TECHNICAL DOSSIER // DEEP LEARNING SYSTEMS & GPU ARCHITECTURE

Sub-Quadratic Linear Attention Kernel Microarchitecture

Chunkwise Parallel State Recurrence, Triton SRAM Tiling, and Long-Context IOPS

Published: October 30, 2026 | Distribution: Open Engineering Community | Document ID: LT-2026-B16-03

EXECUTIVE SUMMARY & MICROARCHITECTURAL SCOPE

While FlashAttention-3 maximizes Tensor Core utilization on Hopper H100 GPUs via asynchronous warp specialization and TMA hardware copies, it remains bound to O(N^2) quadratic memory and compute complexity. FlashLinearAttention (FLA) implements hardware-aligned, chunkwise parallel formulations that decompose linear attention into intra-chunk parallel GEMMs and inter-chunk associative recurrent state passing, achieving strict O(N) linear time and constant O(1) inference memory footprint.

Kernel / Mechanism	Standard Attention	FlashAttention-2	FlashAttention-3	FlashLinearAttention
Time Complexity	O(N^2 * d)	O(N^2 * d)	O(N^2 * d)	O(N * d^2) [Linear]
Decode Memory	O(N * d) [KV Cache]	O(N * d)	O(N * d)	O(1) [Fixed Matrix State]
Arithmetic Intensity	Memory Bound	Medium (Tiled)	High (Asynch TMA)	Very High (Fused SRAM)
128k Context Speed	OOM	Base (1.0x)	2.4x Speedup	6.2x Speedup
Target GPU Core	Generic CUDA	SRAM Shared Mem	Hopper TMA / WGMMA	Triton / Tensor Cores

1. The Quadratic Softmax Barrier

Standard transformer self-attention computes Softmax(QK^T)V, requiring computing and storing an N x N attention matrix. Even with fused online softmax tiling, the required compute FLOPs scale quadratically with sequence length N. In long-context tasks exceeding 64k tokens, memory bandwidth between HBM and on-chip SRAM becomes the dominant execution bottleneck.

2. Linear Kernel Decomposition & Chunkwise Recurrence

Linear attention replaces the non-linear softmax kernel with an associative feature mapping, allowing matrix multiplication reordering: (Q * K^T) * V = Q * (K^T * V). To achieve hardware parallelism on systolic Tensor Cores, FlashLinearAttention adopts a chunkwise parallel formulation where tokens are grouped into chunks of size C = 64 or 128. Intra-chunk attention executes via standard causal GEMMs, while inter-chunk state transfers via associative recurrent scans.

3. Hardware Tiling & Triton Kernel Engineering

FLA leverages OpenAI Triton to map recurrent state updates directly into GPU Streaming Multiprocessor (SM) SRAM and register files. By fusing the recurrent key-value matrix accumulation S_t = lambda * S_{t-1} + K_t^T * V_t into on-chip SRAM, FLA eliminates intermediate HBM round-trips, saturating modern GPU Tensor Core execution pipelines.

4. Constant-Memory Autoregressive Decoding

During inference generation, standard transformers require storing past key-value vectors in an ever-expanding KV cache that consumes tens of gigabytes of VRAM. FlashLinearAttention maintains a fixed-size d x d recurrent state matrix regardless of sequence length, enabling 1M+ token context generation with zero KV cache growth.

STRATEGIC ENGINEERING TAKEAWAYS & DEPLOYMENT CHECKLIST

- **Sub-Quadratic Complexity:** Reduces sequence scaling from O(N^2) to O(N), enabling million-token processing.
- **Chunkwise Parallelism:** Blends systolic matrix-matrix multiplication speed with associative recurrent state updates.
- **SRAM Fused Kernels:** Keeps recurrent state S_t inside on-chip registers, maximizing memory arithmetic intensity.
- **Zero KV Cache Explosion:** Decodes autoregressively with an invariant O(1) state footprint, drastically cutting inference VRAM.