32
PLM Depth	(16384, 32768)/RPEs
Operation Type	Vector/matrix-multiplication, Fused matrix multiplication, Element-wise operation
System-level Parameter	
Parallelism Degree (Accelerator Tiles)	1, 2, 4, 8
Parallelism Strategy	Sequence, Head
Memory Tiles	1, 2
Cache Coherence	Non-, LLC-, Coherent-





System Performance

	GPU	Edge CPU	Server CPU	Our Work
Compute Unit	1GHz 9,216 Cores	1.7GHz Single Core	3GHz 24 Cores	100MHz 4x16 RPEs
On-chip Memory	15MB	16KKB	38MB	4 x1.3MB
Off-chip Bandwidth	600GB/s GDDR6	22.5GB/s DDR4	89.6GB/s DDR4	22.5GB/s DDR4
Power (W)	147.47	0.254	52.25	14.23
Throughput (seq/s)	B: 3,097	B: 0.017	B: 7.42	B: 12.40
	L: 1,130	L: 0.001	L: 4.18	L: 8.16
Efficiency (seq/s/W)	B: 21	B: 0.067	B: 0.016	B: 0.87
	L: 8	L: 0.038	L: 0.08	L: 0.57

***B** refers to BERT-Base (110M) and **L** refers to BERT-Large (340M)*

Conclusion & Future Work

We accelerated BERT by integrating coarse-grained design-time and run-time reconfigurability, including adaptive strategies for intra- and inter-accelerator parallelism and dynamic coherence-mode selection.
Our system achieves **730.18x and 1.67x higher throughput and 13.03x and 5.30x higher efficiency** compared to edge and server CPUs, respectively.
We plan to extend our approach with diverse workload like **generation tasks** (e.g., GPT) with varying sequence length.