BERNSTEIN
SOCIETE GENERALE GROUP

27 October 2026

U.S. IT Hardware

Global Memory & Storage: HBF Deep Dive - Could HBF be the solution to the Memory bottleneck?


Mark C. Newman
+1 212 845 7822
mark.newman@bernsteinsg.com


Mark Li
+852 2123 2645
mark.li@bernsteinsg.com


Phoebe Sun
+1 917 344 8481
phoebe.sun@bernsteinsg.com


Viola Chen
+1 917 344 8614
viola.chen@bernsteinsg.com


Edward Hou, CFA
+852 2123 2623
edward.hou@bernsteinsg.com


Yipin Cai, CFA
+852 2123 2669
yipin.cai@bernsteinsg.com

The Memory Wall and the shift to Memory-centric AI: AI's evolution from training to inference and agentic workloads is making memory capacity, bandwidth, and data reuse increasingly important. Larger models and expanding context push data beyond HBM into the broader memory and storage hierarchy, while caching creates an economic incentive to retain rather than recompute it. This expands memory's role in infrastructure, broadening demand beyond just HBM into conventional DRAM, NAND, and HDD.

HBF could make large-scale AI inference more memory-efficient and cost-effective. HBF could lower the cost of AI inference by combining HBM-like read bandwidth with 8-16x greater capacity at a meaningfully lower cost per GB. In a tiered architecture, HBF could make AI inference cheaper by 1) reducing the GPU and HBM capacity required to host large models and 2) avoiding costly recomputation by retrieving previously processed context. HBF's value proposition is therefore not simply cheaper memory, but a potentially ***more capital- and compute-efficient AI architecture***.

High performance NAND (such as HBF) as new memory tiers will support NAND industry tightness and pricing. Multiple technologies are emerging to add new memory tiers between HBM and conventional NAND, with each addressing different AI requirements. HBF alternatives focus on low-latency flash access, pooled KV-cache retention, and other specialized use cases. As most of these technologies use NAND and often low density NAND with high trade ratios, their adoption should ***consume significant NAND wafer supply and thus tighten NAND supply/demand balance*** regardless of which architecture gains the most share.

HBF economics: Huge trade ratio, positive for NAND overall. Due to the higher performance SLC flash used and additional silicon area needed for TSV, high-bandwidth I/O and related peripheral circuitry, we estimate ***HBF trade ratio at 6.5x***, much higher than the roughly 3x we have seen for HBM. We estimate the***cost ratio of HBF at roughly 10x*** depending on the configuration. By pricing HBF similar to HBM on a per-module basis (with 8-16x more capacity and similar read bandwidth) we estimate that HBF can get similar gross margins of NAND flash therefore being hugely accretive to profits.

HBF could do for NAND what HBM did for DRAM industry dynamics. HBF has the potential to create a meaningful new source of NAND demand while simultaneously reducing the number of gigabytes produced per wafer. This combination could materially improve the industry's supply-demand balance. While HBF could ***reduce GPU intensity and HBM requirements per workload, HBF is more likely to complement than replace HBM*** and with higher efficiency actually drive more demand (Jevons Paradox).

HBF holds huge promise for Sandisk and the entire NAND industry but broader adoption challenges remain. Our biggest concern on HBF is the lack of endorsement from NVIDIA, etc. Google, Meta, and SK Hynix are in the HBF consortium, but NVIDIA, KIOXIA & others are advocating Storage Next & CMX that also alleviates the burden of KV-cache on HBM.

First Published: 27 Oct 2026 20:01 UTC Completion Date: 27 Oct 2026 17:16 UTC

BERNSTEIN TICKER TABLE

Ticker	Rating	Cur	26 Oct 2026 Closing Price	Price Target	TTM Rel. Perf.	Cur	Reported EPS 2025A	2026E	2027E	Reported P/E (x) 2025A	2026E	2027E
SNDK (SanDisk)	O	USD	1,255.58	3,000.00	625.5%	USD	70.08	238.36	272.09	17.9	5.3	4.6
STX (Seagate)	O	USD	644.21	1,350.00	182.5%	USD	15.53	40.61	64.40	41.5	15.9	10.0
WDC (Western Digital)	O	USD	356.13	600.00	191.5%	USD	10.19	27.30	45.26	34.9	13.0	7.9
005930.KS (SEC- Samsung)	O	KRW	222,000	440,000	115.2%	KRW	6,611.53	45,072	74,602	33.6	4.9	3.0
005935.KS (SEC-Pref - Samsung)	O	KRW	171,100	374,000	112.3%	KRW	6,611.53	45,072	74,602	25.9	3.8	2.3
SMSN.LI (Samsung)	O	USD	4,110.00	7,857.00	133.2%	USD	116.15	781.29	1,332.17	35.4	5.3	3.1
000660.KS (SK hynix)	O	KRW	1,341,000	2,700,000	150.5%	KRW	60,341	353,627	554,848	22.2	3.8	2.4
MU (Micron)	O	USD	903.70	1,300.00	310.6%	USD	8.29	73.78	173.33	109.1	12.2	5.2
285A.JP (KIOXIA)	U	JPY	12,605	49,000	295.2%	JPY	1,014.00	10,485	14,140	12.4	1.2	0.9
SPX			7,676.77									
ASIAX			1,783.55									
EM			1,660.97									

O - Outperform, M - Market-Perform, U - Underperform, NR - Not Rated, CS - Coverage Suspended

SNDK, STX, WDC, MU estimate is Adjusted EPS; SNDK, STX, WDC, MU valuation is Adjusted P/E (x); SNDK, STX, WDC base year is 2026;

Source: Bloomberg, Bernstein estimates and analysis.

INVESTMENT IMPLICATIONS

Sandisk: We rate SNDK Outperform with TP of $3000.

Seagate: We rate STX Outperform with TP of $1350.

Western Digital: We rate WDC Outperform with TP of $600.

Samsung Electronics: We rate Samsung Electronics Outperform with price target of KRW 440,000.

SK hynix : We rate SK hynix Outperform with price target of KRW 2,700,000.

Micron: We rate Micron Outperform with price target of US$1,300.00.

KIOXIA: We rate KIOXIA Underperform with price target of JPY 49,000.

Table Of Contents

Part I. The Memory Wall - AI Is Shifting From Compute-Centric to Memory-Centric....3
Part II: The Solution to the Memory Wall - HBF could make AI more memory-efficient and cost-effective....6
Part III: High performance NAND such as HBF as new memory tier to help NAND Supply/Demand Balance and Pricing....9
Part IV: HBF Economic Analysis - High cost & trade ratio but Accretive to Revenue & Profit....13
Part V: HBF's impact on the NAND industry as well as GPU & HBM demand....19
Part VI: HBF Technology Deep Dive: How HBF turns NAND into high-bandwidth memory....21
Part VII: HBF faces both architectural limitations and broader adoption challenges....23
Part VIII: Sandisk is translating NAND and systems leadership into an ecosystem-backed HBF roadmap....25

DETAILS

On the back of our AI Memory Primer, we follow up with today's deep dive into HBF (high-bandwidth flash) and other high performance NAND solutions as new memory tiers to help solve the memory wall and significantly improve memory capacity.

PART I. THE MEMORY WALL - AI IS SHIFTING FROM COMPUTE-CENTRIC TO MEMORY-CENTRIC

AI's evolution from training to advanced inference and agentic workloads is making memory capacity, bandwidth, and data reuse increasingly important alongside compute. Larger models and expanding context push data beyond HBM into the broader memory and storage hierarchy, while caching creates an economic incentive to retain rather than recompute it. The result is a more memory-centric infrastructure mix, broadening demand beyond just GPU and HBM into conventional DRAM, NAND, and HDD.

The widening memory wall is an increasingly important constraint on AI performance. Processor performance has advanced faster than memory bandwidth, creating a growing gap between how quickly XPUs can perform calculations and how quickly memory can supply the required data. Epoch AI estimates that training compute for frontier LLMs has increased by approximately 5x per year since 2020, while GPU memory bandwidth has grown by about 28% per year since 2008. Although these measure different dimensions of scaling, they illustrate the pressure to expand memory capabilities alongside compute. Adding compute alone becomes less effective when memory capacity and bandwidth constrain utilization.

Against this backdrop, the evolution from training to basic inference, **advanced inference, and agentic AI is shifting infrastructure requirements from compute-centric toward memory-centric architectures**. Training placed the most visible demands on GPUs and HBM, while advanced inference and agentic workloads broaden the requirements across processors, memory, and storage. Larger parameter bases, longer contexts, and persistent interactions increase the amount of data that must remain accessible. As these datasets exceed the capacity that can economically reside in HBM, demand extends into conventional DRAM and NAND, with HDD and potentially tape supporting persistent and archival data rather than latency-sensitive inference.

The shift from training to advanced inference makes the memory wall more pronounced. Training is generally compute-bound, but LLM inference is often memory-bandwidth-bound during decoding, when tokens are generated sequentially. In a dense model, each new token requires the XPU to read nearly the entire parameter set from HBM, despite performing relatively little computation per byte transferred. The XPU must also access the KV cache, which stores attention states from previously processed tokens and grows as context length increases. Consequently, compute units can remain underutilized while waiting for model weights and KV-cache data to arrive from memory.

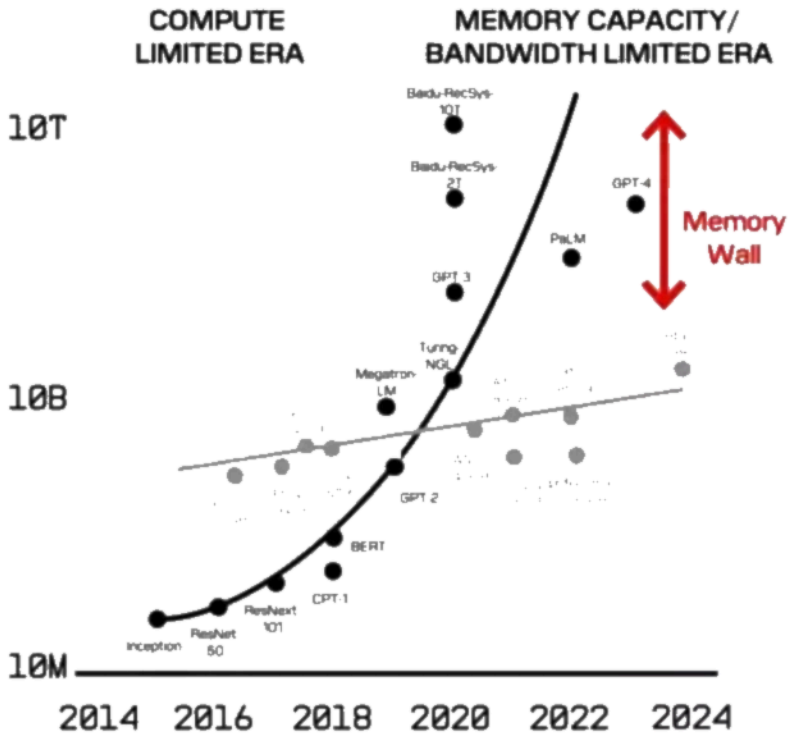
Mixture-of-Experts (MoE) architectures further shift the balance of AI infrastructure from compute toward memory. MoE models improve computational efficiency by routing each token to only a few specialized experts rather than activating the entire model. This allows total parameter counts to increase substantially while active parameters, and therefore computation per token, grow much more slowly (Exhibit 4). However, the full parameter base must still be stored and made accessible,

meaning MoE reduces compute requirements without proportionately reducing memory needs.

Token economics create an additional incentive to store and reuse more data rather than recompute it. Fresh tokens must pass through the full inference pipeline, requiring substantial computation, power, and processing time. Cached tokens instead retrieve previously computed KV states from memory, making them far cheaper and faster to serve. AI labs and Google currently price cached-token reads at approximately 10% of fresh input tokens (Exhibit 5), illustrating the significant economic advantage of retaining prior computations. As advanced inference and agentic AI require models to maintain longer context, remember prior interactions, and reuse previous work, the value of memory increasingly lies not only in supporting larger models, but also in avoiding repeated computation.

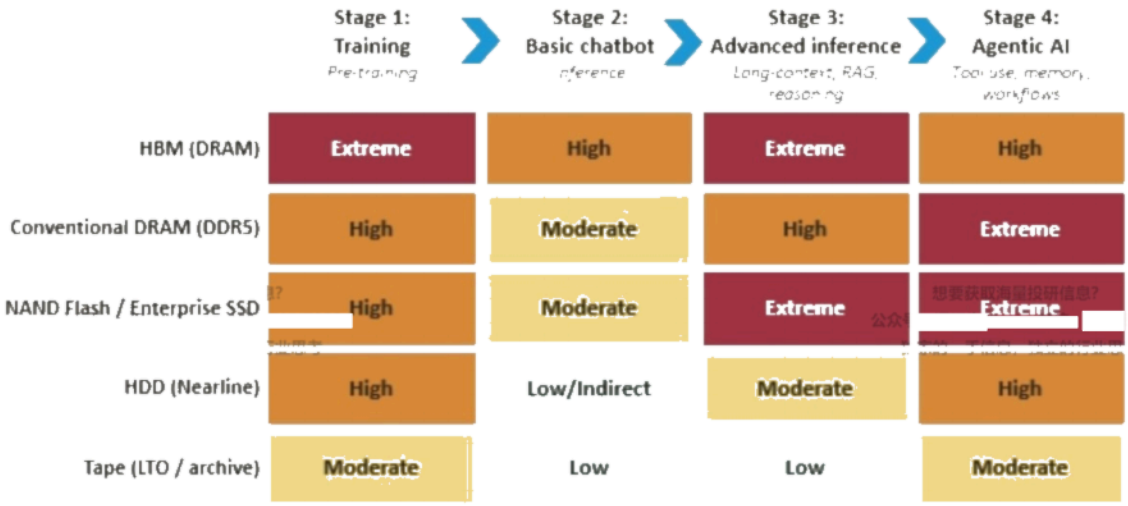
EXHIBIT 1: **The AI Memory Wall**

We are now memory capacity constrained instead of compute constrained



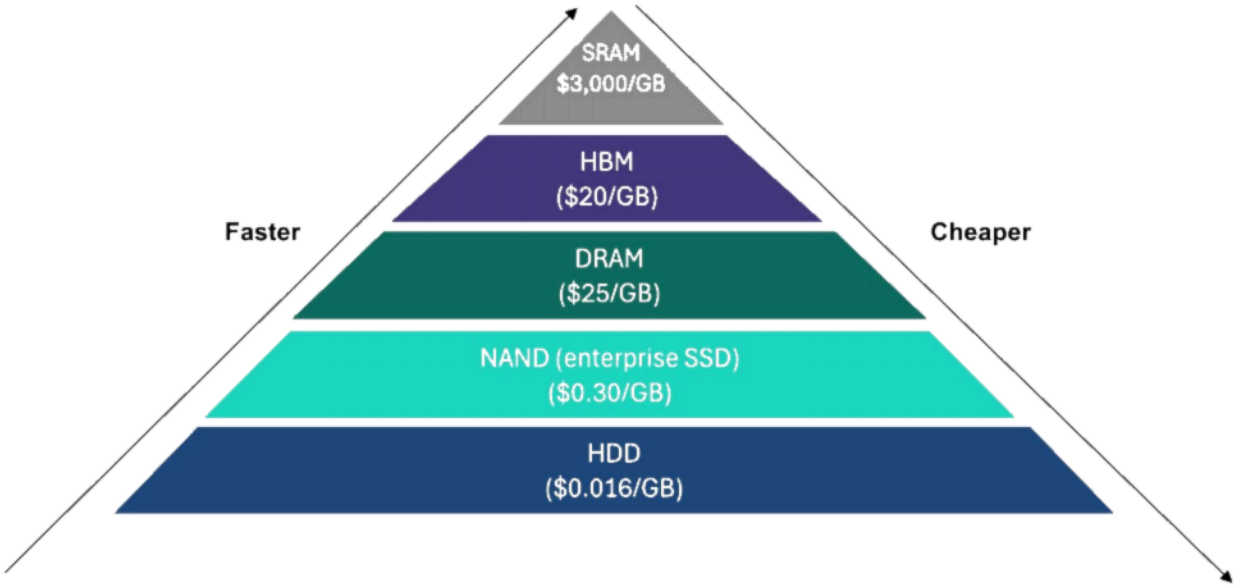
Source: Sandisk, IEEE Micro Journal

EXHIBIT 2: **The four stages of AI and impact on Memory & Storage demand**



Source: Bernstein analysis

EXHIBIT 3: **Memory and storage hierarchy for AI**



Source: Company reports, DRAMeXchange, Trendforce, Bernstein estimates & analysis, based on October pricing data

EXHIBIT 4: **The total parameters of models go up meanwhile active parameters go down**

Models	Released	Total Parameters	Mixture of Experts	Active Parameters	Context Length
Grok 4	Jul 9, 2025	~1.7T[1]	Dense (Est.)[1]	~1.7T[1]	256K
GLM 5.2	Jun 13, 2026	744B	256 (8 Active)[1]	40B	1M
GPT 5.6	Jul 9, 2026	~1.7-1.8T[1]	No Public Est.	No Public Est.	1.05M
Kimi K3	Jul 16, 2026	2.8T	896 (16 Active)	~50B[1]	1M



Source: Sandisk Investor Day presentation

EXHIBIT 5: **Cache tokens cost only 10% of Compute tokens**

Model	Compute ($/1M tokens)	Cache read ($/1M tokens)	Cache read as % of input
OpenAI gpt-5.6-sol	$5.00	$0.50	10.00%
OpenAI gpt-5.6-terra	$2.00	$0.20	10.00%
OpenAI gpt-5.6-luna	$0.20	$0.02	10.00%
OpenAI gpt-5.4	$2.50	$0.25	10.00%
OpenAI gpt-4o	$2.50	$1.25	50.00%
Gemini 3.1 Pro (<=200k)	$2.00	$0.20	10.00%
Gemini 3 Flash	$0.50	$0.05	10.00%
Gemini 2.5 Pro (<=200k)	$1.25	$0.13	10.00%
Gemini 2.5 Flash	$0.30	$0.03	10.00%
DeepSeek V4-Pro	$0.44	$0.00	~0.8%
DeepSeek V4-Flash	$0.14	$0.00	~2%

Source: https://developers.openai.com/api/docs/pricing; ai.google.dev; api-docs.deepseek.com, based on August cost

PART II: THE SOLUTION TO THE MEMORY WALL - HBF COULD MAKE AI MORE MEMORY-EFFICIENT AND COST-EFFECTIVE.

HBF could lower the cost of AI inference by combining HBM-like read bandwidth with 8-16x greater capacity at a meaningfully lower cost per GB. In a tiered architecture, HBM would retain the active, latency-sensitive working set, while HBF stores the broader model parameter base (e.g. inactive MoE experts) and warm or reusable KV cache. This could make AI inferencing cheaper by 1) reducing the GPU and HBM capacity required to host large models and 2) avoiding costly recomputation by retrieving previously processed context, with cached-token reads currently priced at approximately 10% of fresh input tokens. HBF's value proposition is therefore not simply cheaper memory, but a potentially more capital- and compute-efficient inference architecture.

HIGHER CAPACITY AND LOWER COST/GB THAN HBM

HBF is positioned as a potential solution to the AI memory wall because it materially expands the amount of high-bandwidth memory available near an accelerator. Sandisk's stated target is **HBM-class read bandwidth with 8-16x the capacity of HBM**, addressing the widening mismatch between rapidly rising model size and context length versus the limited capacity that can be placed in an HBM footprint. HBF is also intended to be a read-optimized memory technology, making it suitable for inference workloads that must repeatedly access large model weights and growing contexts.

Moreover, it is implied that an HBF stack could be priced at a similar level as an HBM stack while offering substantially higher capacity, implying a **meaningfully lower cost per GB. If realized, that makes HBF a compelling way to add memory capacity** without scaling the number of expensive HBM stacks and associated GPUs proportionately. Sandisk itself framed HBF as the solution to the memory wall problem.

HBF AS A NEW MEMORY TIER IN AI INFERENCE

At Investor Day, Sandisk's principal proposed use case was a new HBM + HBF tiered memory architecture for AI inference. In the hybrid design, a smaller **HBM pool serves as a fast cache tier in front of larger-capacity HBF, with HBF storing model weights and other KV cache.** The focus on inference is logical because HBF is designed for high-bandwidth, read-optimized memory access: inference repeatedly reads large model-weight sets and accumulated KV-cache attention state, whereas training is comparatively more write- and compute-intensive. Decode is especially memory-intensive because each token is generated autoregressively and must reference the prior context and KV cache; it therefore needs both high read bandwidth and substantially more memory capacity than a conventional HBM-only configuration may provide.

This HBM-as-cache, HBF-as-capacity-tier concept also has support from SK Hynix's H3 architecture (SK hynix in Feb, 2026 proposed the "H^3" architecture in a paper. **), which was proposed for cost-efficient LLM inference**. H3 similarly uses HBM and HBF in a hybrid hierarchy, with HBF aimed at large read-only or read-mostly inference data, including model weights and shared, precomputed KV cache. The practical relevance is that HBF does not need to displace HBM to create value: it can allow a fixed GPU/HBM deployment to serve more requests or larger models by adding a substantially larger high-bandwidth capacity tier.

HBF COULD LOWER TOTAL SYSTEM COST BY REDUCING GPU/HBM REQUIRED TO ACCOMMODATE LARGE MODELS.

HBF appears particularly well aligned with large models and sparse Mixture-of-Experts architectures. The key point is that an MoE model may have trillions of total parameters, but **activate only a small subset of experts** for each token. Nevertheless, the total set of model weights still needs to be available with sufficiently high bandwidth. HBF can serve as a high-capacity backing tier for the full model or for less frequently accessed experts, while HBM retains the active working set and caches weight blocks expected to be needed next.

This can reduce the need to provision additional GPUs and HBM capacity merely to fit the complete model in HBM. Sandisk's illustrative simulation using Qwen3-480B-A35B showed an HBF-only setup requiring one GPU to run the 960GB bfloat16 model, versus eight HBM GPUs for the minimum HBM-only configuration, and claimed four HBF GPUs could achieve the token output of eight HBM GPUs. These are internal simulations rather than third-party benchmarks, but they illustrate the CapEx-reduction thesis: added HBF capacity can potentially lower total system cost even if the HBF memory itself is not inexpensive.

The timing of this architecture is favorable because frontier models are becoming larger and more likely to adopt MoE. Sandisk highlighted the combination of rising total parameter counts, longer context windows, and more widespread use of MoE routing. Its examples included models with total parameter counts ranging from hundreds of billions to multiple trillions and context windows approaching or exceeding 1 million tokens, while active MoE parameters may be only tens of billions. **This trend increases the value of a memory tier that can keep a much larger weight footprint close to compute without requiring a commensurate increase in HBM capacity.**

HBF CAN POTENTIALLY REDUCE COMPUTE COST BY STORING KV CACHE

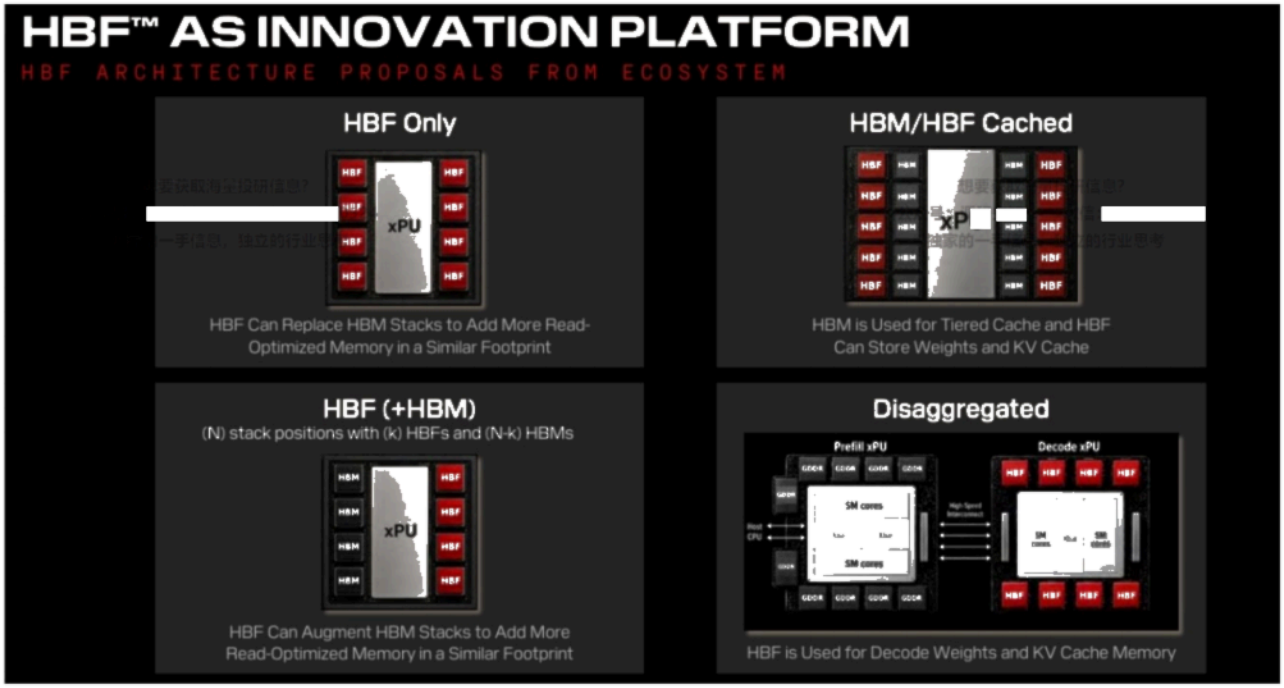
HBF could also be useful for selected portions of KV cache, though the most natural architecture is tiered rather than placing all live KV state in HBF. The hottest, actively growing KV-cache tail is likely best maintained in HBM because it is latency-sensitive and updated continuously during decode. HBF is better suited to a larger retained tier: persistent session context, warm or cold KV segments, reusable shared prompt prefixes, and precomputed KV tensors that may be fetched repeatedly across requests or turns. This extends the useful lifetime of prior compute.

Rather than discard prior attention calculations and pay GPU power and compute cost to recreate them, the system can retrieve the relevant stored context from HBF. Sandisk made this exact economic argument at the Investor Day, describing storage as a "token battery": stored tokens represent work already done, and fetching them can avoid expensive recomputation. Frontier labs and Google currently price cached-token reads at approximately 10% of fresh input tokens (Exhibit 5).

The resulting value proposition is therefore not just lower memory cost per GB. It is an architecture-level improvement: combine a relatively small HBM cache for the hottest state with far more HBF capacity for weights and

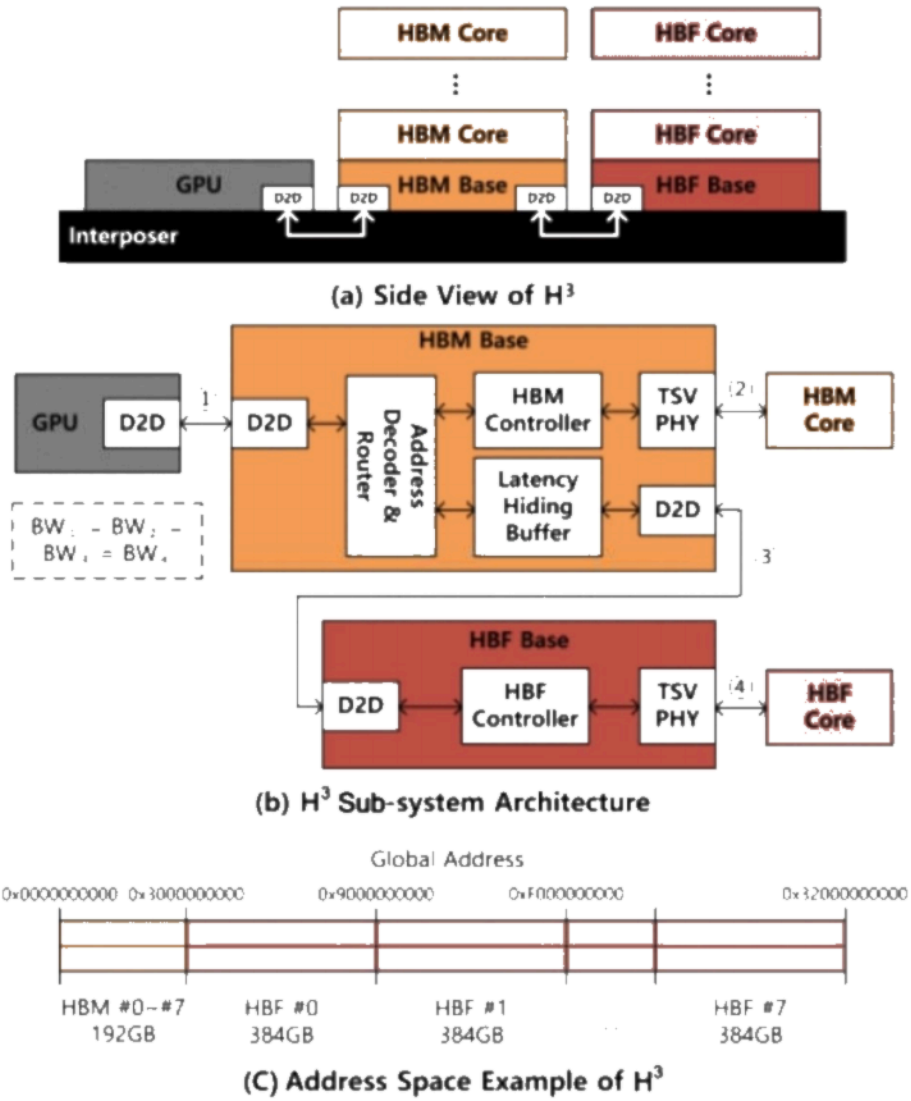
retained context, potentially reducing the GPU/HBM count needed to host a large model and lowering recompute, power, and infrastructure cost. For large, increasingly MoE-based inference workloads, HBF could become a high-bandwidth capacity layer that sits between scarce HBM and slower SSD-based persistent storage.

EXHIBIT 6: **SanDisk proposed four architectural proposals for HBF.**



Source: SNDK Investor Day 2026

EXHIBIT 7: **H3 Architecture overview**



Source: M. Ha, E. Kim and H. Kim, "H3: Hybrid Architecture Using High Bandwidth Memory and High Bandwidth Flash for Cost-Efficient LLM Inference," in IEEE Computer Architecture Letters, vol. 25, no. 1, pp. 49-52, Jan.-June 2026, doi: 10.1109/LCA.2026.3660969.

PART III: HIGH PERFORMANCE NAND SUCH AS HBF AS NEW MEMORY TIER TO HELP NAND SUPPLY/ DEMAND BALANCE AND PRICING

Multiple technologies are emerging to add a new memory tier between HBM and conventional NAND, with each addressing different AI inference requirements. HBF combines high sustained bandwidth with 8-16x greater capacity than HBM, while alternatives focus on low-latency flash access, pooled KV-cache retention, and other specialized use cases. Importantly, many emerging solutions are NAND-based because improving NAND performance may be more practical and economical than trying to make DRAM sufficiently dense to serve this capacity role. As most of these technologies use NAND and often low density NAND with a high trade ratio (see economics section for more details), their adoption should consume significant NAND wafer supply and thus tighten NAND supply/demand balance regardless of which architecture gains the most share.

As AI inference requires more memory capacity and bandwidth than HBM can economically provide, the industry is developing new memory tiers between HBM and conventional NAND. HBM provides the bandwidth and latency required for active computation, but its capacity is limited and expanding the available HBM pool often requires adding expensive GPUs.

Conventional SSDs provide much greater capacity at a lower cost per bit, but generally lack the bandwidth and latency required to supply model weights directly during inference. Technologies including HBF seek to fill this gap by providing more capacity than HBM alone and higher bandwidth than conventional large-capacity SSD storage.

NAND is the most economically logical foundation for this new tier because its density and cost advantages over DRAM are substantial. NAND has historically achieved much greater bit density than DRAM, with SLC NAND roughly 5-10x denser than DRAM. NAND also costs materially less per GB, with SLC approximately 3x cheaper than DRAM. SLC stores one bit per cell, compared with three bits for mainstream TLC, enabling faster access and better performance at the expense of lower density. Even after accepting this approximately 3x density trade-off versus TLC, SLC retains meaningful density and cost advantages over DRAM. It is therefore more practical to sacrifice some of NAND's density to improve its speed and access characteristics than to make DRAM cheap and dense enough to provide comparable capacity.

HBF is one of several technologies being developed to create this new memory tier for AI inference, and all of these alternatives except for CXL-attached DRAM are NAND based solutions:

- **HBF:** HBF stacks NAND in a highly parallel, wide-interface architecture designed to provide sustained bandwidth closer to HBM while offering 8-16x greater capacity. Its most natural use case is storing large, read-intensive datasets such as model weights, including less frequently accessed experts in sparse MoE models. HBF could also retain warm or reusable KV-cache data, supporting both model capacity and token economics within the same tier. **Because HBF uses lower-density NAND and requires significantly more wafer capacity for its architecture, we estimate an approximately 6.5x wafer-capacity trade ratio as the base case.**

- **Super High IOPS SSDs (Storage Next)** - locally attached, GPU-initiated. This often uses SLC (single level cell) NAND instead of TLC (3 bits per cell) with high IOPS (input/output operations per second) and low latency. Storage Next is led by NVIDIA and supported by KIOXIA, Samsung, SK hynix, Micron, Sandisk, and many other companies. KIOXIA's GP-series SSDs for example combine PCIe with XL-FLASH to deliver very high random-read IOPS and lower latency than conventional SSDs. This is one of the closest functional alternatives to HBF, although it focuses on small, highly parallel flash accesses rather than HBF's wide-interface approach to sustained bandwidth. **Because XL-FLASH uses SLC NAND, it stores one bit per cell compared with three bits for TLC, implying an approximately 3x capacity trade ratio before considering additional architectural overhead.**

- **Other low-latency NAND solutions:** Technologies such as **Samsung Z-NAND are similar to KIOXIA's XL-FLASH and** redesign the NAND media itself to reduce read latency through smaller pages, greater parallelism, and SLC or other performance-optimized configurations. These technologies can support faster model-weight access and KV-cache offload, but their primary advantage is lower latency rather than HBF-like aggregate bandwidth. **Their use of SLC or other lower-density configurations also creates a meaningful wafer-capacity trade ratio relative to mainstream TLC NAND.**

- **Pooled flash KV-cache platforms:** NVIDIA CMX/ICMS content memory storage platform creates a pod-level flash tier that retains and shares data across a GPU cluster using DPUs, networking, and NVMe SSDs. It is optimized for KV-cache, with the primary objective of preserving reusable context and avoiding costly recomputation. **CMX could primarily use conventional TLC NAND and therefore may not have the same per-bit trade ratio as HBF or SLC-based flash. However, it could still add substantial incremental NAND demand by adding petabytes of flash-backed context capacity to AI clusters.**

- **CXL-attached DRAM:** CXL is later expanded to support NAND, but the technology is primarily for DRAM. CXL memory expanders place additional DRAM behind the PCIe physical layer and allow memory capacity to be pooled across processors or servers. This can expand KV-cache capacity and provide coherent, memory-like access, but bandwidth and latency are constrained by the CXL link and system topology. However, CXL-attached DRAM retains DRAM's structurally higher cost per bit: **CXL improves how DRAM is connected and shared, but does not eliminate DRAM's underlying cost and density disadvantage.**

ALTERNATIVE HIGH PERFORMANCE NAND BASED SOLUTIONS OTHER THAN HBF CAN ALSO HAVE POSITIVE IMPACT ON NAND INDUSTRY

Even if HBF faces adoption challenges, the proliferation of higher-performance NAND technologies should benefit the NAND industry by creating incremental demand and shifting wafers toward lower-density products that consume more capacity per bit, tightening supply and supporting pricing. The profitability of these products will be subject

to competition with DRAM (currently priced almost 100x more expensive), supply/demand balance & ultimately the economy of AI.

Importantly, competition among these technologies does not diminish the broader NAND-industry opportunity. Several of the most relevant solutions use NAND as the underlying capacity medium and improve performance by sacrificing bit density relative to mainstream TLC. HBF, SLC-based XL-FLASH, low-latency NAND, and flash-backed KV-cache platforms could all consume additional NAND wafer capacity. Some have a direct trade ratio because they store fewer bits per cell, while approaches like CMX increase NAND demand by adding entirely new pools of AI-related storage.

The broader shift to higher-performance and smaller-capacity NAND could therefore tighten industry supply even if HBF encounters adoption challenges. Lower-density, higher-performance products would reduce the usable NAND capacity produced per wafer, while model storage and KV-cache retention would create incremental AI-related demand. Together, these trends could constrain conventional TLC bit-supply growth and support NAND pricing even if HBF runs into snags.

The shift toward higher-performance NAND could have significant impact on industry capacity because even a modest amount of demand may consume a disproportionate number of wafers. Converting TLC to SLC reduces the bits stored per cell from three to one, creating an approximately 3x trade ratio before considering additional architectural overhead. More specialized products may also require larger logic areas, or more complex packaging, pushing the effective trade ratio higher for architectural reasons. In HBF's case, we estimate a midpoint trade ratio of approximately 6.5x versus conventional TLC.

As an illustrative sensitivity, **if 10-20% of current TLC-equivalent GB demand migrated to higher-performance NAND** with a 3-7x wafer trade ratio, total industry **wafer requirements would increase by approximately 20-120%** versus producing the same usable capacity in TLC. While a higher trade ratio reduces the almost 100x cost advantage of NAND vs. DRAM, and this is not our official demand forecast, it illustrates how even modest adoption could have a disproportionate impact on NAND wafer requirements.

The industry would be unlikely to accommodate such incremental wafer demand quickly. Suppliers can increase bit output through higher layer counts, productivity improvements, and higher utilization, but meaningful additions to wafer capacity require substantial capital spending, equipment installation, and multi-year fab construction. **NAND supply could therefore tighten further before sufficient new capacity becomes available, supporting utilization, pricing, and margins at these levels for longer.**

Higher-performance NAND products could also command a premium over conventional NAND. Products such as HBF deliver value through bandwidth, latency, packaging, and system-level efficiency rather than raw storage capacity alone. **Customers may therefore be willing to pay more per usable GB, creating an additional benefit to NAND pricing and product mix beyond the wafer-capacity impact.** As the trade ratios and additional complexities will also result in higher cost, and how the profitability of these high-performance NAND products will be determined by the competition vs. DRAM products, supply & demand and utimately by the economy of AI. HBF economics is studied in depth in the later Economics section.

WE BELIEVE HBF IS COMPETITIVE THOUGH BARRIERS TO ADOPTION REMAIN

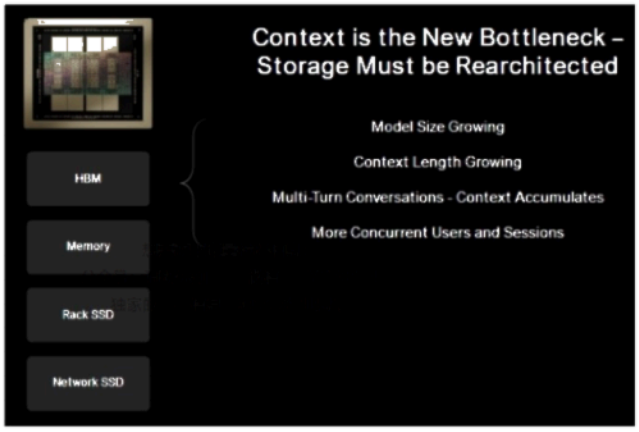
While multiple technologies could coexist, HBF is differentiated by combining proximity to the accelerator, large capacity, and high sustained bandwidth to support both model parameters and reusable KV cache within a single tier.

These emerging technologies could coexist because each optimizes for different performance requirements. CMX and other pooled flash platforms focus on retaining and sharing reusable KV-cache data, while GPU-direct flash emphasizes fine-grained, low-latency, high-IOPS access through a storage-oriented architecture. HBF instead targets large, read-intensive datasets through a wide, highly parallel interface optimized for sustained bandwidth. AI systems could therefore deploy different solutions for various combinations of capacity, bandwidth, latency, access granularity, and cost.

We believe HBF's competitive advantage is its ability to address both model capacity and reusable context within a single tier close to the accelerator. Its large capacity and high sustained bandwidth allow it to store and supply large model parameter sets, including inactive MoE experts, while also retaining warm or reusable KV-cache data to avoid costly recomputation. KIOXIA's GP-series and other GPU-centric SSDs are optimized for quickly retrieving small pieces of data, but their narrower storage connection makes them less suited to continuously supplying large model-weight datasets. CMX provides much greater shared capacity across a GPU cluster, but stores data in a separate networked pool, making it more suitable for retaining reusable KV cache than serving model weights. HBF therefore offers a differentiated combination of

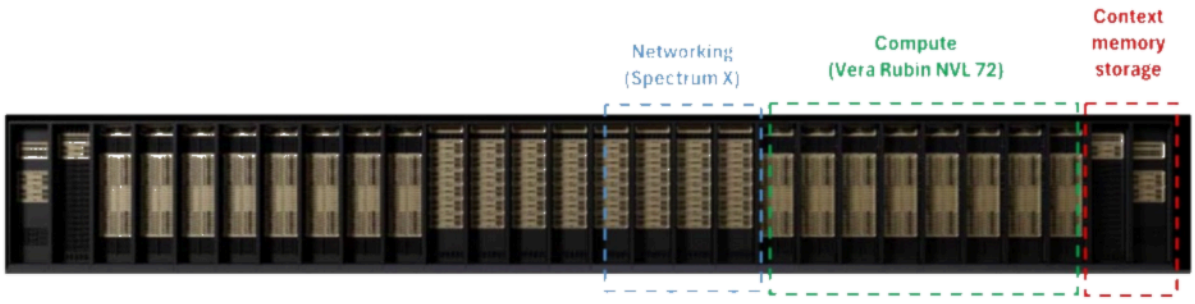
proximity, capacity, and sustained bandwidth across both use cases.

EXHIBIT 8: **Jensen's comments on storage**



Source: Nvidia CES 2026

EXHIBIT 9: **Vera Rubin Pod demonstration**



Source: Nvidia CES 2026

EXHIBIT 10: **KIOXIA's GP1 is an example of its GP-series SSDs.**



Source: KIOXIA company website

PART IV: HBF ECONOMIC ANALYSIS - HIGH COST & TRADE RATIO BUT ACCRETIVE TO REVENUE & PROFIT

In this section we do a deep dive in to the potential economics of HBF. Due to the higher performance and higher durability SLC flash used and additional silicon area needed for TSV, high-bandwidth I/O and related peripheral circuitry, we estimate HBF trade ratio at 6.5x, much higher than the roughly 3x we have seen for HBM (due mainly to the additional SLC penalty). We estimate the cost ratio of HBF at roughly 10x depending on the configuration. Despite this huge cost ratio, by pricing HBF similar to HBM on a per-module basis (with 8-16x more capacity and similar read bandwidth) we estimate that HBF can get similar gross margins of NAND flash (in the 80-90%) therefore being hugely accretive to ASP, revenue and overall profit.

HBF-TO-NAND TRADE-RATIO FRAMEWORK

We estimate that HBF could consume approximately 6.5x as much NAND manufacturing capacity per bit as conventional TLC NAND in our base case. Our framework begins with the established relationship between HBM and conventional DRAM. An HBM die is estimated to be approximately 2.3x larger than a conventional DRAM die of equivalent capacity, reflecting the additional area required for TSVs, high-bandwidth I/O, and related circuitry. Applying an approximately 1.3x effective yield and manufacturing penalty for stacking, bonding, and other process complexity produces an aggregate HBM-to-DRAM trade ratio of approximately 3.0x:

HBM-to-DRAM trade ratio = approximately 2.3x die-area penalty × approximately 1.3x effective yield penalty = approximately 3.0x.

We use the same basic framework to estimate the HBF-to-NAND trade ratio but adjust the die-area assumption to reflect important architectural differences between HBF and HBM.

The first component is an approximately 2.9x TLC-to-SLC density penalty. Certain patent disclosures suggest that HBF could operate NAND in SLC mode, which we therefore use as our working assumption. SLC stores one bit per cell, while TLC

stores three bits per cell, implying a theoretical density difference of 3.0x. However, TLC's tighter voltage-state separation and lower effective quality result in some loss of usable capacity. We therefore use an effective TLC-to-SLC density penalty of approximately 2.9x rather than the theoretical 3.0x.

The second component is an approximately 1.725x HBF architectural and die-area penalty in our base case. Like HBM, HBF requires additional die area for TSVs, high-bandwidth I/O, and related peripheral circuitry. However, we believe HBF's incremental area penalty should be lower than HBM's approximately 2.3x penalty because HBF uses circuit-bonded array, or CBA, architecture.

Under CBA, the CMOS and peripheral circuitry are fabricated separately and vertically bonded underneath the NAND array. This allows much of the circuitry to overlap the NAND-array footprint rather than expanding the memory die laterally. HBF does not eliminate per-die peripheral CMOS, as the NAND arrays still require local CMOS and peripheral circuitry, but CBA allows the circuitry to be incorporated more area-efficiently. By contrast, HBM does not use NAND-style CBA, meaning more of its TSV, I/O, and peripheral overhead directly expands the DRAM die footprint.

Our base case assumes that CBA reduces HBF's architectural area penalty by 25% relative to HBM:

2.3x HBM die-area penalty × (1 − 25% CBA savings) = 1.725x HBF die-area penalty.

The final component is an approximately 1.3x effective yield and manufacturing penalty. We apply the same penalty used in the HBM framework because both HBF and HBM require TSVs, vertical stacking, bonding, wafer thinning, and other advanced manufacturing processes. Although the ultimate penalty will depend on HBF's final stack configuration and manufacturing yields, we believe HBM provides a reasonable initial analogy.

Combining these three components produces a base-case HBF-to-NAND trade ratio of approximately 6.5x:

HBF-to-NAND trade ratio = 2.9x TLC-to-SLC density penalty × 1.725x die-area penalty × 1.3x effective yield penalty = approximately 6.5x.

HBF-to-NAND trade-ratio scenarios

We model three scenarios based on the amount of area savings HBF realizes from its CBA architecture. The SLC density and effective yield assumptions remain unchanged across the scenarios; **the only changing variable is the assumed CBA-related reduction in HBF's die-area penalty relative to HBM.**

- **High trade-ratio scenario — 8.7x:** Assumes HBF receives no area benefit from CBA and therefore incurs the full 2.3x HBM-like die-area penalty.
- **Base-case scenario — 6.5x:** Assumes CBA reduces the HBF die-area penalty by 25%, resulting in a 1.725x area penalty.
- **Low trade-ratio scenario — 4.3x:** Assumes CBA reduces the HBF die-area penalty by 50%, resulting in a 1.15x area penalty.

We believe approximately 6.5x is the cleanest base-case HBF-to-NAND trade ratio. This assumption recognizes that HBF should incur a meaningful architectural area penalty relative to conventional NAND, while also giving credit to CBA's ability to place CMOS and peripheral circuitry underneath the NAND array and thereby limit lateral die expansion.

EXHIBIT 11: **Assumed HBF to NAND Trade Ratio Scenarios**

	HBM to DRAM	HBF to NAND (High)	HBF to NAND (Mid)	HBF to NAND (Low)
Net Trade Ratio	**3.0**	**8.7**	**6.5**	**4.3**
=				
Total Die Size Penalty		**6.7**	**5.0**	**3.3**
Die Size Penalty	*2.3*	*2.3*	*1.725*	*1.15*
includes HBF Architecture Savings [CBA]		*0%*	*25%*	*50%*
x SLC to TLC Density		*2.9*	*2.9*	*2.9*
x Other Yield Loss	**1.3**	**1.3**	**1.3**	**1.3**

Source: Bernstein estimates and analysis

CONVERTING HBF TRADE RATIOS INTO COST RATIOS

To estimate HBF manufacturing costs, we begin with SanDisk's FQ4 blended NAND manufacturing cost of approximately **$0.0485/GB**. We estimate that raw TLC NAND die cost/GB is roughly 70% of the total blended cost/GB, arriving at approximately **$0.034/GB**.

We then apply our assumed **HBF-to-TLC trade ratios** to estimate HBF raw die cost. Under our low, base, and high cases, the trade ratios are **4.34x, 6.5x, and 8.67x**, respectively. This results in estimated HBF raw die costs of approximately **$0.15/GB, $0.22/GB, and $0.29/GB**.

Translating these per-GB costs into stack-level die costs, an **8-hi HBF stack with 256 GB of capacity** carries an estimated raw die cost of **$38-$75 per module**, with a base-case estimate of **$57**. A **16-hi HBF stack with 512 GB of capacity** carries an estimated raw die cost of **$75-$151 per module**, with a base-case estimate of **$113**.

HBM4 Is the Appropriate Benchmark for HBF

To translate HBF raw die costs into total module costs, we benchmark the remaining logic/base die, packaging, stacking, and testing costs against HBM4.

We believe HBM4 is the most appropriate benchmark for two reasons. First, SanDisk has stated that its first-generation HBF product is purpose-built to closely match the **physical footprint, power profile, and stack height of HBM4**. This suggests that HBF might require a similar packaging architecture, stack integration, and testing process. Second, HBM4 is expected to become the primary HBM volume driver by the time HBF enters commercial production. Benchmarking HBF against HBM4 therefore provides a more relevant estimate of the manufacturing cost structure HBF will face at launch than benchmarking against today's HBM3E products.

Detailed HBM4 cost estimates are not widely available. We therefore first use HBM3E to establish a reasonable baseline, since HBM3E is the current volume driver, before adjusting the assumptions to reflect HBM4's more advanced base die and moderately higher integration costs. In other words, **HBM3E is used only as an input to estimate certain HBM4 cost components, while the final HBF cost ratios are benchmarked against HBM4**.

Using HBM3E to Establish the Cost Baseline

For HBM3E, we assume an ASP of **$20/GB** and a gross margin of approximately **72%**, implying a total manufacturing cost of approximately **$134** for an 8-hi, 24 GB HBM3E stack.

Based on our Micron model, DRAM COGS is approximately **$1.52/GB**. Assuming that **75%** of DRAM COGS represents leading-edge raw DRAM die cost and applying a **3x HBM-to-DRAM die cost ratio**, we estimate HBM3E raw memory die cost at

approximately:

$1.52/GB x 75% x 3 x 24 GB = $82

This leaves approximately **$52** of non-memory-die costs, which we allocate as follows:

- **Logic/base die:** $12
- **Packaging and stacking:** $25
- **Testing and other:** $15

For a 16-hi stack, we keep the logic/base die cost unchanged while doubling packaging, stacking, and testing costs to reflect the greater stack height and manufacturing complexity. This implies approximately **$93** of non-memory-die costs for a 16-hi HBM3E stack.

These HBM3E estimates provide a baseline for conventional HBM packaging and testing costs, but they are not the final benchmark used to calculate HBF cost ratios.

EXHIBIT 12: **HBM3E Cost Breakdown Assumptions**

	HBM3E other costs		
8-hi	Base Die Cost	$	12
	Packaging/stacking	$	25
	Testing and other	$	15
	Other costs total excl raw dies	$	52
16-hi	**Other costs total excl raw dies**		$93
Assumptions	HBM3E ASP	$	20.00
	HBM3E gross margin		72%
	HBM3E 8hi capacity		24
	Cost of full stack	**$**	**134**
	DRAM COGS/GB from MU model		1.52
	Leading edge die cost ratio		75%
	HBM 3E raw die cost	**$**	**82**
	Other costs	**$**	**52**

Source: Company reports, Bernstein estimates and analysis

Adjusting the HBM3E Baseline Cost for HBM4 Cost

The primary difference between HBM3E and HBM4 is the logic/base die. For HBM4, we estimate **$57 base die cost**, compared with approximately $12 for HBM3E. HBM4's base die is substantially more expensive because HBM4 transitions from a lower-cost proprietary memory process to an advanced foundry logic node, such as a 5 nm-class process, to support its doubled 2,048-bit interface and more advanced logic functionality. Based on the JEDEC standard, we estimate that the HBM4 base die measures approximately 9.5 mm x 13 mm, a footprint that HBF's base die is expected to match. This die size implies roughly 500 gross dies and 400 good dies per 12-inch wafer; assuming a 5 nm wafer cost of approximately $22,000, we estimate an HBF base die cost of approximately $55 per good die, and to be conservative we used $57.

We assume packaging and stacking costs of $30 and testing and other costs of $20 for an 8-hi HBM4-equivalent module, modestly above our HBM3E estimates of $25 and $15, respectively. We believe the more advanced base die should account for most of the cost increase from HBM3E to HBM4, while packaging and testing costs should increase only moderately.

For a 16-hi module, we keep the **$57 base die cost unchanged**, while doubling packaging, stacking, and testing costs to reflect the larger number of memory dies and greater integration complexity.

This produces the following HBM4-based non-memory-die cost assumptions:

EXHIBIT 13: **HBM4 Cost Breakdown Assumptions**

	HBM4 Cost Breakdown	
8-hi	Capacity (GB)	32
	Base Die Cost	$57
	Packaging/stacking	$30
	Testing & other	$20
	Total other cost exl dies	**$107**
	Total Package cost	$217
16-hi	Capacity (GB)	64
	Total other cost exl dies	**$158**
	Total Package cost	$377

Source: Bernstein estimates and analysis

Cost Ratio: Comparing HBF Cost/GB With Enterprise SSD Cost/GB

Combining the HBM4 other costs (excluding raw DRAM dies) with our estimated HBF raw die costs produces the following total HBF module costs:

- **16-Hi:** $233-308 per module (base: **$271**)
- **8-Hi:** $145-$183 per module (base: **$164)**

Dividing by capacity gives HBF cost/GB. Finally, comparing HBF module cost per GB with the eSSD cost of **$0.056/GB** produces the implied **cost ratio**.

The eSSD cost estimate is derived from SanDisk's disclosed FQ4 blended cost per GB and segment mix, assuming that the data-center segment primarily represents enterprise SSDs while the remaining segments consist of a combination of finished products and raw NAND.

Therefore, we estimate an HBF manufacturing cost premium relative to enterprise SSDs of approximately **10.2x-12.8x for 8-hi HBF** and **8.2x-10.8x for 16-hi HBF**. Under the base case, the implied cost ratios are approximately **11.5x for 8-hi** and **9.5x for 16-hi**.

The lower cost ratio for 16-hi reflects better fixed-cost absorption. The same **$57 logic/base die** is spread across **512 GB of HBF capacity in a 16-hi stack**, compared with only **256 GB in an 8-hi stack**. As a result, 16-hi HBF carries a higher absolute module cost but a lower manufacturing cost per GB.

EXHIBIT 14: **HBF Cost Breakdown and Cost Ratio Analysis**

	High	Mid	Low
Trade Ratios	8.67	6.50	4.34
TLC Raw Die Cost/GB	$0.034	$0.034	$0.034
HBF Raw Die Cost/GB	$0.29	$0.22	$0.15
16-hi (GB)	512	512	512
8-hi (GB)	256	256	256
HBF Raw Die Cost 16-hi	$ 151	$ 113	$ 75
HBF Raw Die Cost 8-hi	$ 75	$ 57	$ 38
Other Costs 16-hi	$158	$158	$158
Other Costs 8-hi	$107	$107	$107
Module Cost 16-hi	$308	$271	$233
Module Cost 8-hi	$183	$164	$145
Module Cost/GB 16-hi	$0.60	$0.53	$0.46
Module Cost/GB 8-hi	$0.71	$0.64	$0.57
Cost Ratio to eSSD TLC (16hi)	10.8	9.5	8.2
Cost Ratio to eSSD TLC (8hi)	12.8	11.5	10.2

Source: Bernstein estimates and analysis

HBF PRICING AND GROSS MARGIN ANALYSIS

On pricing, SanDisk has stated that HBF is designed to deliver comparable bandwidth to HBM while offering **8-16x greater capacity at a similar module cost**. For our analysis, we assume that a 256 GB 8-hi HBF module is comparable to a 32 GB 8-hi HBM4 module and a 512 GB 16-hi HBF module is comparable to a 64 GB 16-hi HBM4 module. If HBF and HBM4 have similar module prices, the implied HBF ASP per GB is therefore calculated by dividing the HBM4 module ASP with the corresponding HBF capacity.

We consider three HBM4 pricing scenarios:

- **Bear case:** HBM4 ASP of **$23.50/GB**, representing current pricing, implies an HBF ASP of approximately **$2.94/GB**
- **Base case:** HBM4 ASP of **$38.00/GB**, representing 2027 pricing expectations, implies an HBF ASP of approximately **$4.75/GB**
- **Bull case:** HBM4 ASP of **$50.00/GB**, representing a potential 2027 upside scenario in which HBM4 pricing roughly doubles from current levels, implies an HBF ASP of approximately **$6.25/GB**

These assumptions produce equivalent module prices for HBM4 and HBF despite HBF's substantially greater capacity. Under the base case, for example, a 32 GB HBM4 module and a 256 GB HBF module would both sell for approximately **$1,216**, while a 64 GB HBM4 module and a 512 GB HBF module would both sell for approximately **$2,432**.

EXHIBIT 15: **HBF Pricing Assumptions**

HBM ASP Scenarios	HBM4 ASP/GB	Implied HBF ASP/GB	HBM4 Module ASP (8hi 32GB)	HBM4 Module ASP (16 hi 64GB)	HBF Module ASP (256 GB)	HBF Module ASP (512 GB)
Low	$23.50	$2.94	$752	$1,504	$752	$1,504
Mid	$38.00	$4.75	$1,216	$2,432	$1,216	$2,432
High	$50.00	$6.25	$1,600	$3,200	$1,600	$3,200

Source: Bernstein estimates and analysis, DRAMeXchange

On cost, we estimate enterprise SSD manufacturing cost at approximately **$0.056/GB**. We then apply the cost ratios we found in the previous analysis to find the HBF cost/GB. Taken together with our pricing assumptions, we calculate low, mid, and high scenarios for HBF gross margin.

At our base-case HBF ASP of **$4.75/GB**, the implied gross margin is approximately **88.9% for 16-hi HBF** and **86.5% for 8-hi HBF**. Both configurations could therefore be accretive relative to SanDisk's **84.6% FQ4 gross margin**, with the 16-hi configuration benefiting from superior fixed-cost absorption. Although margins would be lower under the bear pricing scenario and higher under the bull scenario, the base case suggests that HBF could generate gross margins comfortably above SanDisk's current corporate level.

The ultimate gross margin will therefore depend primarily on two variables:

1. **The realized HBF cost/GB**, which will be determined by the HBF-to-NAND trade ratio and the ultimate cost of the base die, packaging, stacking, and testing.

2. **The HBF selling price**, which will depend on HBF's performance relative to HBM as well as the memory (DRAM/NAND) cycle strength.

The conclusion above admittedly is subject to the price of HBM, and thus can be impacted by the supply/demand of DRAM as HBM largely shares the same production capacity with conventional DRAM. And as said by Micron in the latest earnings release, though recent contract negotiations make HBM price higher YoY in 2027, HBM margin will stay below conventional DRAM margin in 2027. This means suppliers recognize the "strategic" value of HBM and agree to accept lower profitability on HBM than on conventional DRAM. We could see a similar dynamic play out in HBF. Regardless, we conclude that **HBF will hugely accretive to NAND ASP, revenue and profits and gross margin should be highly attractive (though the actual pricing level is too early to know).**

EXHIBIT 16: **HBF Gross Margin Scenario Analysis**

	HBF:eSSD Cost Ratio	HBF Cost/GB	GM (Low)	GM (Mid)	GM (High)
Low 16hi	8.2	$0.46	84.5%	90.4%	92.7%
Low 8hi	10.2	$0.57	80.7%	88.1%	90.9%
Mid 16hi	9.5	$0.53	82.0%	88.9%	91.5%
Mid 8hi	11.5	$0.64	78.2%	86.5%	89.8%
High 16hi	10.8	$0.60	79.5%	87.3%	90.4%
High 8hi	12.8	$0.71	75.7%	85.0%	88.6%

Source: Bernstein estimates and analysis

PART V: HBF'S IMPACT ON THE NAND INDUSTRY AS WELL AS GPU & HBM DEMAND

HBF COULD DO FOR NAND WHAT HBM HAS DONE FOR DRAM INDUSTRY DYNAMICS

HBF could do for NAND what HBM has done for DRAM industry dynamics. HBF has the potential to create a meaningful new source of NAND demand while simultaneously reducing the number of gigabytes produced per wafer. This combination could materially improve the industry's supply-demand balance.

The demand opportunity is significant. As AI systems require larger pools of high-bandwidth, high-capacity memory, HBF could expand NAND's role within the AI memory hierarchy. Rather than simply displacing existing enterprise SSD demand, HBF could become an incremental demand driver by addressing AI inferencing workloads that require substantially more capacity than HBM can economically provide.

At the same time, HBF is highly intensive in wafer capacity. A typical HBF die may store only approximately one-quarter as many bits as a conventional TLC die, while also requiring a larger die area. Based on our analysis, each unit of HBF output could consume approximately 6.5x more effective wafer capacity than conventional NAND relative to the number of bits produced.

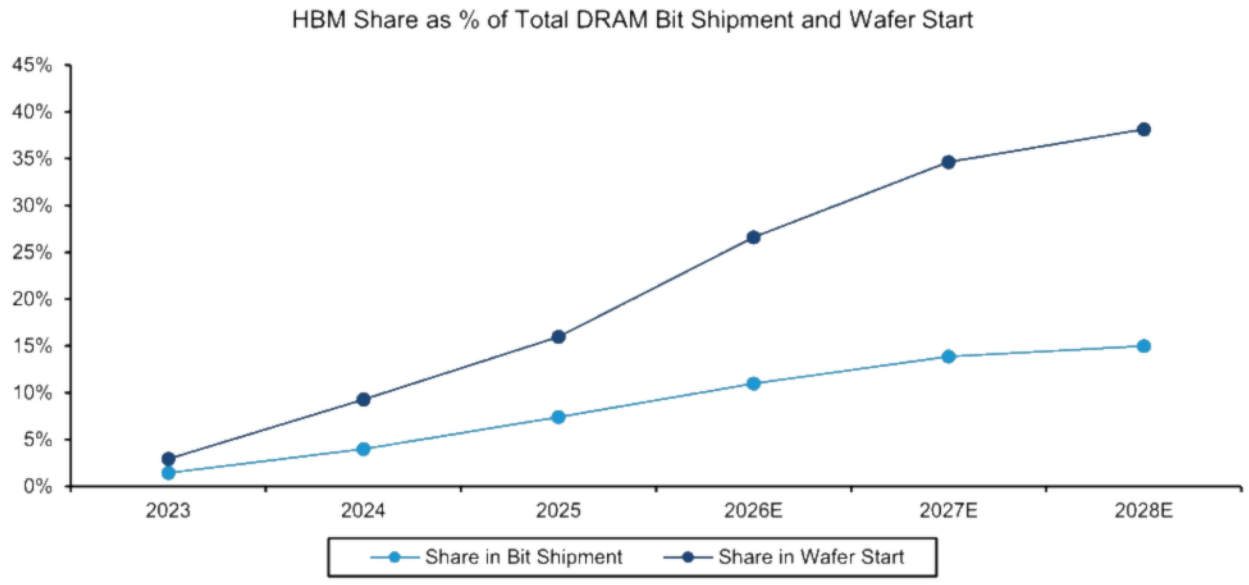
As HBF volumes scale, this large trade ratio could result in a significant reduction in industry-wide gigabytes produced per wafer. Even a relatively modest shift in wafer allocation toward HBF could therefore absorb a disproportionate amount of NAND capacity, limiting conventional NAND bit-supply growth.

In this respect, HBF could provide a structural benefit to NAND similar to the impact of HBM on the DRAM market:

- Create a new, premium AI-driven demand category
- Consume significantly more wafer capacity per bit than conventional NAND and tighten industry supply
- Provide incremental support for NAND pricing and margins

Overall, we view HBF as both a potentially attractive product opportunity and a meaningful constraint on future NAND supply. If adoption scales, the combination of incremental AI-related demand and an estimated 6.5x wafer-capacity trade ratio could materially improve NAND industry dynamics.

EXHIBIT 17: **HBM Share by Wafer has risen steadily, tightening overall DRAM supply - we believe HBF could have a similar benefit on NAND**



Source: Company reports, TrendForce, Bernstein estimates and analysis

COULD HBF CANNIBALIZE HBM?

Successful HBF commercialization could reduce HBM requirements per workload, but HBF is more likely to complement than broadly replace HBM within the AI memory hierarchy. Moreover, if HBF reduces the cost of AI, Jevons Paradox can actually lead to more aggregate HBM demand.

The clearest cannibalization risk is that HBF could reduce the amount of HBM required to accommodate a given model. Today, AI workloads may need to provision additional GPUs and HBM primarily to fit the full parameter base, even when only a portion of those weights is accessed at any one time. This is particularly inefficient for sparse MoE models, which may contain trillions of total parameters but activate only a small subset of experts for each token. By moving inactive or less frequently accessed weights into HBF, data centers could reserve HBM for the active working sets and avoid purchasing incremental HBM capacity simply to fit the complete model.

However, HBF is less likely to displace the HBM required for performance. Active weights and the hottest portions of the KV cache require the low latency, high bandwidth, and write performance of HBM. HBF is more naturally suited to read-intensive data that requires substantial capacity. As a result, HBF could reduce HBM capacity provisioned for mo it would not eliminate the need for HBM to support active computation and token generation.

The aggregate impact on HBM demand could ultimately be positive if HBF meaningfully improves AI ec Lower memory costs and better GPU utilization could reduce the cost per token, making AI viable across a broad applications. This could drive more inference workloads and wider adoption of agentic workloads. Each of these

人在草木间

would mean more use of AI concurrently, potentially offsetting the reduction in HBM required for an individual model.

HBF could therefore create a Jevons paradox for HBM: greater memory efficiency per workload may ultimately drive greater aggregate HBM consumption. For a fixed model and token volume, HBF could reduce the amount of HBM needed to store the full parameter base. However, if better token economics stimulate substantially greater AI usage, the resulting growth in token volumes and active working sets could more than offset this efficiency benefit. We therefore see HBF as a potential optimizer of HBM usage rather than a structural threat to HBM demand. It may reduce HBM content per workload, but it could also broaden the AI market and support continued growth in aggregate HBM consumption.

HBF COULD IMPROVE GPU EFFICIENCY WITHOUT NECESSARILY REDUCING AGGREGATE GPU DEMAND

HBF could reduce GPU intensity by allowing large models to run on fewer GPU/HBM pairs, improving GPU utilization, and enabling prior context to be reused rather than recomputed. Yet lower total system cost could accelerate inference adoption and increase the number of deployed systems, potentially offsetting the decline in GPUs required per workload.

HBF could reduce GPU requirements for inference workload through three distinct mechanisms: fewer GPU/HBM pair needed to fit the model, higher output from each GPU, and less compute spent recreating previously processed context. However, these same efficiencies could lower total system cost and accelerate the adoption of AI inference. The ultimate impact on aggregate GPU demand will therefore depend on whether lower GPU intensity per workload is outweighed by growth in the number of deployed systems and overall inference activity.

First, HBF could reduce the number of GPU/HBM pairs needed to provide enough memory capacity to accommodate large models. Sandisk's illustrative simulation of the 960GB Qwen3-480B-A35B model showed that one HBF-equipped GPU could provide sufficient capacity to run a model that would require a minimum of eight HBM-equipped GPUs simply to fit the complete parameter set. This is the basis for Sandisk's claim of 8x improvement in CapEx efficiency: HBF could avoid the need to purchase additional GPUs mainly for their attached HBM capacity rather than for their compute resources. This benefit could be particularly relevant for large MoE models, where the full parameter base must remain accessible even though only a small subset of experts is activated for each token.

Second, HBF could improve the productivity of the GPUs that remain in the system. In the same simulation, four HBF-equipped GPUs generated the same token output as eight HBM-equipped GPUs, supporting Sandisk's claim of 2x greater GPU efficiency. The 2x GPU efficiency is achieved because the massive capacity of on-package HBF prevents data from spilling over slow external buses, keeping the GPU cores continuously fed and eliminating idle wait times.

Third, HBF could reduce GPU compute requirements by making it more economical to store and reuse previously computed context. Rather than discarding prior KV states and consuming GPU power and compute to recreate them, we could retain selected reusable context in HBF and retrieve it for subsequent requests or interactions. With cached-token reads priced at approximately 10% of fresh compute tokens, AI systems have a strong economic incentive to preserve and reuse prior work rather than recompute it. Over time, this "token battery" model could reduce the incremental GPU compute required.

These efficiency gains could reduce GPU content per system and per token, but they would not necessarily reduce aggregate GPU demand due to Jevons Paradox. Lower memory requirements, better GPU utilization, and less recomputation would reduce total system cost and improve token economics, making advanced inference viable across a wider range of applications. Even if each inference system requires fewer GPUs, lower costs could drive substantially more system deployments, broader enterprise adoption, and greater token consumption across more use cases.

PART VI: HBF TECHNOLOGY DEEP DIVE: HOW HBF TURNS NAND INTO HIGH-BANDWIDTH MEMORY

HBF increases bandwidth through parallelism, wide data paths, and sophisticated control logic rather than by making individual NAND cells as fast as DRAM. This section dives deeper into HBF technology and how it works.

1. Physical structure: stacked NAND above a base die, closely connected to the processor

HBF combines NAND's storage density with an HBM-like stacked package and wide data paths. An HBF package can be visualized as a tower of multiple NAND core dies above a dedicated base die. Each core die provides storage capacity and the local circuitry needed to access its NAND cells, while TSVs carry signals and data between the stacked dies and the base

die. The base die serves as the stack's logic foundation, coordinating activity across the NAND dies, aggregating their output, and connecting the complete package to a compatible GPU or other processor through multiple UCIe host channels. HBF generates high bandwidth by reading many NAND resources concurrently and providing sufficient internal and processor-facing transport capacity to deliver their combined output (Exhibit 18).

2. Parallel access within and across stacked NAND dies

HBF increases bandwidth by allowing many NAND resources to read data simultaneously, both across stacked dies and within each die. The HBF specification supports 8-high and 16-high NAND stacks, but simply adding more dies does not increase bandwidth if they must take turns using the same read circuitry or data path. Each die therefore needs independently accessible planes or sub-arrays, while the full stack needs enough channels to operate multiple dies concurrently. The resulting bandwidth comes from aggregating many parallel NAND reads rather than making each NAND read as fast as DRAM.

3. CBA: bonding the NAND array to separately optimized peripheral circuitry

CBA allows each NAND array and the CMOS peripheral circuitry that operates it to be manufactured separately, optimized independently, and then bonded together. The NAND array stores the data, while the local CMOS circuitry selects cells, applies and senses the required voltages, decodes addresses, and moves the retrieved data out of the array. Each stacked NAND die effectively combines its own NAND array with its own locally bonded CMOS circuitry.

Manufacturing the array and CMOS separately gives the designer more flexibility to use capable control circuitry without sacrificing as much NAND-array density. Face-to-face bonding can also provide a dense set of short connections between the memory array and its local logic. All these make CBA-based NAND faster, and hence HBF based on CBA more likely to narrow the performance gap vs. HBM.

4. TSVs: carrying the parallel output vertically through the stack

Through-silicon vias, or TSVs, are conductive connections that pass through silicon dies. Together with connections between adjacent dies, they form vertical routes for data and other signals between the NAND core dies and the base die. HBF proposals use HBM-like TSV interconnects to support wide internal data movement. TSVs do not make NAND cells read faster. Their role is to provide enough vertical transport capacity that the output of many concurrent reads can leave the core dies without being squeezed through a narrow connection. The architecture therefore needs both independent read resources and sufficiently wide interconnects: either one alone is insufficient.

5. Base die and UCIe: delivering the stack's output to the processor

The dedicated base die coordinates the 8-high or 16-high NAND stack and combines the output of the individual dies. The local CMOS circuitry on each NAND die physically accesses and senses its own NAND cells. The base die operates one level above this: it receives requests from the processor, schedules operations across the appropriate NAND dies, and buffers and aggregates the data returned by those dies.

The base die can keep multiple operations in flight by staggering activity across the stack. For example, one NAND resource may be sensing new data while another transfers a completed read and a third prepares its next operation. This pipelining does not reduce the latency of an individual NAND read, but it reduces idle time and helps produce a more continuous stream of data once the pipeline is full.

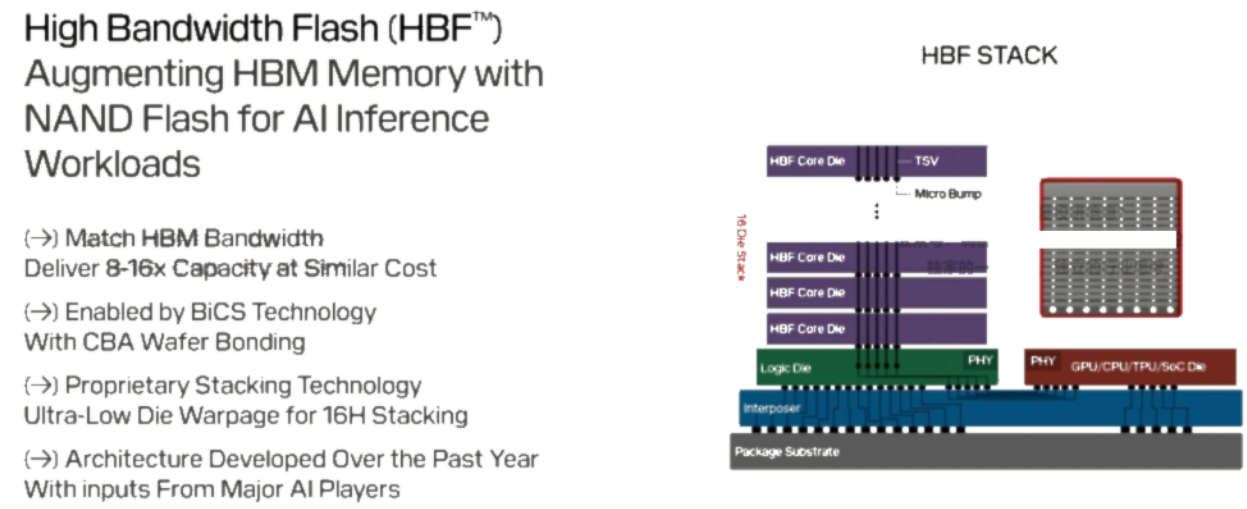
The output is organized into multiple independent UCIe host channels rather than necessarily passing through one centralized output path. The initial specification supports up to 16 host channels per stack, with each channel accessing its own independent set of NAND resources. Distributing data and requests across these channels enables the processor to draw on multiple paths simultaneously.

6. Potential use of SLC

SLC trades capacity for faster and more predictable NAND access. SLC stores one bit per cell using two voltage states, versus three bits and eight states for TLC. Distinguishing between fewer states can reduce sensing time, improve performance consistency, accelerate programming, and increase endurance. These characteristics could make SLC attractive for an HBF tier intended to deliver data quickly and consistently rather than maximize storage density. The trade-off is substantially lower usable capacity. SLC stores only one-third as many bits per cell as TLC, requiring roughly three times as much raw NAND

capacity to deliver the same usable gigabytes. We believe HBF potentially use SLC for faster access, though it has not been officially confirmed.

EXHIBIT 18: **HBF stack illustration**



Source: Sandisk HBF fact sheet

EXHIBIT 19: **HBM vs. HBF batch size**

	Per-op rate	Batch	Bandwidth
HBM	fast cells (~100 ns)	moderate	~2-3.3 TB/s (HBM4)
HBF	slow cells (~µs)	very large (must overcompensate)	0.4-3 TB/s (target)

Source: Bernstein analysis

EXHIBIT 20: **SLC NAND is capable of much lower read latency than the much more common TLC NAND.**

Approximate Read Latency (microsecond)			
SLC	**MLC**	**TLC**	**QLC**
Single-digit to 25	30-50	60-100	120-200

Source: Bernstein analysis

PART VII: HBF FACES BOTH ARCHITECTURAL LIMITATIONS AND BROADER ADOPTION CHALLENGES

HBF has many merits: it offers substantially greater capacity than HBM and much higher bandwidth than conventional NAND storage. However, it is not a perfect substitute for HBM, and it may face certain challenges in commercialization. Its NAND foundation creates structural limitations in latency, write performance, endurance, and access granularity. Separately, HBF must compete for scarce advanced-packaging resources and secure support from processor and software ecosystems.

ARCHITECTURAL LIMITATIONS: HBF RETAINS FUNDAMENTAL NAND CHARACTERISTICS

1. HBF delivers high bandwidth, but not HBM-like latency

HBF achieves high aggregate bandwidth through parallelism rather than by making individual NAND cells as fast as DRAM. NAND-array reads remain on a microsecond timescale, compared with nanoseconds for DRAM-array access. HBF compensates by reading from many independently accessible NAND resources simultaneously and moving their combined output through wide internal and processor-facing interfaces.

This architecture should work best for large, read-intensive workloads that generate sufficient concurrency to keep HBF's data paths occupied. It is less naturally suited to latency-sensitive workloads involving small, irregular, or unpredictable requests. HBF can therefore approach HBM-like aggregate bandwidth under favorable conditions without offering HBM-like responsiveness for every individual access.

2. Page-level reads can reduce useful bandwidth

HBF's page-level access creates a mismatch when the processor requires only small or scattered pieces of data. NAND retrieves data in relatively large pages rather than the fine-grained units typical of DRAM. When the requested data is stored together and most of each page is useful, HBF can transfer it efficiently. However, if the required data is scattered across many pages, HBF may need to perform multiple reads and transfer substantial amounts of data that the processor does not need.

3. Write performance and endurance remain structural constraints

HBF inherits NAND's slower write performance and finite endurance. Unlike DRAM, NAND cannot directly overwrite existing data in place. It must first erase a larger block before that block can be reprogrammed, and each program/erase cycle causes physical wear. NAND cells therefore support a finite number of write cycles before reliability begins to deteriorate.

These limitations make HBF better suited to data that is read frequently but changes infrequently. Model parameters are a natural fit because they are repeatedly accessed during inference but generally remain static. Warm, precomputed, or reusable KV-cache segments may also be suitable when they are retained and read repeatedly. By contrast, the hottest KV-cache data is continuously updated during decoding and would likely remain in HBM.

Further because HBF is packaged inside a CoWoS package, it is not serviceable, when any HBF can't endure P/E cycles and breaks. The entire accelerator chip needs to be replaced. This is distinctively different from other NAND-based technologies, as they're done in SSD format, and can be easily serviced by replacing the broken SSD with a good one. For this reason, HBF may employ the more resilient and naturally longer lifetime of SLC flash rather than cheaper, mainstream TLC for part or all of the capacity.

4. HBF requires processor and software redesign

HBF is not a drop-in addition to existing GPUs, despite its UCIe-based host interface. **A processor must be designed to recognize and communicate with HBF, while system software must determine** which data resides in HBM, which data resides in HBF, and when data should move between the two tiers. Software will also need to optimize parameter placement, caching, scheduling, and prefetching around HBF's latency and page-level access. HBF's realized performance will therefore depend not only on the memory hardware, but also on coordinated optimization across the processor and software stack.

HBF IS A COMPLEMENT TO HBM, NOT A WHOLESALE REPLACEMENT

Taken together, these architectural limitations make HBF a specialized complement to HBM rather than a general-purpose replacement. HBF offers substantially greater capacity than HBM and much higher bandwidth than conventional NAND storage, but it retains NAND's higher latency, page-level reads, slower writes, and finite endurance. These characteristics make it less suitable for latency-sensitive, write-intensive, and highly irregular workloads.

HBF instead appears best suited to well-defined, read-intensive inference use cases. HBM would retain the active, latency-sensitive, and frequently updated working set, including the hottest model parameters and active KV cache. HBF could provide a much larger capacity tier for the broader model parameter base, inactive MoE experts, and warm or reusable KV-cache data. This division of labor could expand the memory available to AI inference without requiring all data to reside in expensive HBM.

BROADER INDUSTRY CHALLENGES: PACKAGING, PRODUCT FLEXIBILITY, AND ECOSYSTEM ADOPTION

CoWoS real estate is a scarce resource. Should we give it to compute or HBM over HBF? Burdened by the round format of wafer, CoWoS is seeing an increasing challenge to support chips with a large package footprint. The cost of CoWoS is also high. This makes the real estate of CoWoS package a critical resource, and whether to allocate it to HBF over computation dies or HBM becomes debatable.

An accelerator often needs to support both training & inference. The ideal use case of HBF is to store large amount of read-only data, and with the access to the data deterministic or reasonably predictable by speculative pre-fetching. However accelerators such as GPUs from NVIDIA are often deployed for both training & inference, but pairing them with HBF will render them effective only in inference.

Finally, why no endorsement from heavyweights such as NVIDIA? Or KIOXIA? As discussed above, HBF is distinctly different from HBM. The benefits in cost, capacity and also the drawbacks in write bandwidth, endurance, etc. are so different that a careful and well-orchestrated adoption is critical to reap the benefits & avoid the drawbacks. Processor, model, algorithm, data partition, etc. all needs to be optimized. We are aware that***Google, Tenstorrent, Meta, as well as Sandisk & SK hynix are members of the HBF consortium***. However, we are surprised that NVIDIA has not yet officially endorsed HBF. Instead, NVIDIA has been actively promoting CMX (Content Memory Storage) as well as Storage Next and enlisting support from memory and other companies to address the needs of KV cache. In fact, besides Google and Tenstorrent, no other XPU company is actively involved yet to our knowledge. NVIDIA announced its future GPUs Rubin, Rubin Ultra & Feynman with Feynman supposedly in volume shipment in 2028, but none of them features HBF. This also suggests the adoption of HBF by NVIDIA, if any, will be a few years out.

Moreover, HBF will be mixed together with HBM & also packaged in CoWoS but we're also surprised that TSMC has not yet mentioned any preparation at all. We are also puzzled why KIOXIA, Sandisk's close partner, hasn't announced the support of HBF, but instead highlighted the support of CMX and Storage Next in its investor presentations.

PART VIII: SANDISK IS TRANSLATING NAND AND SYSTEMS LEADERSHIP INTO AN ECOSYSTEM-BACKED HBF ROADMAP

HBF has progressed from concept to standardization and initial silicon, with Sandisk targeting first inference-product samples in 2027 and developing a second generation for edge AI. The ecosystem now includes SK hynix, Google, Meta, and Tenstorrent, adding memory, hyperscaler, and accelerator expertise. Moreover, Sandisk's NAND IP and vertically integrated manufacturing and systems capabilities strengthen execution credibility. We also note that HBF remains outside Sandisk's financial targets, so could provide potential upside to targets in the future.

HBF ROADMAP

The HBF roadmap has progressed from concept and standardization toward initial hardware. Sandisk disclosed that the first HBF memory die has taped out and showed an image of the die; it expects its first HBF inference-product samples in 2027. The company also outlined a second-generation HBF roadmap targeting the edge, with the stated ambition of enabling 100B-parameter models on edge devices. Sandisk says this second-generation technology is being actively co-developed with multiple customers, although it did not identify them or provide a commercialization date. Importantly, HBF remains outside Sandisk's published financial projections and long-term model, so its ramp could provide potential upside to targets in the future.

STANDARDIZATION AND PARTNERS

The HBF ecosystem has expanded meaningfully over the past year. Sandisk and SK hynix announced a joint agreement to standardize HBF in August 2025. In February 2026, the HBF Technology Consortium was formed under the Open Compute Project, with Sandisk, Google, SK hynix, and Tenstorrent shown as the initial members. In August 2026, the consortium released the inaugural HBF specification and **added Meta**.

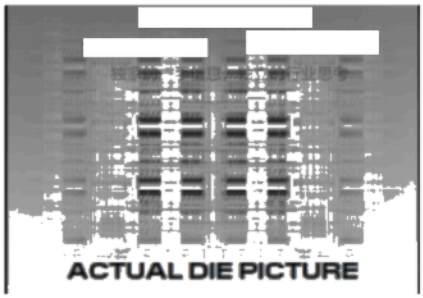
The partner set is strategically relevant. Google and Meta bring hyperscaler AI-infrastructure perspectives and potential demand-side input; **SK hynix brings leading HBM expertise** and HBF co-development credibility; and **Tenstorrent offers an inference-accelerator** and systems partner that can help shape the hardware/software architecture around HBF.

Sandisk also expanded its HBF Technical Advisory Board with David Patterson, Raja Koduri, and Jim Keller. Patterson brings computer-architecture expertise, Koduri contributes GPU and visual-compute experience, and Keller brings AI-inference silicon and system-design expertise through Tenstorrent. Together, these additions strengthen the technical ecosystem around HBF.

PLATFORM CREDIBILITY

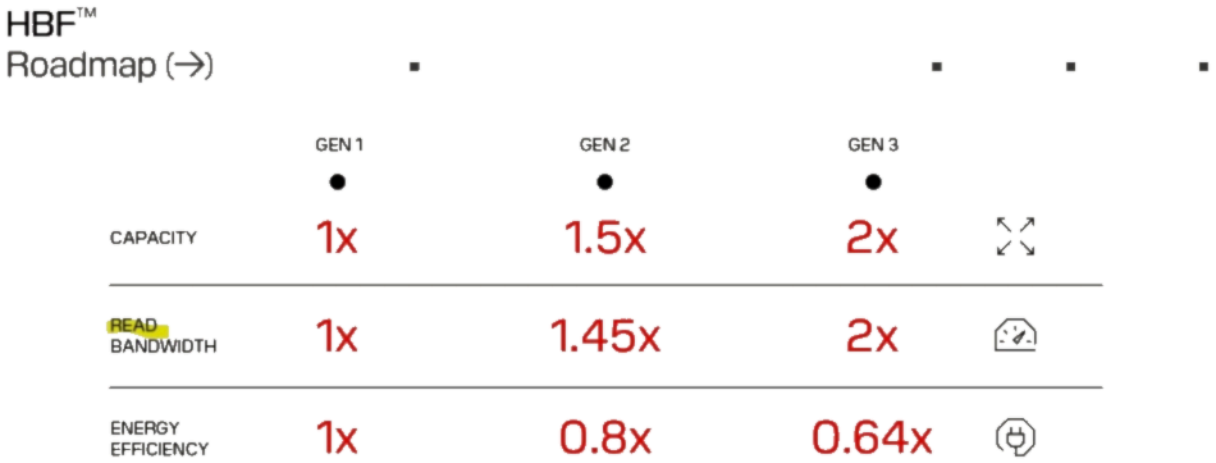
Sandisk's HBF initiative is highly credible in part because the company brings substantial NAND technology and system-level product expertise. Management emphasizes that Sandisk owns the NAND IP, front-end manufacturing through its KIOXIA joint venture, systems engineering, back-end manufacturing, and go-to-market capabilities - a vertically integrated stack that matters when introducing a new memory architecture rather than simply a new NAND node. The company also highlights a 19-generation NAND scaling engine and continued investment in NAND performance, power efficiency, density, controllers, and systems capabilities.

EXHIBIT 21: **HBF Die Taped Out**



Source: 2026 SNDK Investor Day

EXHIBIT 22: **HBF Roadmap**



Source: Company website