PLINK

A GPU-Initiated I/O Platform Exploiting NVMe Parallelism

Jaewoong Jung, Hyukyul Kang, Jaegyu Choi

Samsung Electronics

GPU storage access

GPU memory pressure brings NVMe into the data path

LLM KV-cache offload

Long contexts create pressure on GPU memory capacity.

Access granularity

Fine-grained reads demand many concurrent requests.

How many I/O warps does it take to saturate an SSD?

Baseline I/O path

BaM synchronous application path

Requesting thread	Work
1. Submit	Write a command to the SQ
2. Poll	Search the CQ for its own CID
3. Resume	Consume data, then issue the next I/O

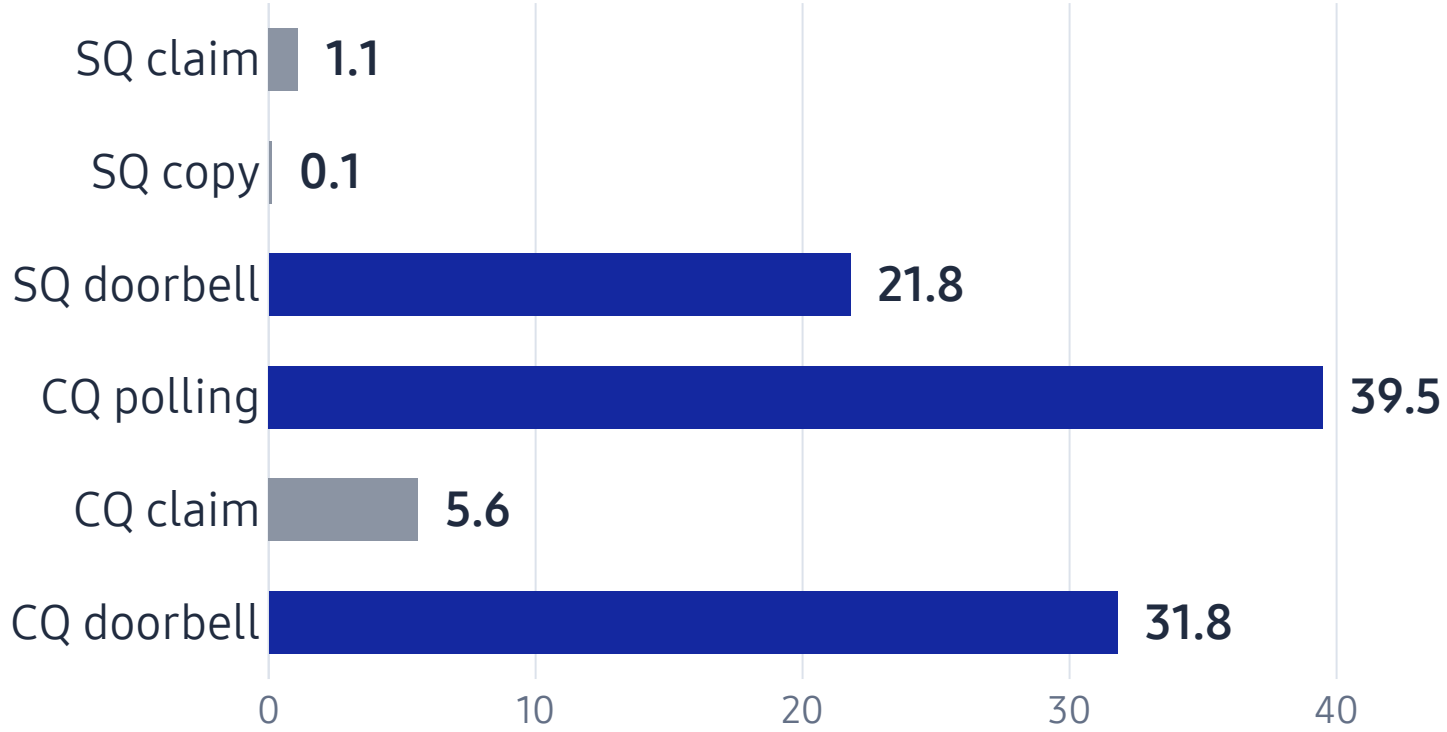
One outstanding request per thread

Threads reach polling and doorbells at different times.

CID: command ID. SQ: submission queue. CQ: completion queue.

Three phases dominate turnaround time

Share of I/O turnaround time (%)



93.1%

in CQ polling and SQ/CQ doorbells

Identify the execution patterns behind this time

Polling includes device waiting time

Baseline profile: 4 KB random reads, 64 warps / QPs, QD 1024, mean of 10 runs.

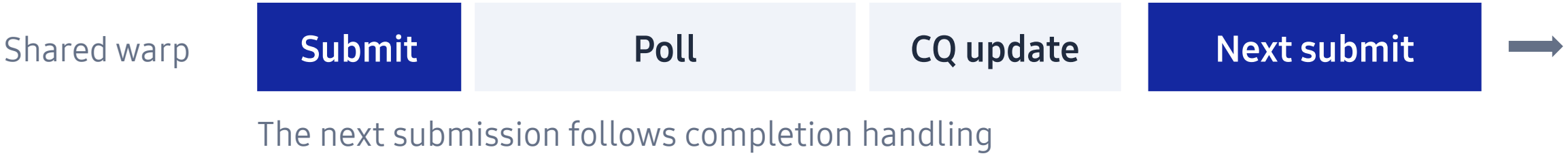
Execution patterns behind the measured time

Execution pattern		Design response
Waiting blocks the next submission	→	**SQ/CQ decoupling**
Finding a CID requires CQ searches	→	**Slot-based completion polling**
Lanes reach doorbells at different times	→	**Warp-level doorbell batching**

Each change can affect more than one measured phase

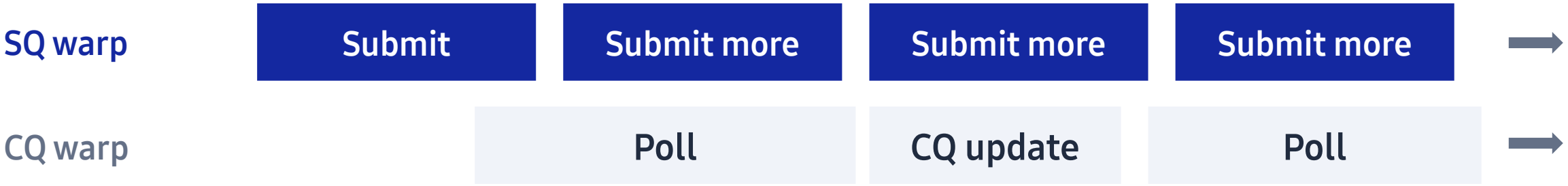
SQ/CQ decoupling

Baseline: one warp handles both queues



The next submission follows completion handling

PLINK: a dedicated warp for each queue



Slot-based completion polling

Baseline

Search for my command ID: CID 9



Scan CQ entries for the matching CID

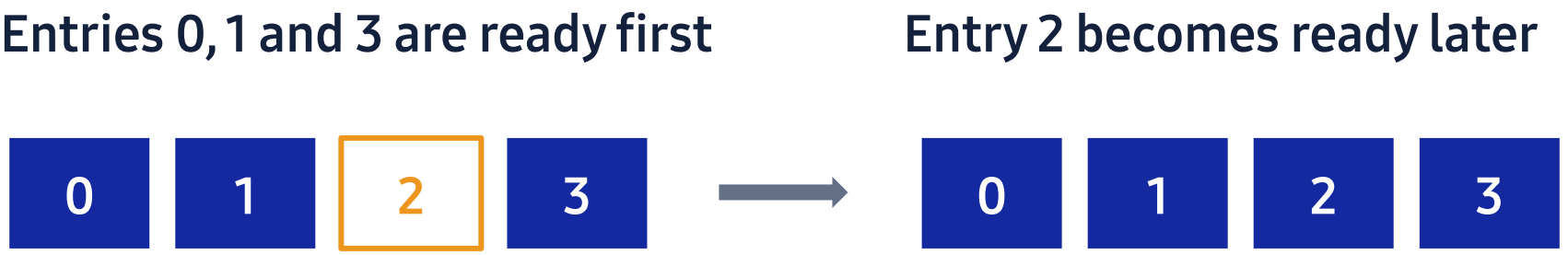
PLINK

Claim a slot, then wait for it to be valid



Read CID 7 from slot 0 and complete request 7

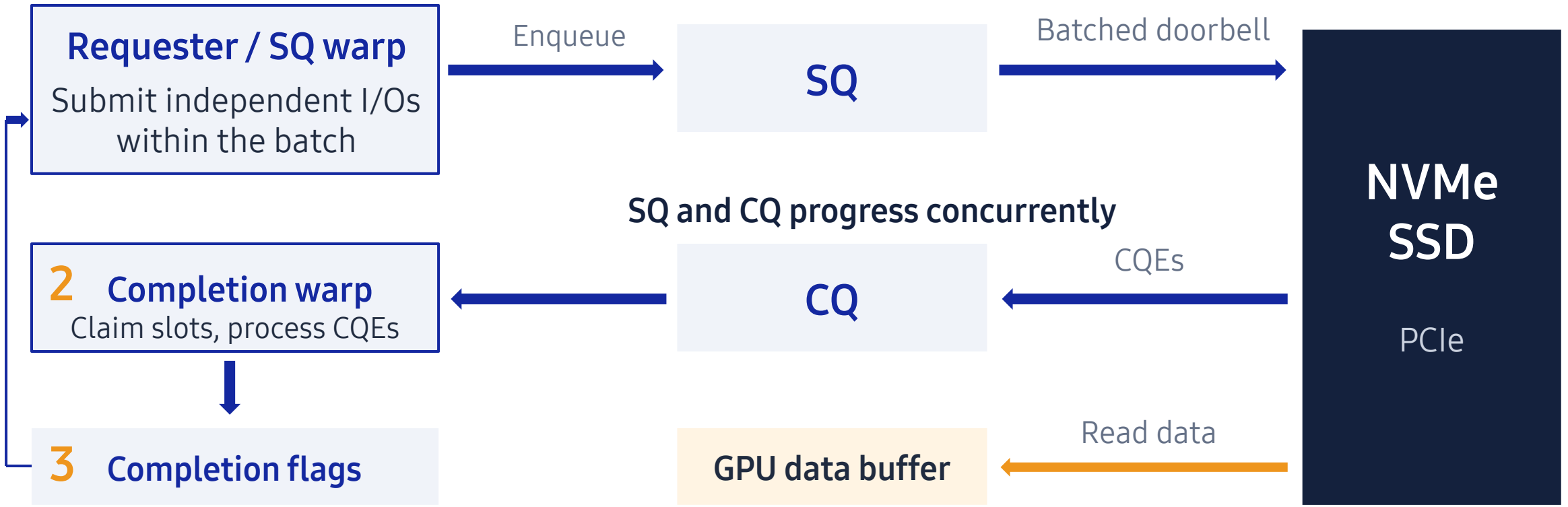
Warp-level doorbell batching



Baseline	Publish 0–1	Publish 2–3	**2 writes**
PLINK	Wait for entry 2	**Publish 0–3**	**1 write**

PLINK architecture and request lifecycle

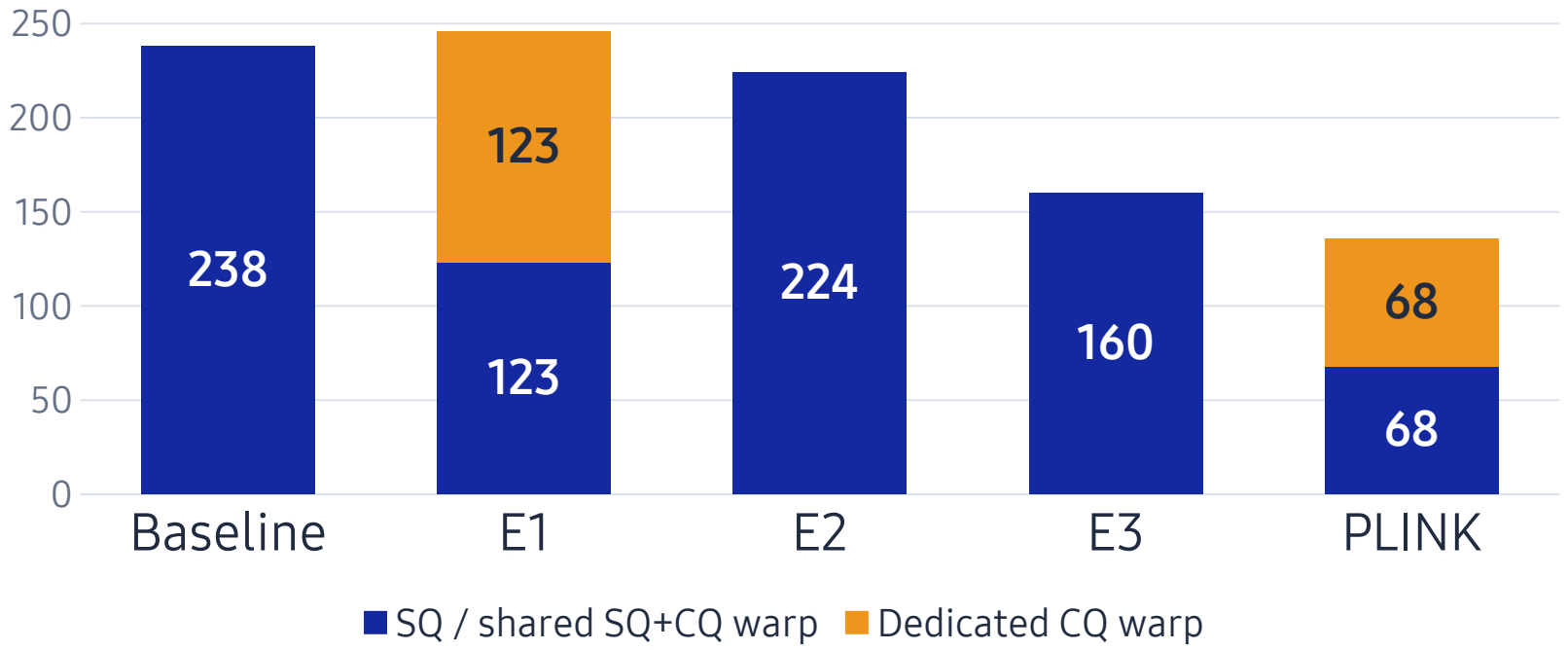
1 Issue a batch of independent requests



4 Wait for the batch, then update cache state and use the data

Decoupling alone does not reduce total warp use

Total I/O warps at saturation



E1 Decoupling

246 total warps

E2 Polling

224 total warps

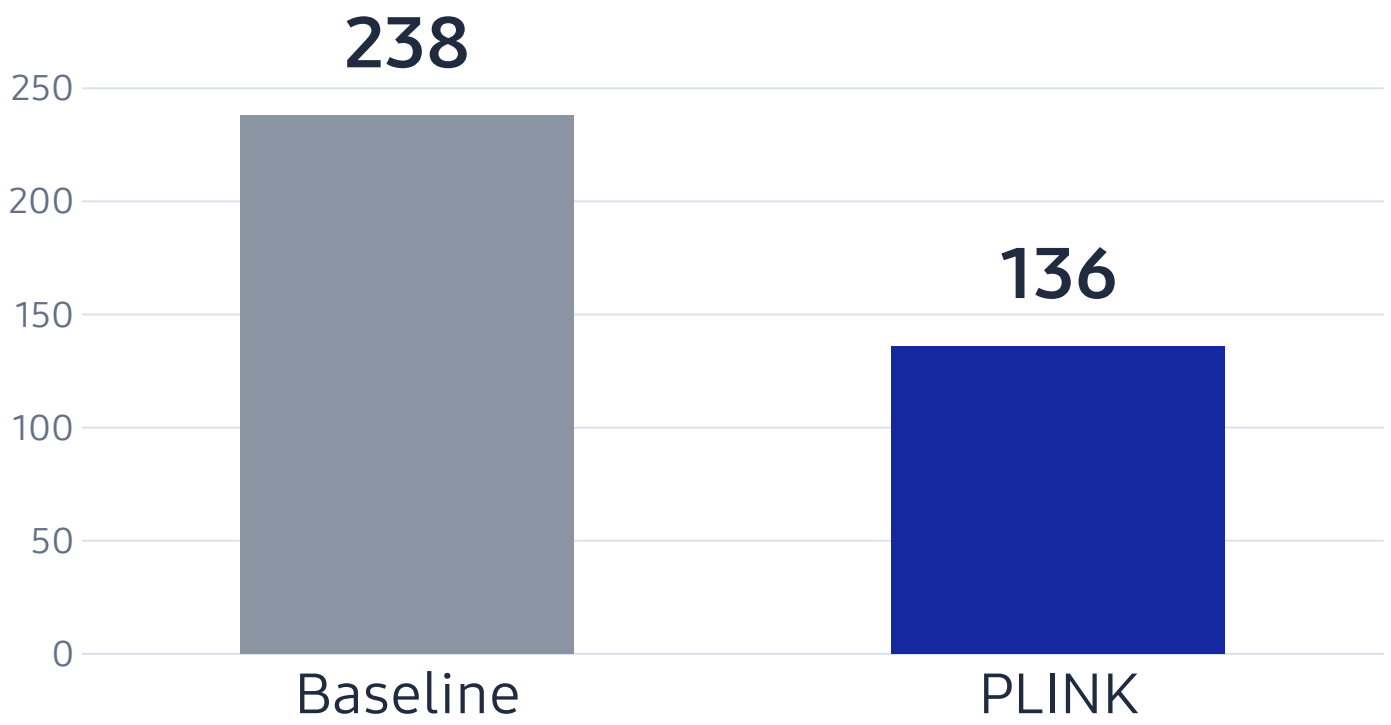
E3 Batching

160 total warps

All three together: 136 total warps

PLINK saturates the SSD with 43% fewer warps

Total I/O warps at saturation



43%

fewer total I/O warps

PLINK: 68 SQ + 68 CQ

68 queue pairs

PLINK:

A GPU-Initiated I/O Platform Exploiting NVMe Parallelism

Jaewoong Jung

jaew.jung@samsung.com

I/O backend evaluation

Component	Configuration
GPU	NVIDIA RTX PRO 6000 Blackwell
SSD	Samsung PCIe Gen 6 NVMe
Connectivity	GPU and SSD below the same PCIe switch
Workload	4 KB random reads, queue depth 1024
Resource sweep	One QP per submission warp

SSD peak: 5.65 million IOPS

Saturation: smallest warp count reaching at least 99% of that peak

Backend benchmark excludes completion notification