On-Package Memory with Universal Chiplet Interconnect Express™ (UCIe™): A Low-Power, High-Bandwidth, Low-Latency, and Low-Cost Approach

Debendra Das Sharma[1], Swadesh Choudhary[1], Peter Onufryk[1], and Rob Pelt[2]

[1]Intel Corporation

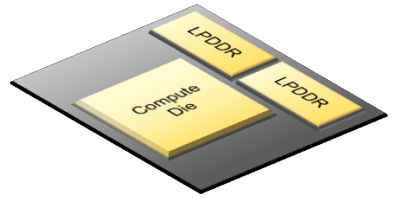
[2]AMD Corporation

Agenda

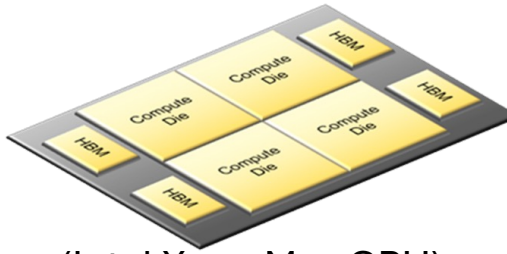
- Introduction
- Overview of UCIe
- Proposed Approaches for On-Package Memory with UCIe
- Analysis and Results
- Conclusions

On-Package Memory across Compute Segments

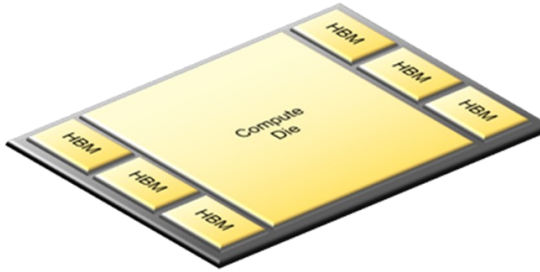
- On-Package Memory is ubiquitous across segments
- Hand-held, Laptop, PC: LPDDR
  - Motivation: low power, board area constraints, cost
  - Examples: Apple M-series, Intel Lunar Lake AI PC, Qcom Snapdragon
- AI, HPC, Server: CPU, GPU, Accelerator: HBM
  - Motivation: High bandwidth (20x b/w for same capacity over LPDDR) but 5-10x more expensive to meet bandwidth constrained applications
  - Examples: AMD MI300, AWS Trainium, Google TPU, NV GPUs, Xeon Max CPU
  - Memory still a bottleneck here: existing approaches challenged to meet the annual exponential growth
- Need to think of a power-efficient, high-bandwidth, cost-effective solution(s)



(Apple M2)



(Intel Xeon Max CPU)

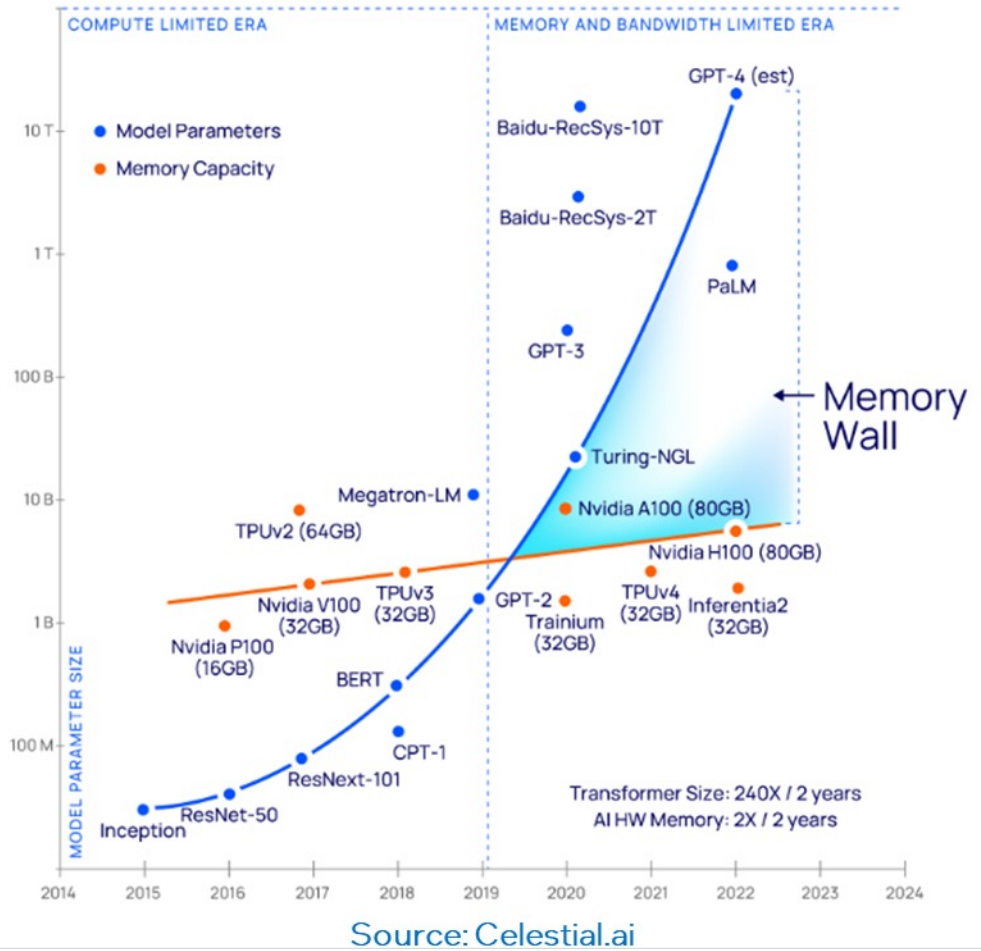


(NV H100 GPU)

On-Package Memory is widely deployed in commercial solutions across the compute landscape

Existing On-Package Memory Approaches

- Memory a bottleneck in AI applications
  - HBM challenged to deliver bandwidth and capacity within the shoreline constraints
  - LPDDR to a lesser extent in the handheld/ laptop/ PC segments
- Existing approach: Bi-directional multi-drop bus (LPDDR/ HBM)
  - Rationale: Memory process friendly – slow but wide, memory cells are bidirectional; latency advantage [avoid (de)serialization overhead]
  - Cons: Bump-inefficient bandwidth,
- All other system buses have transitioned to point-to-point decades back as the multi-drop buses don't scale in frequency and are pin-inefficient
  - Examples: PCI bus -> PCI Express link in 2003, Coherency from Front-side bus to link based (primarily PCIe PHY based) around 2005



Source: Celestial.ai

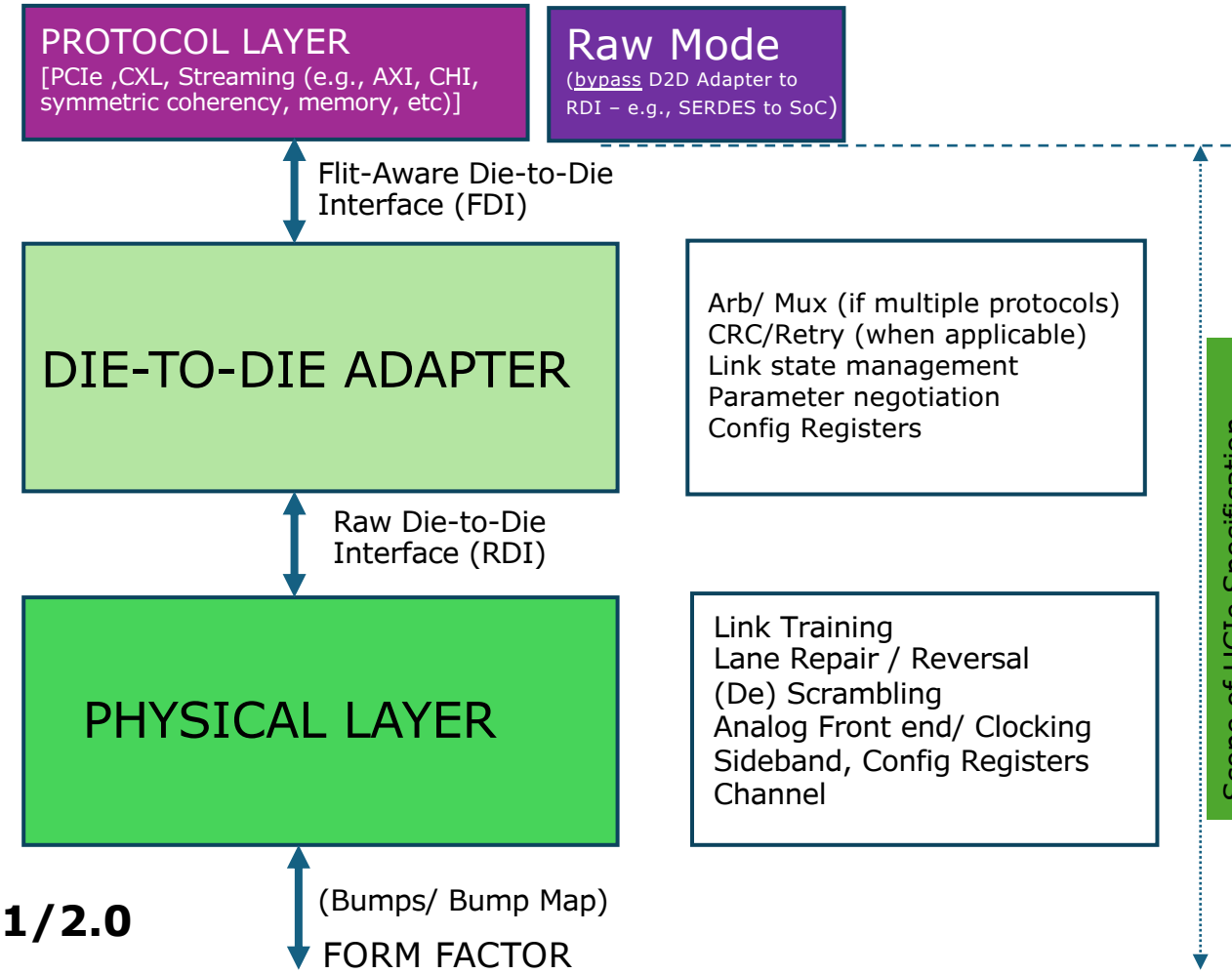
Our Proposal: On-Package memory on point-to-point unidirectional UCIe PHY for scalable bandwidth, power-efficient performance, and cost-effective solutions

Agenda

- Introduction
- Overview of UCIe
- Proposed Approaches for On-Package Memory with UCIe
- Analysis and Results
- Conclusions

UCIe 1.0 and 1.1 Specification: 2D/ 2.5D interconnect

- **Layered Approach - industry-leading KPIs**
- **Physical Layer:** Die-to-Die I/O
- **Die to Die Adapter:**
  - Reliable delivery, Multi-protocol support
- **Protocol:**
  - **CXL™/PCIe® for volume attach, plug-n-play**
    - SoC construction issues are addressed w/ CXL/PCIe
    - Usages: I/O attach, Memory, Accelerator
  - **Streaming for other protocols**
    - Scale-up (e.g., CPU/ GP-GPU/Switch from smaller dies)
- **Well defined specification**
  - Configuration register for discovery and run-time
  - Form-factor and Management
  - Compliance for interoperability
  - Plug-and-play IPs with RDI/ FDI interface
- **UCIe 3.0 backwards compatible with UCIe 1.0/1.1/2.0**

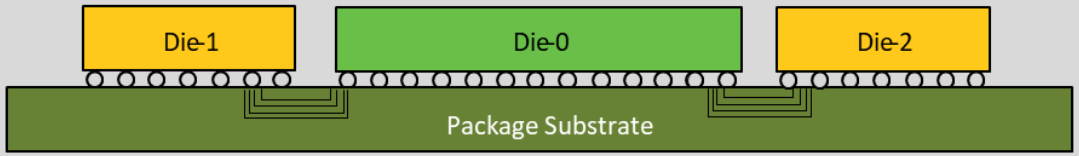


Details: D. Das Sharma et. al., "Universal Chiplet Interconnect Express (UCIe)®: An Open Industry Standard for Innovations with Chiplets at Package Level", invited paper, IEEE Transactions on Components, Packaging, and Manufacturing Technology, Oct 2022.

D. Das Sharma and T. Coughlin, "Universal Chiplet Interconnect Express: An Open Industry Standard for Memory and Storage Applications", IEEE Computer, Jan 2024

D. Das Sharma, "Universal Chiplet Interconnect Express (UCIe)®: An Open Industry Standard for Innovations with Chiplets at Package Level", IEEE Micro Special Issue, Mar-Apr 2023

UCIe Planar: Supports Standard and Advanced Packages

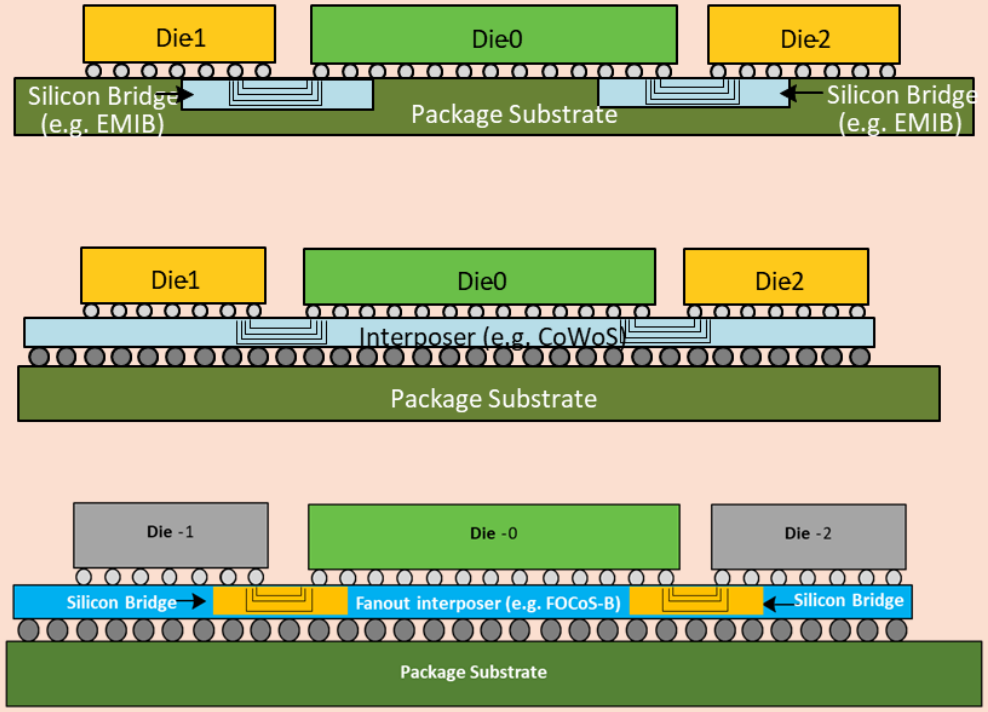


(Standard Package)

Standard Package: 2D – cost effective, longer distance

Advanced Package: 2.5D – power-efficient, high bandwidth density

Dies can be manufactured anywhere and assembled anywhere – can mix 2D and 2.5D in same package: Flexibility for SoC designer



(Multiple Advanced Package Options)

UCIe Key Metrics

Metrics	UCIe-2D[1]	UCIe-2.5D[1]	UCIe-3D[1]
Data Rate (GT/s)	4, 8, 12, 16, 24, 32		<= 4G
Width (per direction)	16	64	80
Bump Pitch (µm)	100-130	25-55	<=1 - 9
Channel Reach	25 mm	2 mm	~0 mm (Hybrid Bonding)
B/W Shoreline (GB/s/mm)	28-224	165-1317	N/A (areal only)
B/W Density (GB/s/mm^2)	22 – 125	188-1350	4000 (9µ) – 300,000 (1µ)
	B/W depends on frequency. UCIe 2D @ 110 µm; 2.5D @ 45 µm		At 4G
Power Efficiency (pJ/b)	0.5 (<=16G) / 0.6 (>16G)	0.25 (<=16G) / 0.3 (>16G)	0.05 (9µ) - 0.01 (1 µ)
Dynamic Power Savings	<1ns entry/exit with 85%+ power savings		
Latency (round-trip)	2ns		< 1ns

[[1]: From UCIe 1.0 and 2.0 Specifications] [Bump Pitch reduction by x increases the areal bandwidth density by $1/x^2$]

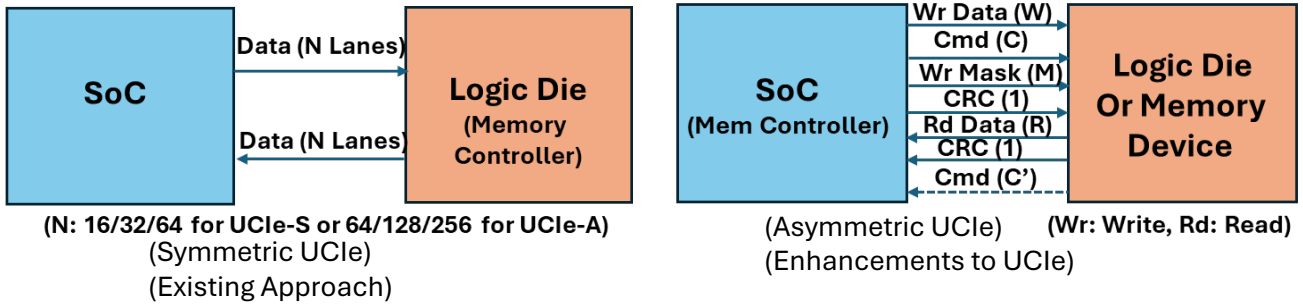
Industry-leading KPIs. UCIe continues to be bump-limited. UCIe 3.0 doubles the planar data rate to 64 GT/s!

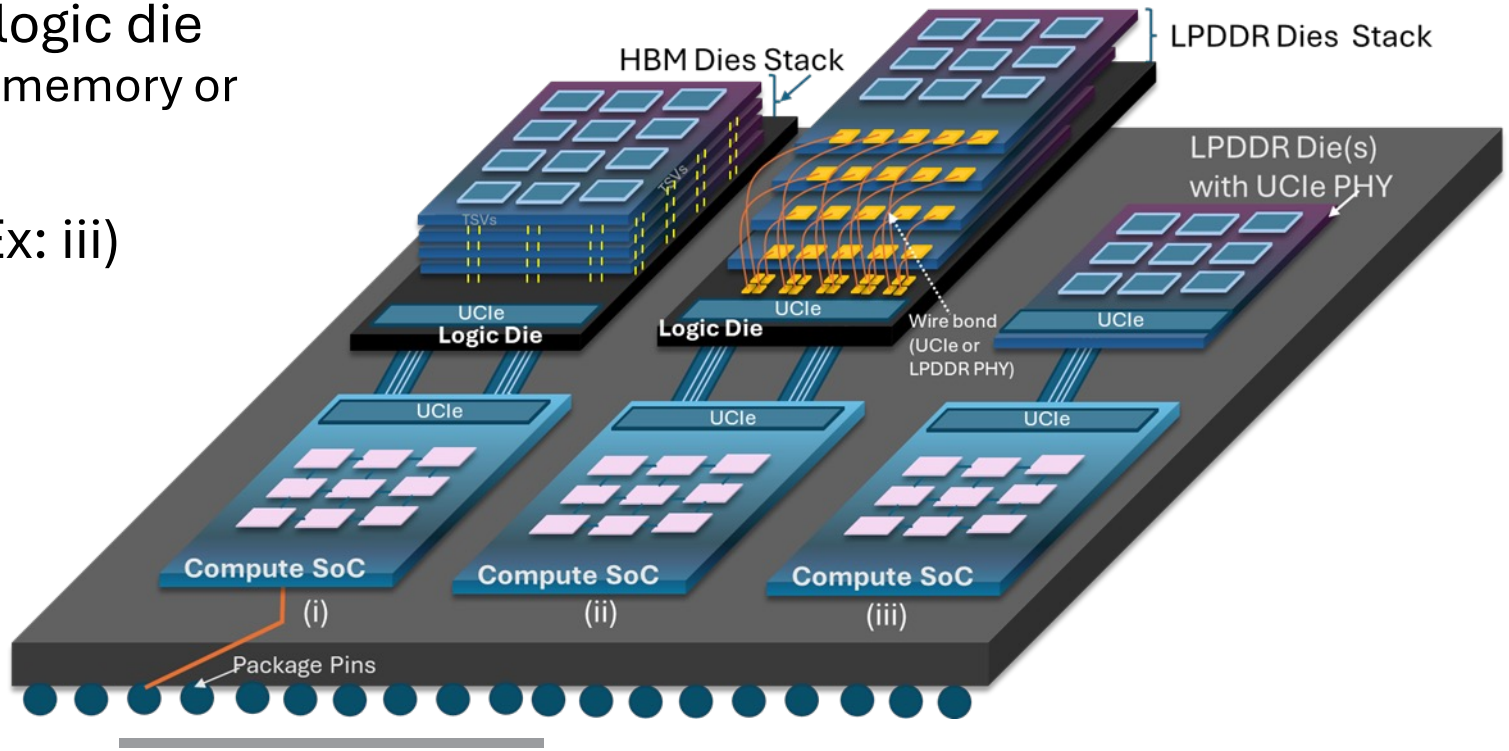
Agenda

- Introduction
- Overview of UCIe
- Proposed Approaches for On-Package Memory with UCIe
- Analysis and Results
- Conclusions

Proposed Approach

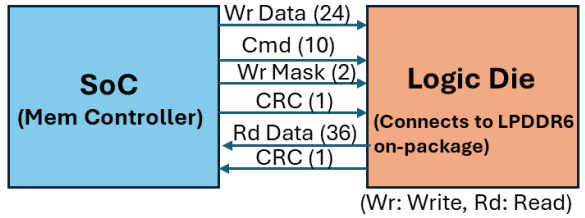
- UCIe PHY and D2D adapter used
  - Defined symmetric widths (x16/64) and asymmetric widths (proposed enhancements)
- Two broad categories:
  - Memory connected through a logic die
    - Ex: Logic die connects to HBM memory or LPDDR6 dies (Ex: i and ii)
    - Likely initial intercept
  - Memory die has native UCIe (Ex: iii)
- Protocols Mapped:
  - CXL (w/ optimizations)
  - CHI
  - LPDDR 6 (with timing)
  - HBM 4 (with timing)



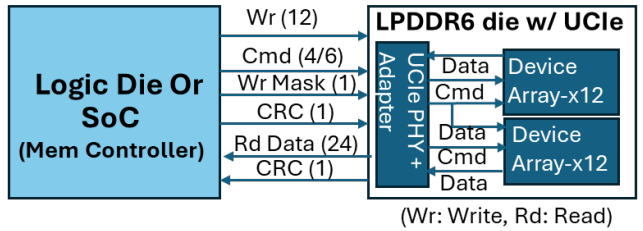


LPDDR6 Protocol Mapped on Asymmetric UCIe

- Connection to SoC: (a) logic die (b) LPDDR6 die with UCIe PHY
- UCIe interface optimized for Read-Write ratios of
  - (a) 3:2: 74 wires – closely mimics a x32 UCIe by adding one row of bumps
  - (b) 2:1: 43 or 45 wires – needs a new layout
  - Can choose other ratios for read-write
- LPDDR6 protocol and timing maintained as-is
  - All wires run at same frequency on UCIe
  - Signal list of LP6 and its mapping to UCIe shown here



(a) On-Package LPDDR6 through Logic Die



(b) LPDDR6 die with native UCIe PHY

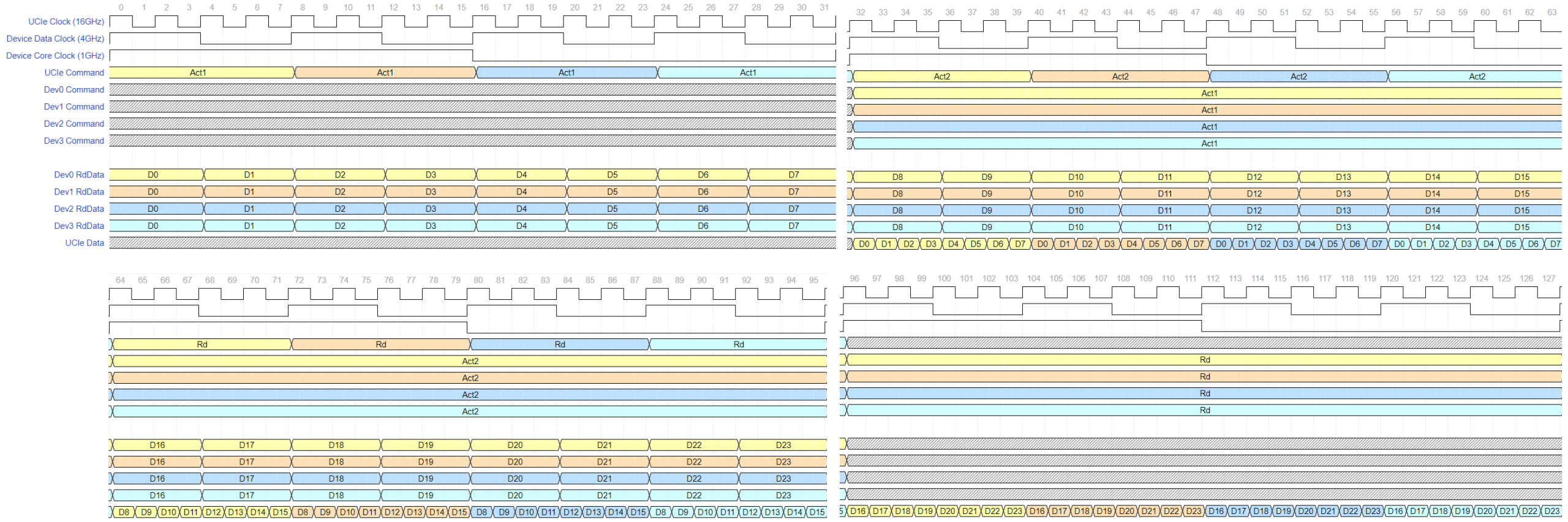
Signal Name	Direction	Frequency	Count
Command / Address (CA)	SoC->Mem	1/2	8
Chip Select (CS)	SoC->Mem	1/2	2
Wr Clk (wclk_t, wclk_c)	SoC->Mem	1/2	4
Rd Clk (RDQS_t, RDQS_c)	Mem->SoC	1/2	4
Clock (ck_t, ck_c)	SoC->Mem	1/4	4
Data	Bi-directional	1	24
Total (46)			46

(c) Signal List for 2 sub-channel x24 LPDDR6

Signal Name	SoC-> Mem	Mem -> SoC
Command/ Address	4	0
Data	12	24
Wr Mask	1	0
CRC	1	1
UCIe Data Total	18	25
UCIe Clock, Track, Valid	2, 1, 1	2, 1,1
Total = 51	22	29
64B transfer (UI)	48	24

(d) LPDDR6 signal mapping on UCIe

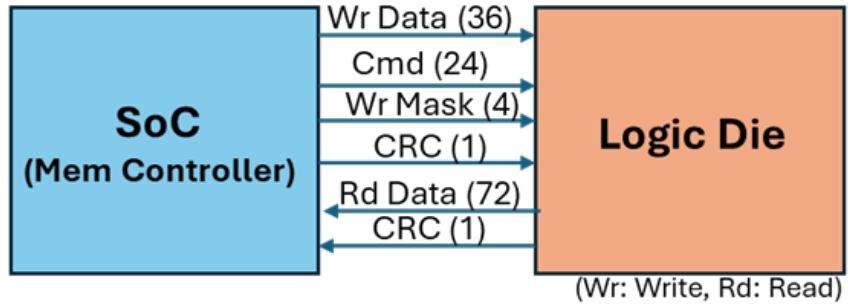
Example: UCIe to Logic Die hosting 4 LPDDR6 Devices



Time multiplexing example for Reads at 8-bit granularity showing pipelining of Activate and Read Commands. Four LPDDR6 devices are aggregated behind the logic die. Each color represents the command or data for a different x12 LPDDR6 device with a burst length of 24. Each of the 4 sub-figures is showing 32 clocks of the 16GHz clock which is used for the 32GT/s data rate over UCIe. Each sub-figure is a continuation in time, as indicated by the cycle number, from the previous one.

HBM3/4 Mapping to Asymmetric UCIe

- Memory Controller in SoC
- SoC connects to logic die using UCIe
  - HBM3/4 signals mapped as-is
  - Read-Write ration 2:1 with 138 signals
- Logic die connects to an HBM Stack
  - UCIe frequency expected to be a multiple of the HBM stack frequency (of 1-2 GT/s)



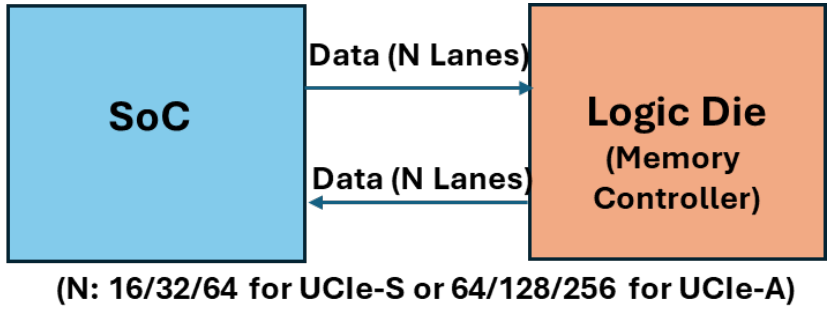
(a. HBM3/4 protocol on UCIe)

Signal Name	SoC->Logic	Logic->SoC
Command	24	0
DRAM Data, Wr Mask	36, 4	72, 0
CRC	1	1
Clk,Track,Valid	2,1,1	2,1,1
Total (Data)	69 (65)	77 (73)
Cache transfer (UI)	16	8

(b. HBM3/4 Signal Mapping on UCIe)

CHI Mapped on Symmetric UCIe

- Memory controller on logic die
- SoC – Logic Die: CHI on UCIe (symmetric)
- CHI Mapping:
  - 256B containers of CHI mapped to a 256B UCIe latency-optimized Flit
  - CHI Format X: 12 x 20 B granules for memory access; Rest 16B are used for Link and Protocol Headers (CRC, FEC, Credits etc)



(N: 16/32/64 for UCIe-S or 64/128/256 for UCIe-A)

	0 1	2 3	4 5	6 7	8 9	10 11	12 13	14 15
Byte 0	LinkHdr0,1	G0 Bytes 0 through 13						
Byte 16	G0 Bytes 14 through 19			G1 Bytes 0 through 9				
Byte 32	G1 Bytes 10 through 19					G2 Bytes 0 through 5		
Byte 48	G2 Bytes 6 through 19							ProtHdr0,1
Byte 64	ProtHdr2,3	G3 Bytes 0 through 13						
Byte 80	G3 Bytes 14 through 19			G4 Bytes 0 through 9				
Byte 96	G4 Bytes 10 through 19					G5 Bytes 0 through 5		
Byte 112	G5 Bytes 6 through 19							LinkHdr2,3
Byte 128	ProtHdr4,5	G6 Bytes 0 through 13						
Byte 144	G6 Bytes 14 through 19			G7 Bytes 0 through 9				
Byte 160	G7 Bytes 10 through 19					G8 Bytes 0 through 5		
Byte 176	G8 Bytes 6 through 19							ProtHdr6,7
Byte 192	ProtHdr8,9	G9 Bytes 0 through 13						
Byte 208	G9 Bytes 14 through 19			G10 Bytes 0 through 9				
Byte 224	G10 Bytes 10 through 19					G11 Bytes 0 through 5		
Byte 240	G11 Bytes 6 through 19							LinkHdr4,5

CXL.Mem on Symmetric UCIe

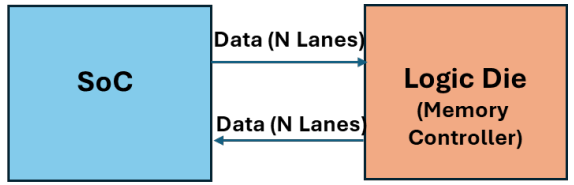
- Memory controller on Logic Die; SoC – Logic Die is CXL.Mem on UCIe
- Two Flavors: CXL.Mem as defined; Optimized CXL.Mem (proposed)
  - Optimized packs more headers in a slot by reducing some fields due to on-package
- Standard UCIe widths and 256B Flit

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Byte 0	HDR - 2B		Slot 0 - 14B H-Slot													
Byte 16	Slot 1 - 16B (G-Slot)															
Byte 32	Slot 2 - 16B (G-Slot)															
Byte 48	Slot 3 - 16B (G-Slot)															
Byte 64	Slot 4 - 16B (G-Slot)															
Byte 80	Slot 5 - 16B (G-Slot)															
Byte 96	Slot 6 - 16B (G-Slot)															
Byte 112	Slot 7 - 16B (G-Slot)															
Byte 128	Slot 8 - 16B (G Slot)															
Byte 144	Slot 9 - 16B (G-Slot)															
Byte 160	Slot 10 - 16B (G-Slot)															
Byte 176	Slot 11 - 16B (G-Slot)															
Byte 192	Slot 12 - 16B (G-Slot)															
Byte 208	Slot 13 - 16B (G-Slot)															
Byte 224	Slot 14 - 16B (G-Slot)															
Byte 240	Reserved - 10B										Credit-2B		CRC0-2B		CRC1-2B	

(CXL.Mem on UCIe)

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Byte 0	Slot 0 - 16B (G-Slot)															
Byte 16	Slot 1 - 16B (G-Slot)															
Byte 32	Slot 2 - 16B (G-Slot)															
Byte 48	Slot 3 - 16B (G-Slot)															
Byte 64	Slot 4 - 16B (G-Slot)															
Byte 80	Slot 5 - 16B (G-Slot)															
Byte 96	Slot 6 - 16B (G-Slot)															
Byte 112	Slot 7 - 16B (G-Slot)															
Byte 128	Slot 8 - 16B (G Slot)															
Byte 144	Slot 9 - 16B (G-Slot)															
Byte 160	Slot 10 - 16B (G-Slot)															
Byte 176	Slot 11 - 16B (G-Slot)															
Byte 192	Slot 12 - 16B (G-Slot)															
Byte 208	Slot 13 - 16B (G-Slot)															
Byte 224	Slot 14 - 16B (G-Slot)															
Byte 240	Slot 15 - HS Slot (10B)											HDR (2B)		Credit 2B		CRC (2B)

(Optimized CXL.Mem on UCIe)



(N: 16/32/64 for UCIe-S or 64/128/256 for UCIe-A)

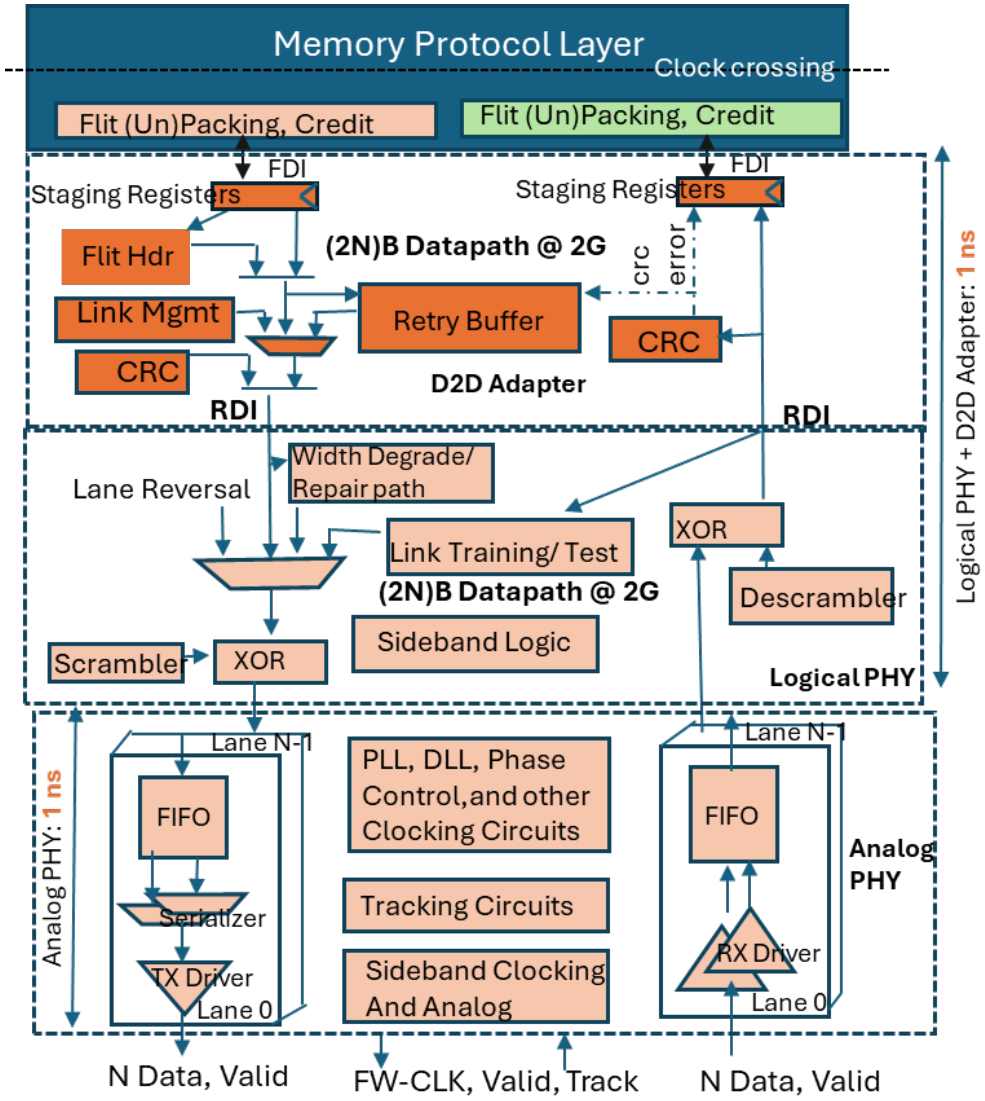
Field Name	SoC ->Mem Req (Rd/Wr)		Mem -> SoC Resp (Data, Cmpl)	
	Unopt	Opt	Unopt	Opt
Cmd	4	3	3	3
Meta Data	7	4	4	4
Devload	0	0	2	0
Tag	16	8	16	8
Address	46	46	0	0
Poison	1	1	1	1
Total	74	62	26	16

Agenda

- Introduction
- Overview of UCIe
- Proposed Approaches for On-Package Memory with UCIe
- Analysis and Results
- Conclusions

Micro-Architecture

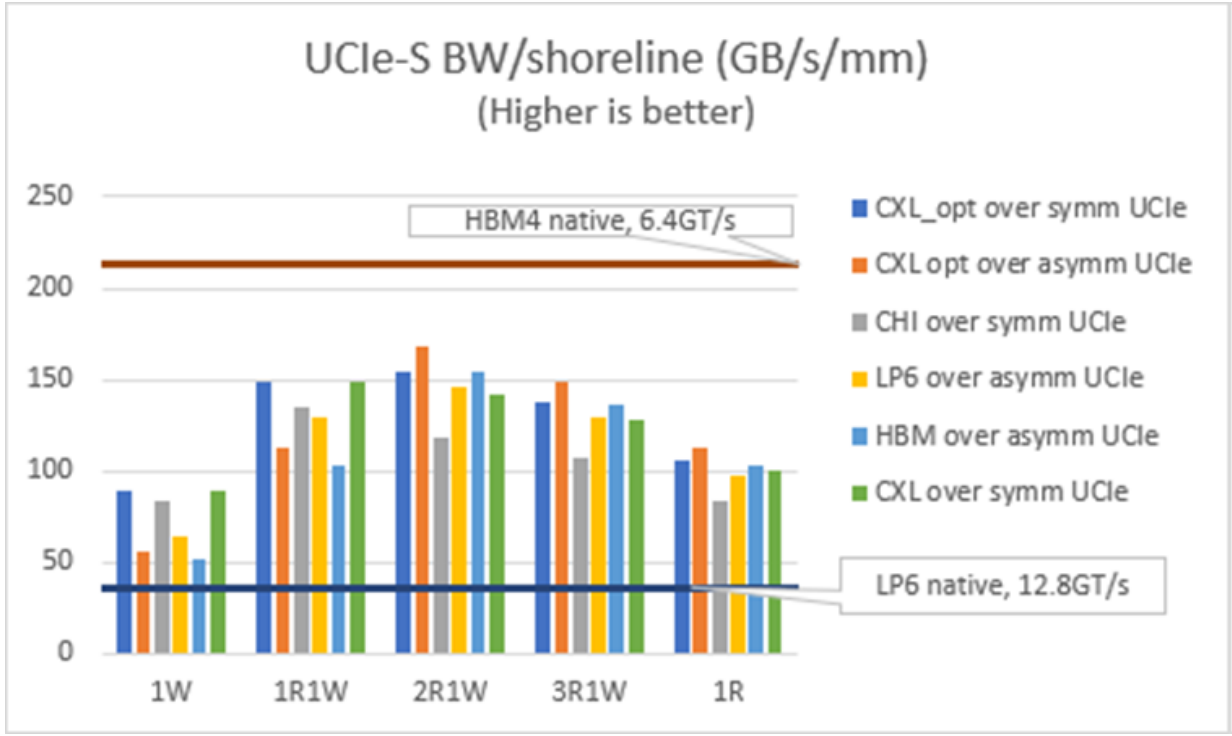
- Symmetric UCIe at 32G shown here with internal data path width of 2G
  - Asymmetric similar except the widths on the two directions are different
  - Other UCIe frequencies would be similar
- Analog PHY: Drivers (Tx/Rx), clock, track, sideband, FIFO to frequency transition (e.g., 32G to 2G)
- Logical PHY: 1 ex-or on critical path (de)scrambling – (de)scrambler values pre-calculated
- D2D adapter: CRC (5 levels of logic), replay, Flit Header generation
- Flit Packing and Unpacking Logic
- Round-trip Latency: 3 ns
  - Analog: 1 ns, Logical PHY: 1ns, Flit Pack/Unpack: 1ns – one 2GHz flop at each interface crossing)
- Existing LPDDR5 measured: 7.5ns; HBM3 measured: 6ns RT

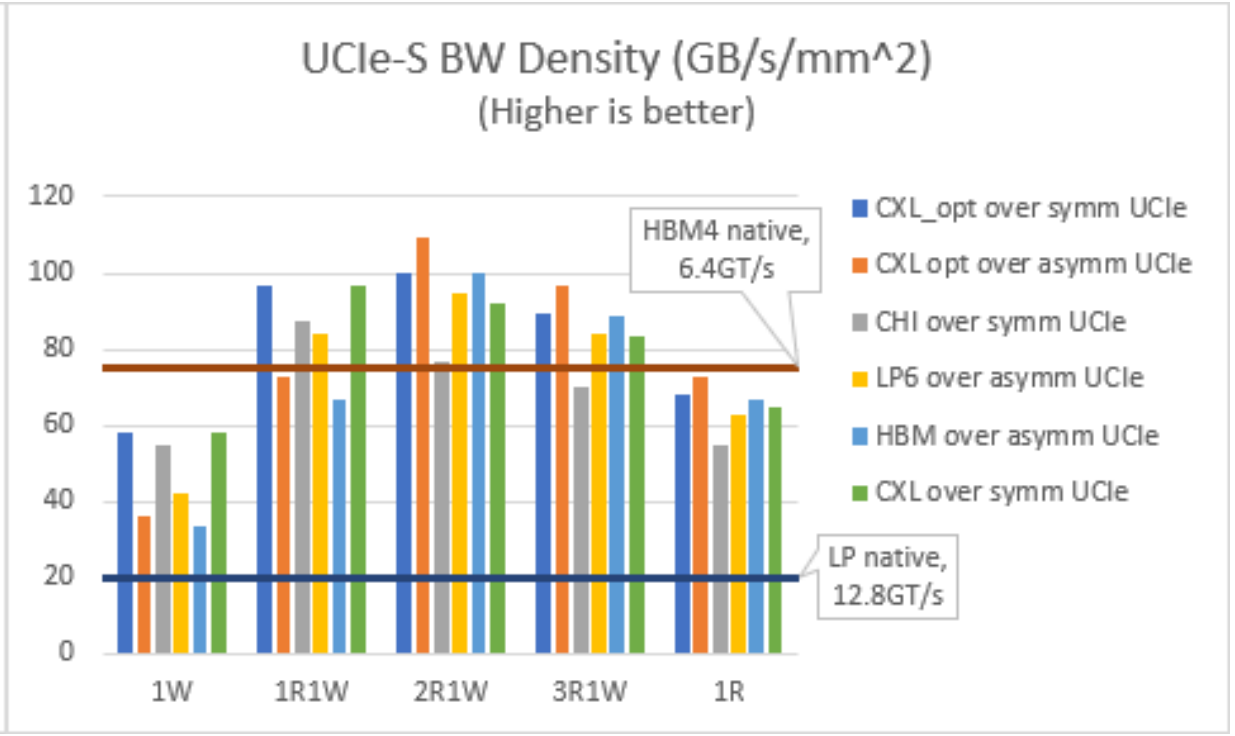


Bandwidth Density and Power Efficiency Evaluation

- UCIe is bump-limited – so circuits fit within bump area at 32G
  - UCIe-S: 1.143 mm (shoreline) x 1.54 mm (depth) for x32 @ 110um
  - UCIe-A: 0.389 mm x 1.585 mm for x64 @ 55 um
- LPDDR6 and HBM4 also assumed (optimistic) to be also bump-limited
  - Area and power efficiency assumed same as from prior gens
  - LP5: 128 DQ @ 9.6: 5.8 x 1.75 – LP6 projected by multiplying 12.8G/ 9.6 G. Power: 2.8 pJ/b measured LP5
  - HBM4: 2048 DQ @ 6.4G, 45-55 um: 8 x 2.5. Using HBM3 power: 0.9 pJ/b measured
- Only data transferred in DQ considered "payload"
  - address/ command/ ecc/crc/credit/ header/etc are overheads
  - Reserved lanes are overhead but for power assumed turned off
  - For HBM and LPDDR assuming 0 overhead for bus turn-around, scheduling
- Traffic mix: x reads and y writes ($xRyW$) (1W, 1R1W, 2R1W, 3R1W, 1W)
  - Header: $(x + y)$ request (SoC -> Mem), $(x + y)$ response (Mem -> SoC) [0 for HBM/LPDDR/direct connect to memory]
  - Data: y cache line SoC -> Mem, x cache line Mem -> SoC : total bits: 512.$(x + y)$
  - Bandwidth density is actual cache data transferred divided by shore-line (bump area) multiplied by raw b/w density
- Power: Lanes grouped by function and direction independently (e.g., DQ + Wr Mask, Cmd, CRC) – clock gated when not used with 85% power savings

Bandwidth Density Comparison (UCIe-Standard)





UCIe bandwidth density changes significantly according to payload reflecting the nature of unidirectional link

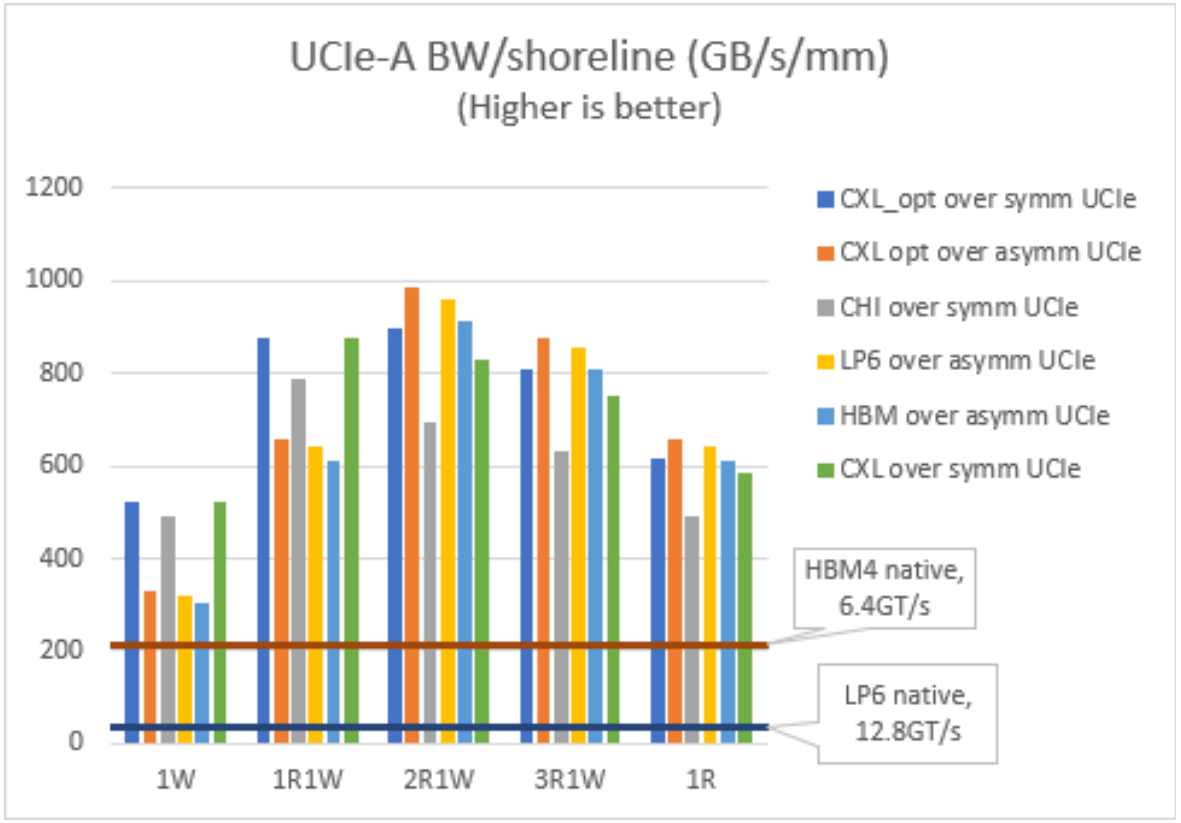
LP and HBM are flat since we assumed no bus turn around time overhead

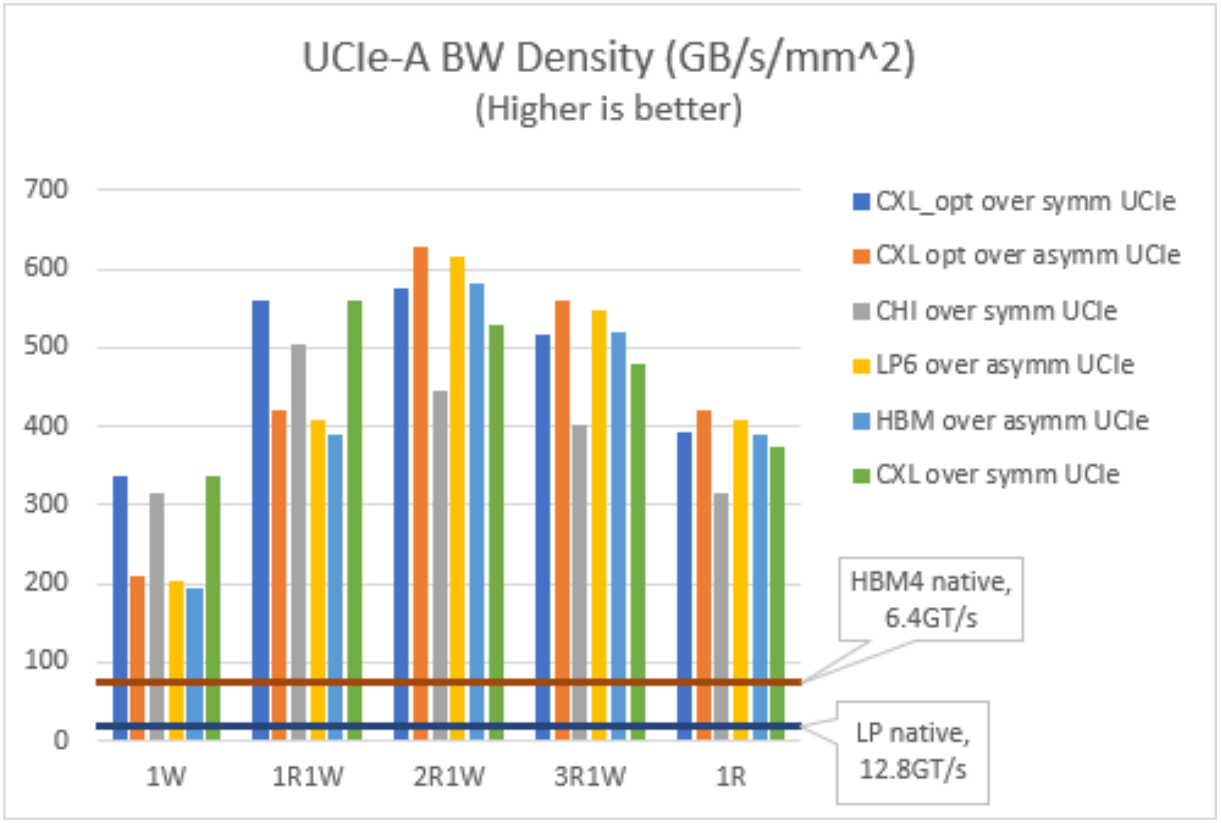
Protocol choice has an impact on the bandwidth density – CXL Optimized does best among UCIe mappings as header overhead is least (20B granule in CHI causes its inefficiency – can have a simple logic gasket to convert to optimized CXL.Mem mappings / packing and get the advantage). Asymmetric UCIe does slightly better over asymmetric as it is optimized for higher read ratios

UCIe-S approaches do better than LP6 (and HBM4 for area in many cases) due to higher data rate, even with unidir disadvantage

UCIe-S does worse than HBM4 for linear density as the latter is advanced packaging (so more bumps in same shore line) – even there the difference can be overcome with the latest UCIe 3.0 64 GT/s

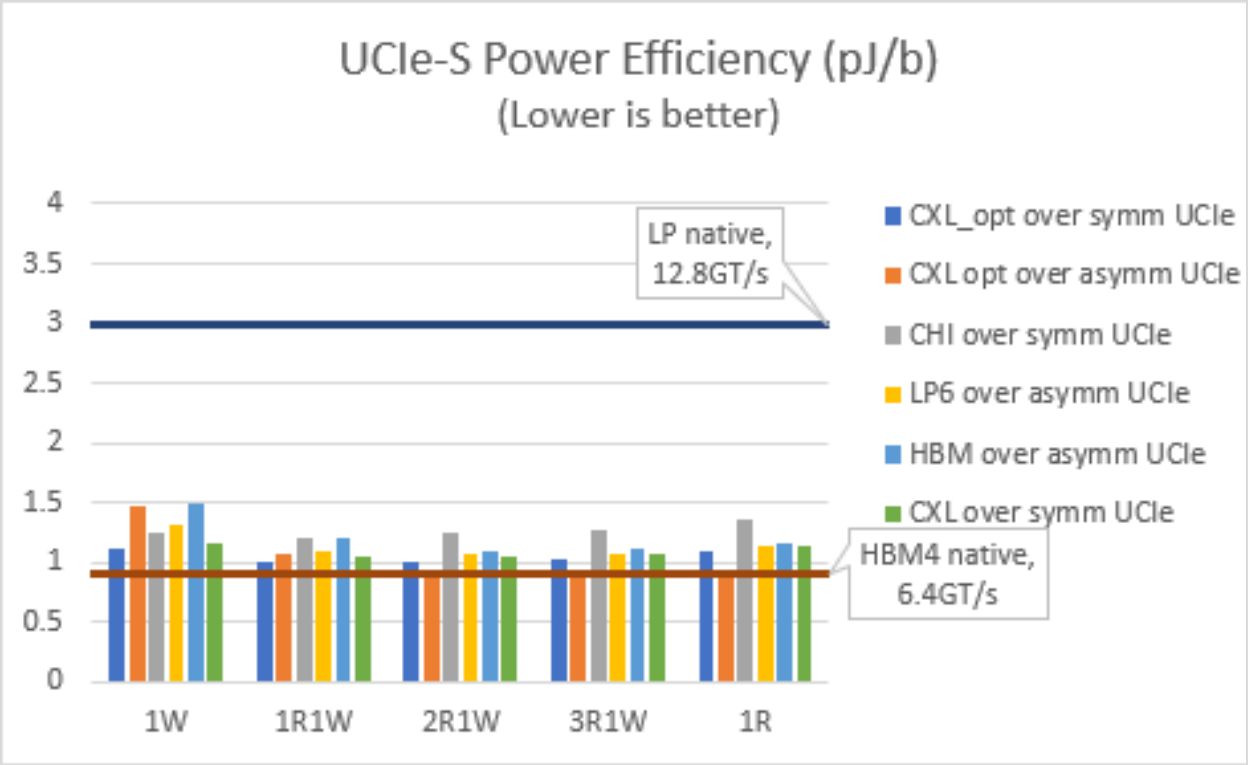
Bandwidth Density Comparison (UCIe-Advanced)

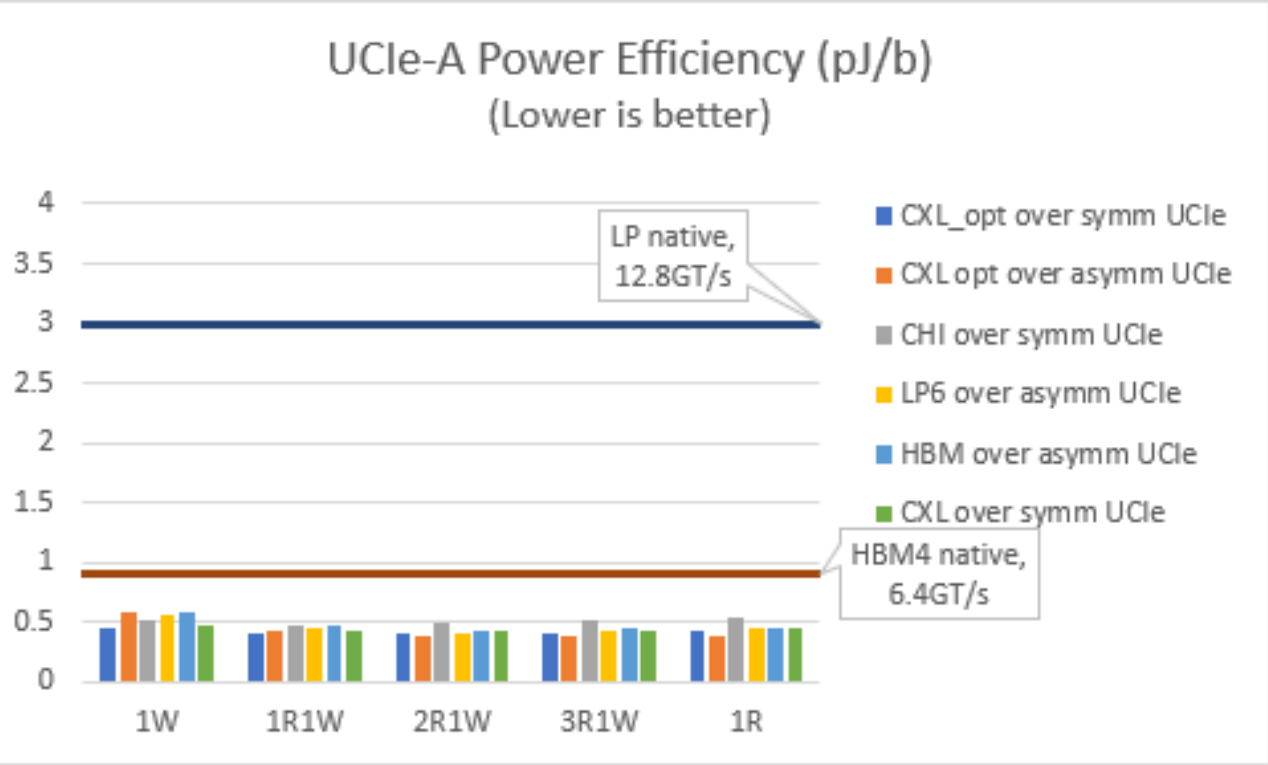




UCIe-A outperforms HBM4 significantly despite the unidirectionality disadvantage
Unidirectional => Higher Frequency / less wires => Better Performance
Expected to increase 2x with 64GT/s. BW scales as bump pitches reduce (circuits are bump-limited)

Power Comparison





Idle power consumption in unused lanes more prominent in asymmetric traffic

Power Efficiency is comparable with UCIe-S and better with UCIe-A

Conclusions

- UCIe offers a good path for providing bandwidth improvements on-package
- Starting point could be the HBM stack with UCIe connecting the SoC to Logic Die (especially as logic dies are getting manufactured using logic process technology lately)
- Approaches to continue increasing bandwidth:
  - Frequency upgrade in short term (64 GT/s)
  - Bump Pitch reduction with UCIe-A in mid-term
  - UCIe-3D with 9-<1 um bump pitch (300 TB/s/mm^2 bandwidth) in long run
- Challenges: reliability and availability: ability to deal with memory failures (intermittent and permanent faults)