RISC-V Near-Memory Processing Accelerators

Peter Hsu, Ph.D.

Peter Hsu & Associates, S.L.
Barcelona, Spain

Email: peter.hsu@phaa.eu
LinkedIn: www.linkedin.com/in/peter-hsu-122a315

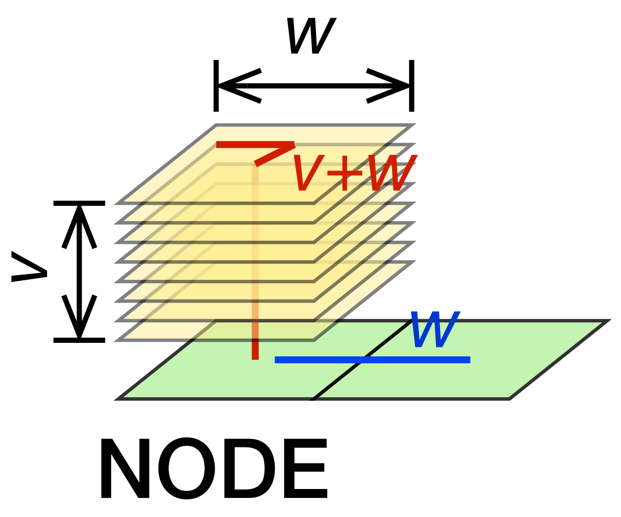
Introduction

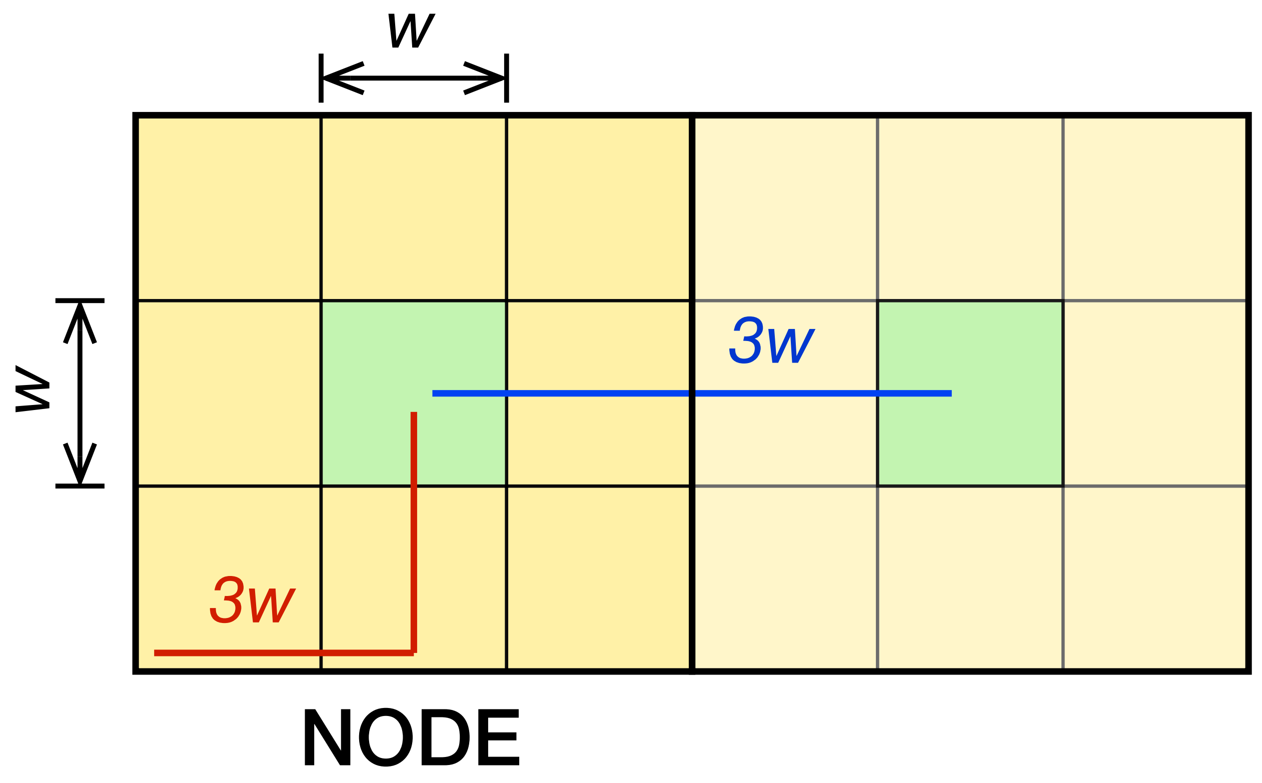
- RISC-V opens many new opportunities for innovation and enables smaller organizations to design computer hardware
- But no matter how creative, a new chip design using advanced process technology faces enormous development cost
- We propose a “mix and match” paradigm combining state-of-the-art memory technology with mature ASIC logic to reduce development cost of near-memory computing architecture accelerators
- Machine learning and supercomputing accelerators examples

Why Stacked Memory?

Technical

- Shorter SRAM access path, node communication distance

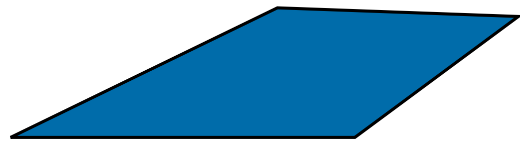




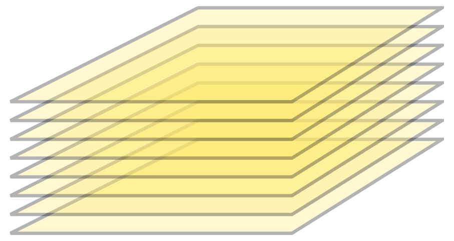
Business

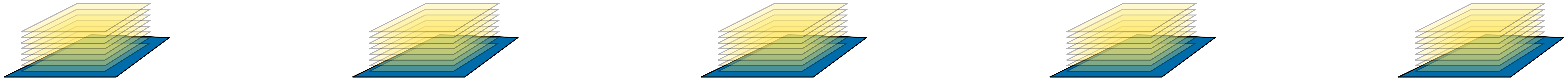
- Reduce development cost

Save cost by building custom logic on mature ASIC technology



Reuse stacked memory for different designs





Multiple accelerators specialize for different applications

Agenda

1. Economics & technology
2. Accelerator architecture
3. Examples
4. Programming model & evaluation
5. Memory technologies
6. Conclusion

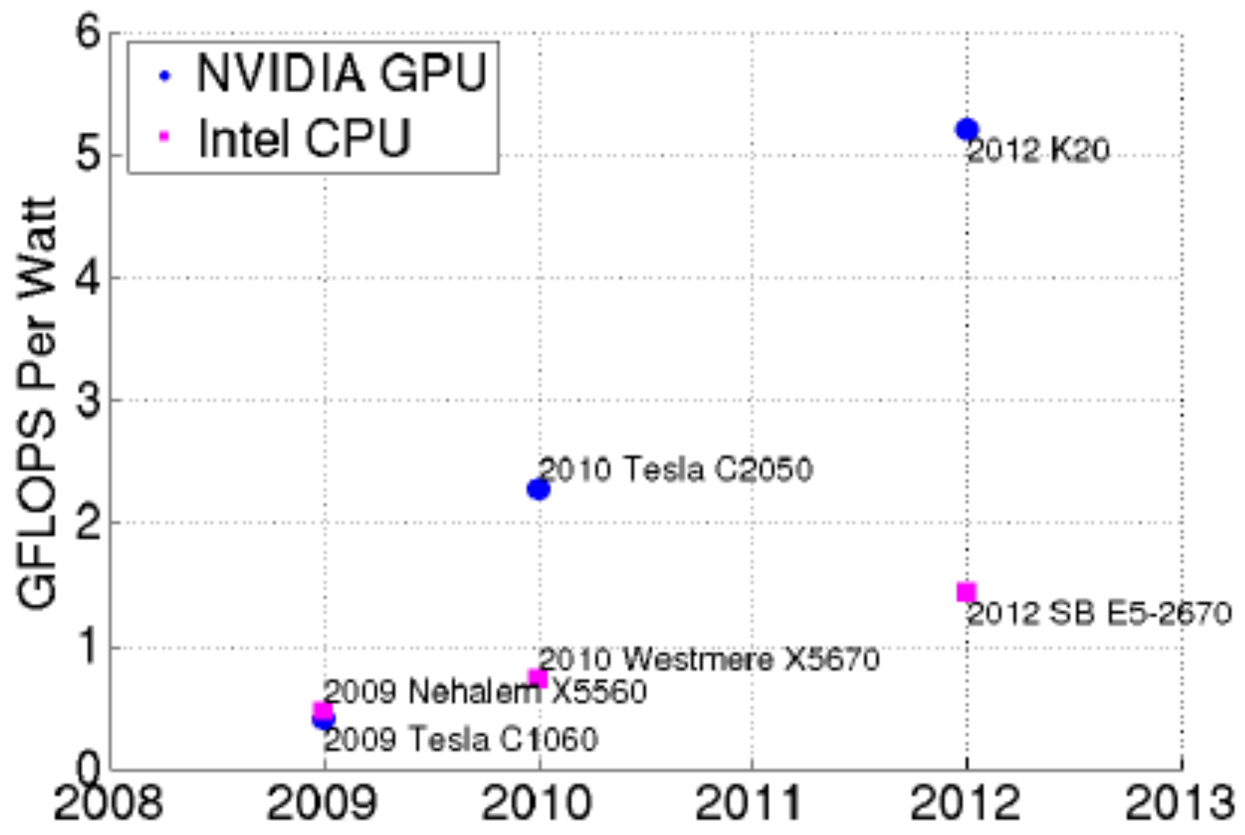
Many interesting accelerator architectures are possible

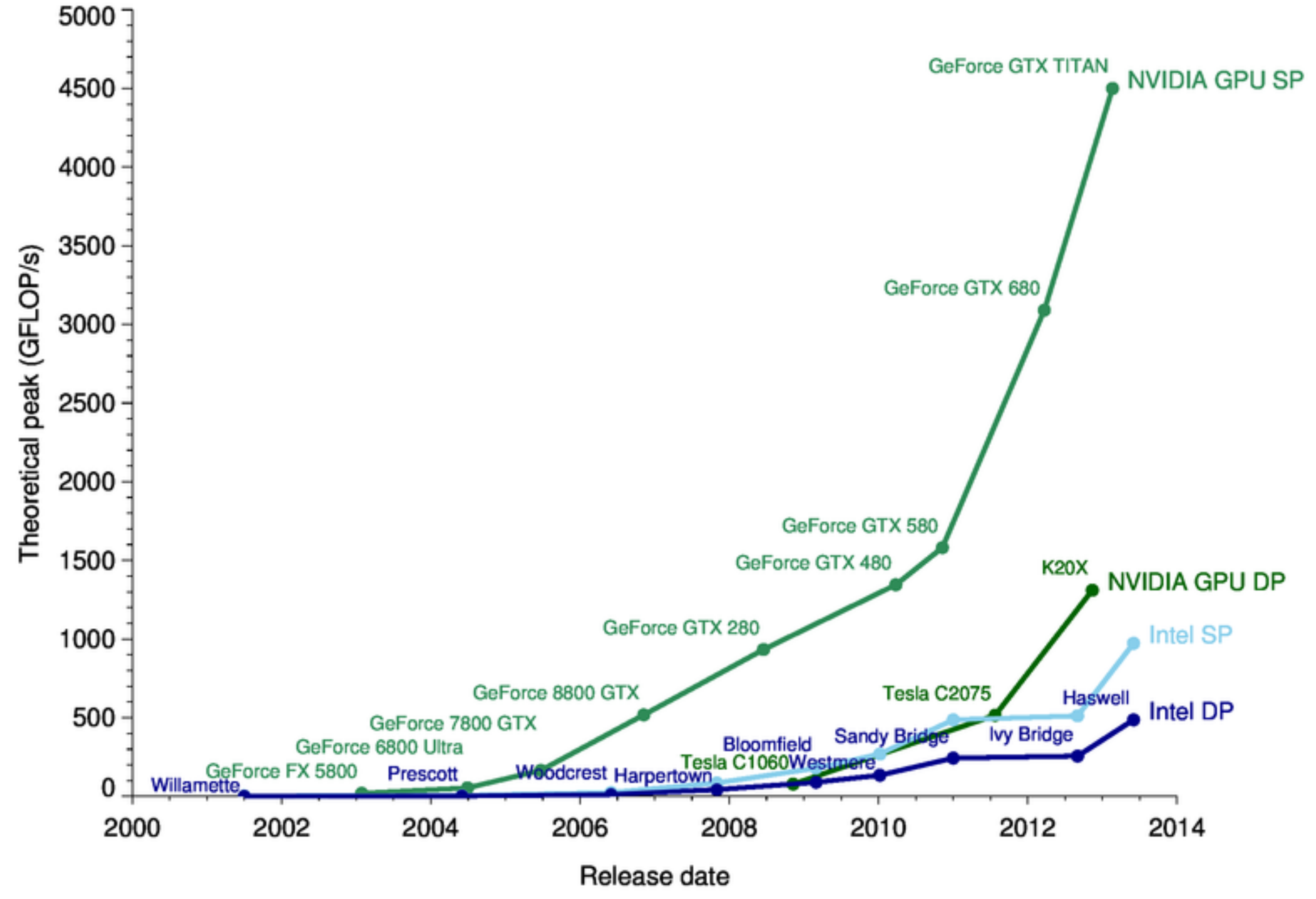
We propose a technical solution for near-memory computing accelerators that is economically feasible for smaller organizations with limited resources

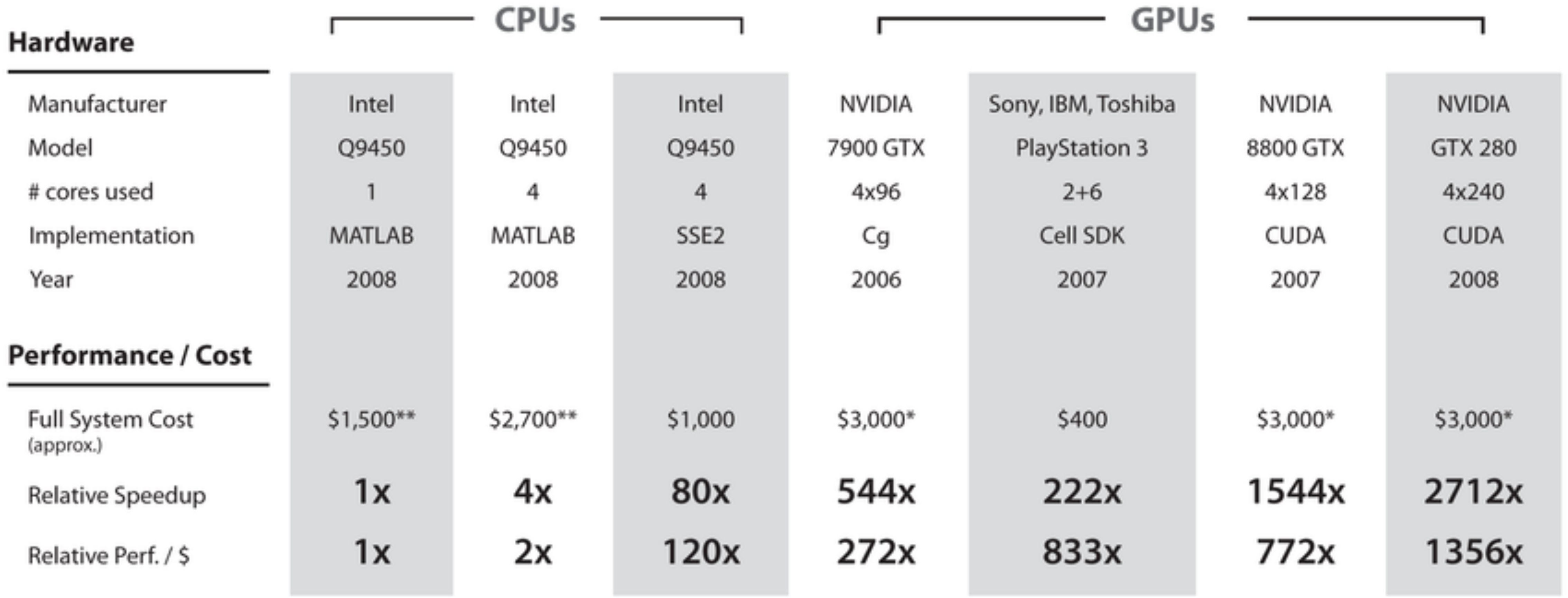
More details in paper “In-Memory Accelerators Using Stacked Memory” (PDF)

Why Specialize?

- Performance
- Energy







Hardware	CPUs			GPUs			
Manufacturer	Intel	Intel	Intel	NVIDIA	Sony, IBM, Toshiba	NVIDIA	NVIDIA
Model	Q9450	Q9450	Q9450	7900 GTX	PlayStation 3	8800 GTX	GTX 280
# cores used	1	4	4	4x96	2+6	4x128	4x240
Implementation	MATLAB	MATLAB	SSE2	Cg	Cell SDK	CUDA	CUDA
Year	2008	2008	2008	2006	2007	2007	2008
Performance / Cost							
Full System Cost (approx.)	$1,500**	$2,700**	$1,000	$3,000*	$400	$3,000*	$3,000*
Relative Speedup	1x	4x	80x	544x	222x	1544x	2712x
Relative Perf. / $	1x	2x	120x	272x	833x	772x	1356x

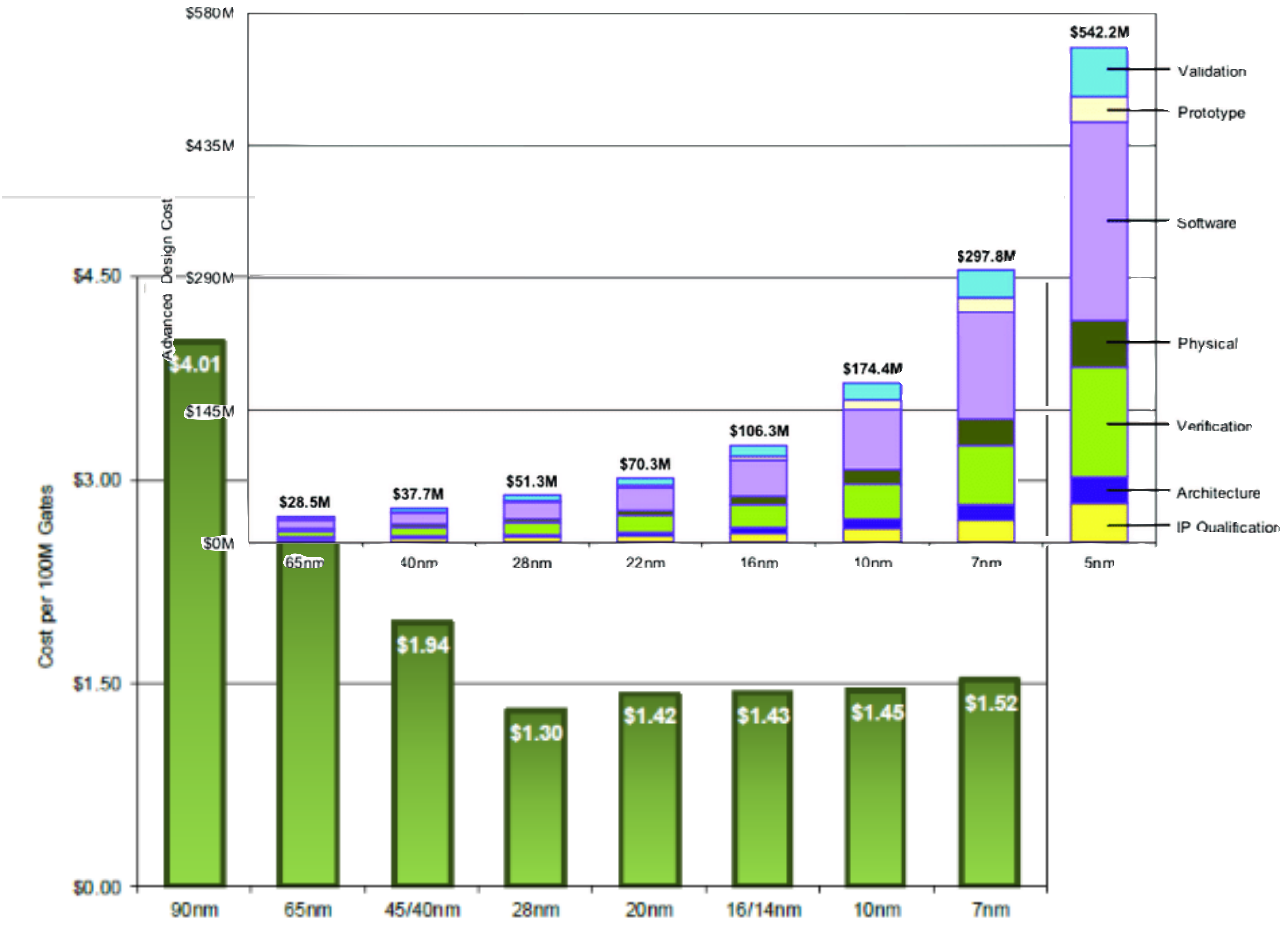
Economic Considerations

Cost of silicon device

- Recurring cost (RE)
- Non-recurring cost (NRE)

5nm chip $500M NRE

- 50M devices → $10/device
- 100K → $5000/device



Source: International Business Strategies, Inc.

Problem

- RISC-V enables hardware innovation by smaller organizations
    - No architecture licensing fee, no limits on customizing ISA
    - Open source software ecosystem with compilers, OS…
- But creating commercially competitive accelerator is challenging
    - Leading edge semiconductor design is extremely expensive
    - Novel ideas take time to gain momentum and volume in market

Solution

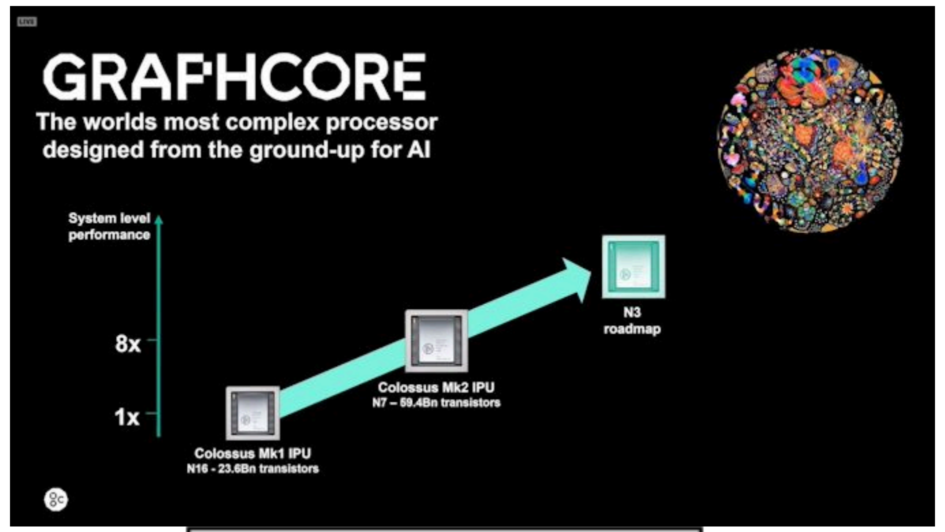
“End of Moore’s Law” → rapid advances in packaging technology

- Multiple chips on substrate (Open Chiplet Initiative)
- Chip/wafer stacking (TSMC 3DFabric™)
- Example: GPU die with stacked HBM memories on substrate

Mix and match advanced memory and mature logic technology

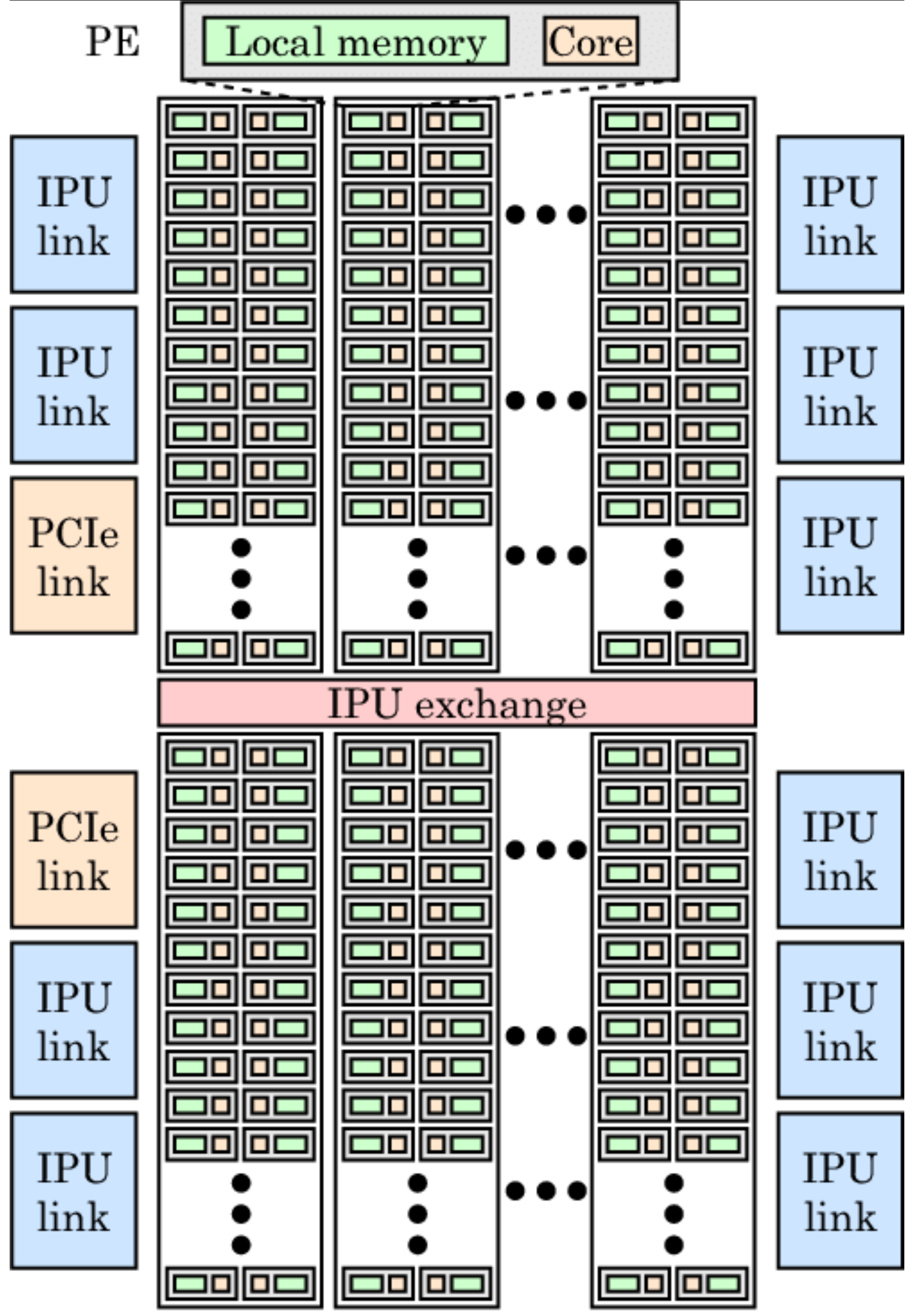
- Multiple accelerators share design cost of stacked memory

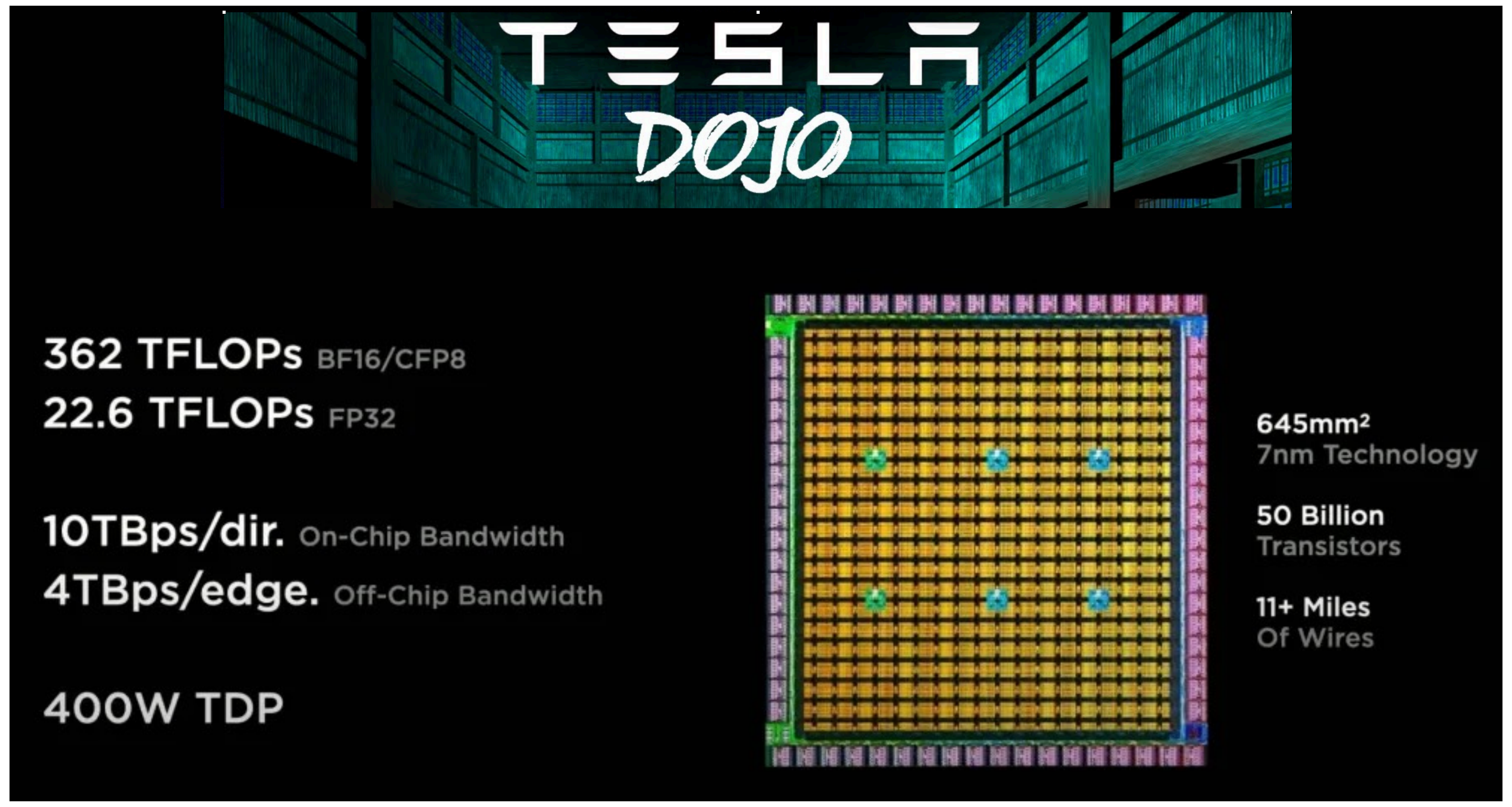
Some Recent Accelerators



GRAPHCORE

The worlds most complex processor designed from the ground-up for AI





TESLA DOJO

362 TFLOPs BF16/CFP8

22.6 TFLOPs FP32

10TBps/dir. On-Chip Bandwidth

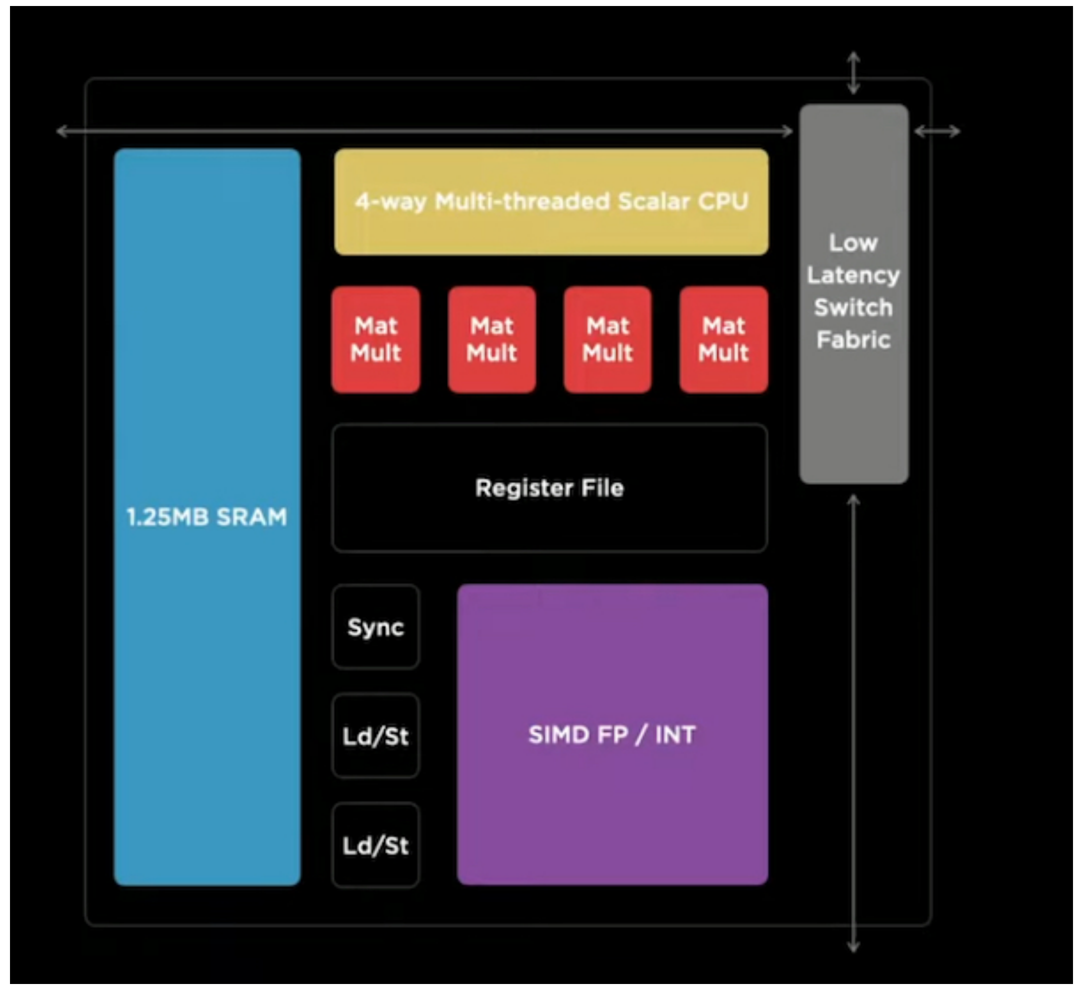
4TBps/edge. Off-Chip Bandwidth

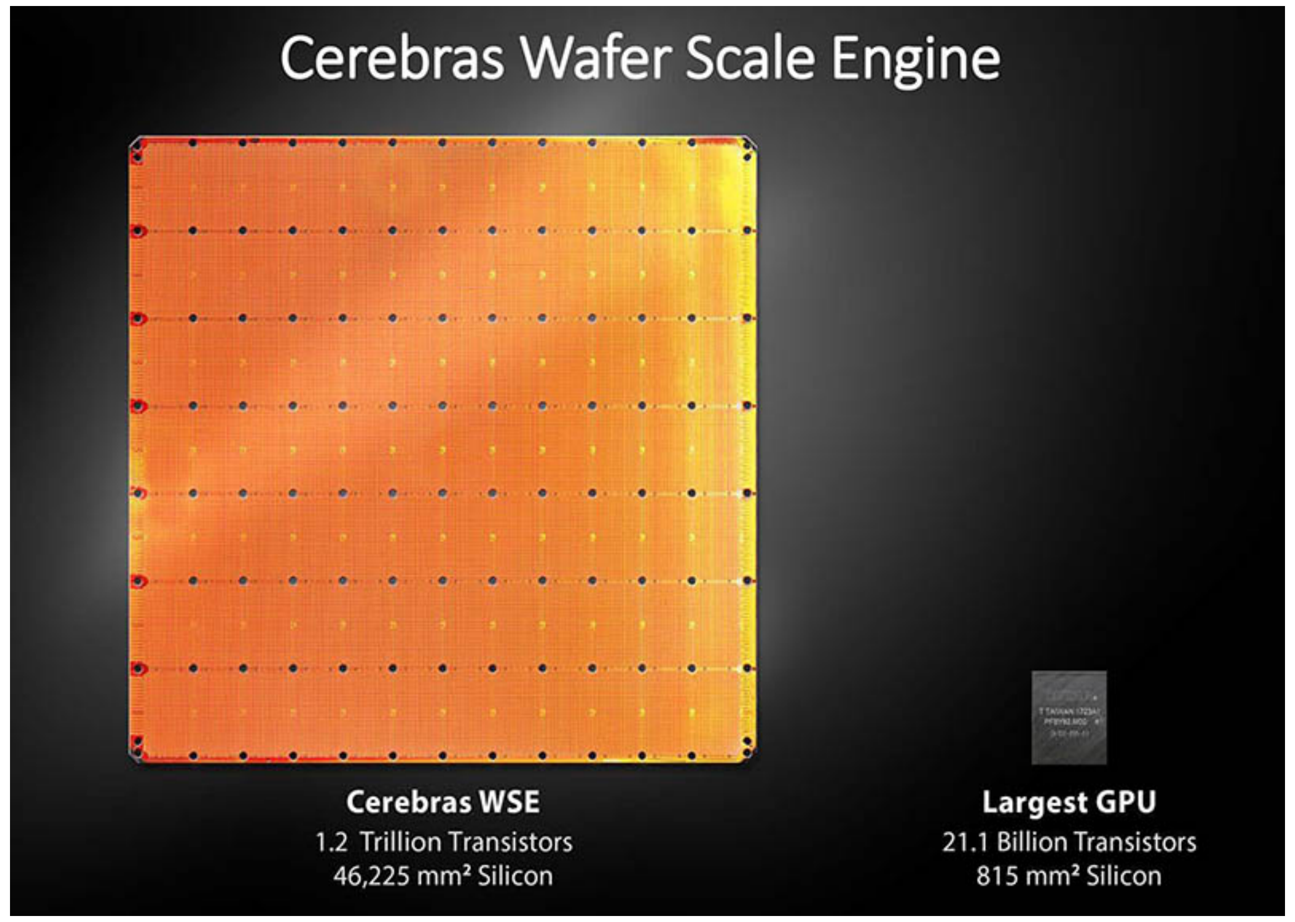
400W TDP

645mm²
7nm Technology

50 Billion
Transistors

11+ Miles
Of Wires

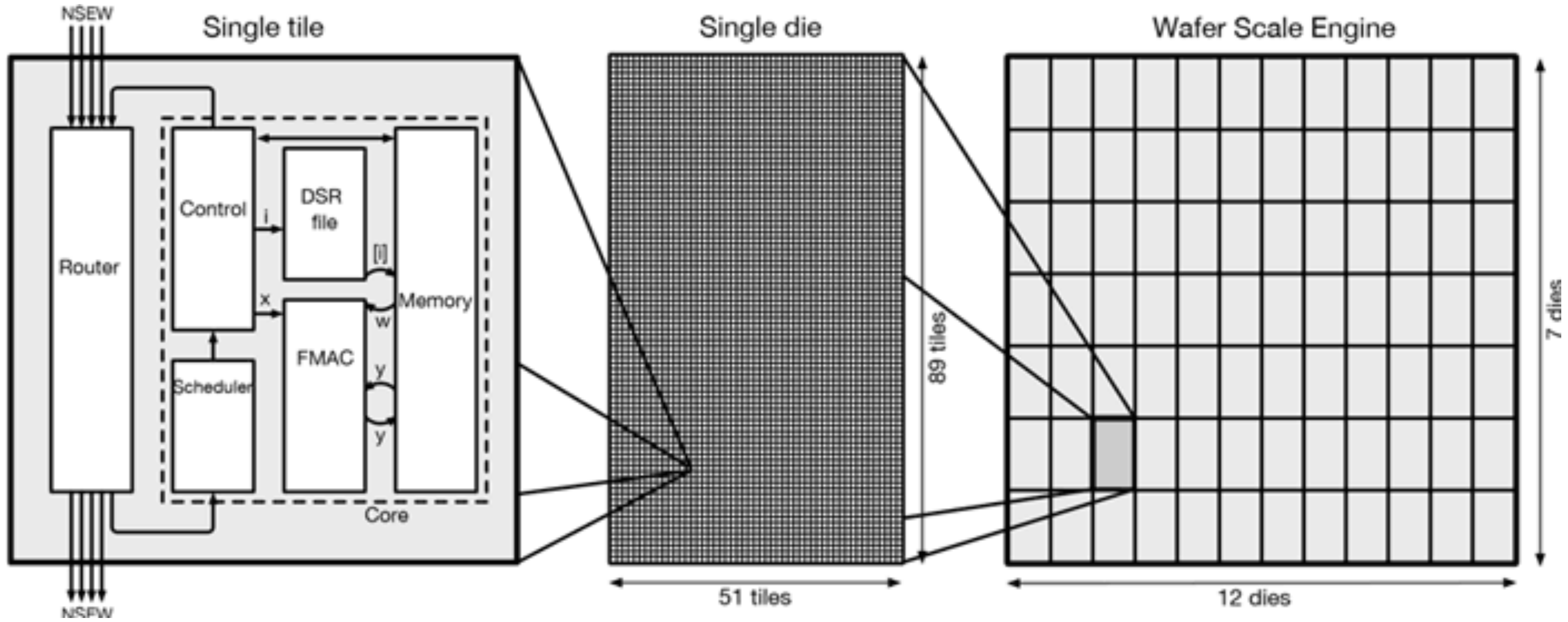




Cerebras Wafer Scale Engine

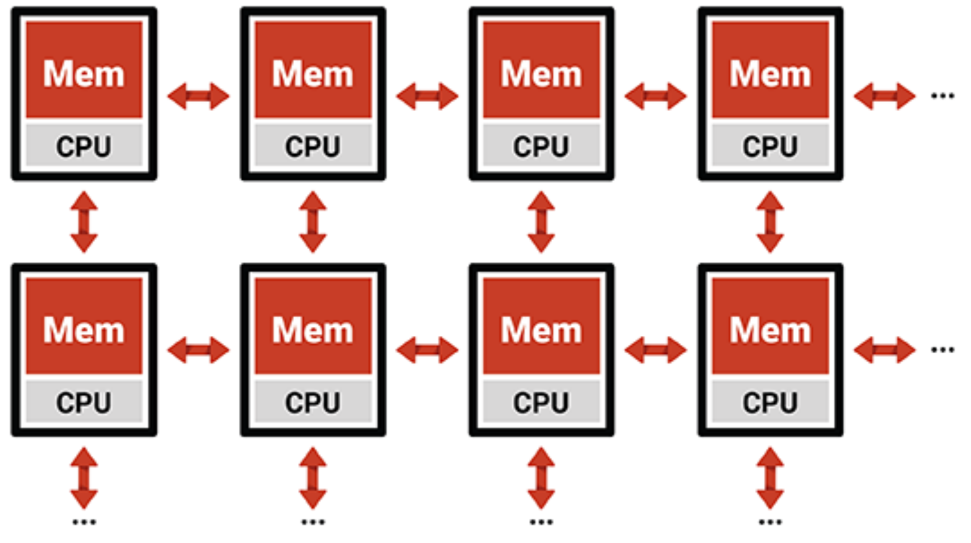
Cerebras WSE
1.2 Trillion Transistors
46,225 mm² Silicon

Largest GPU
21.1 Billion Transistors
815 mm² Silicon



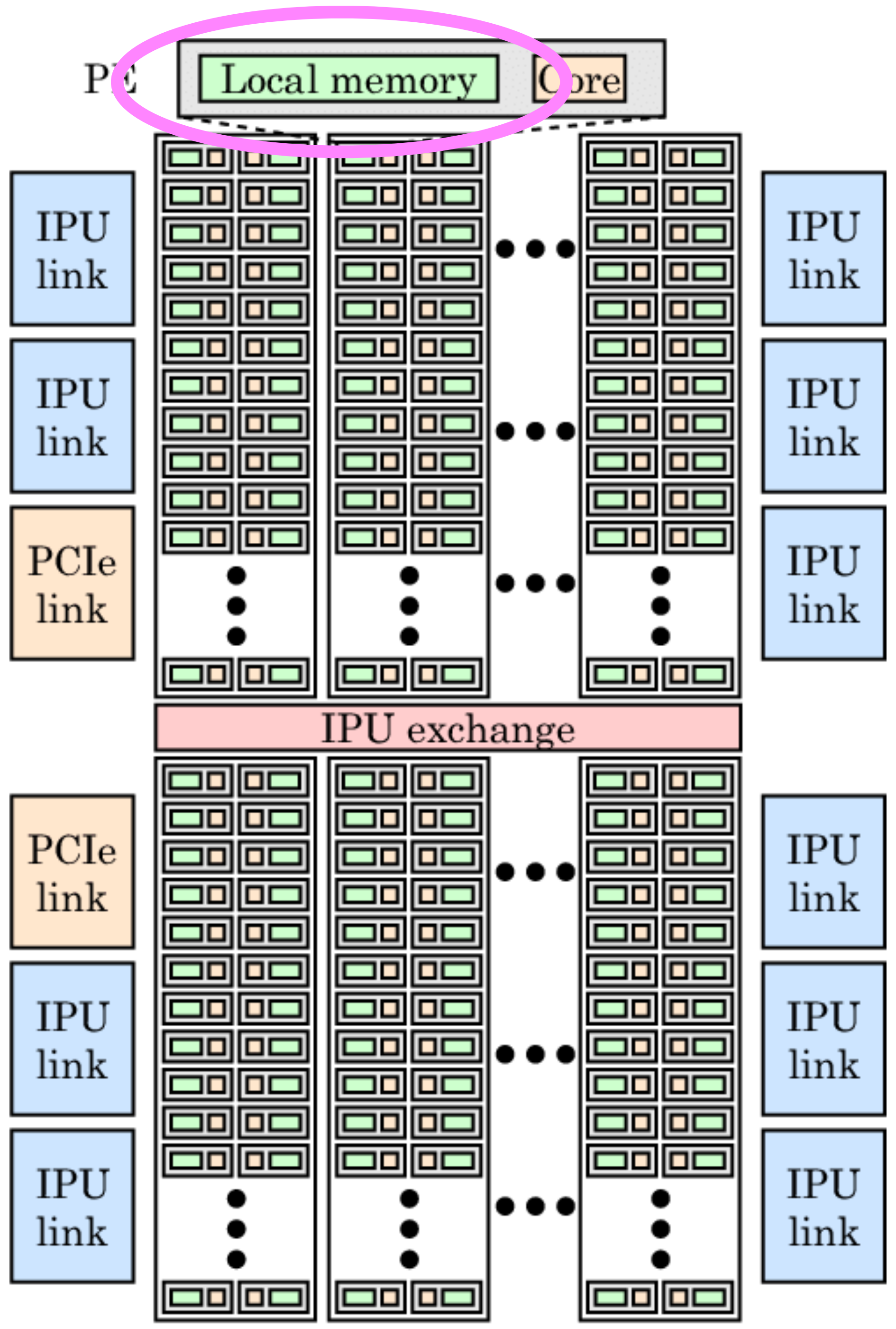
In-, At-, Near-Memory Computing

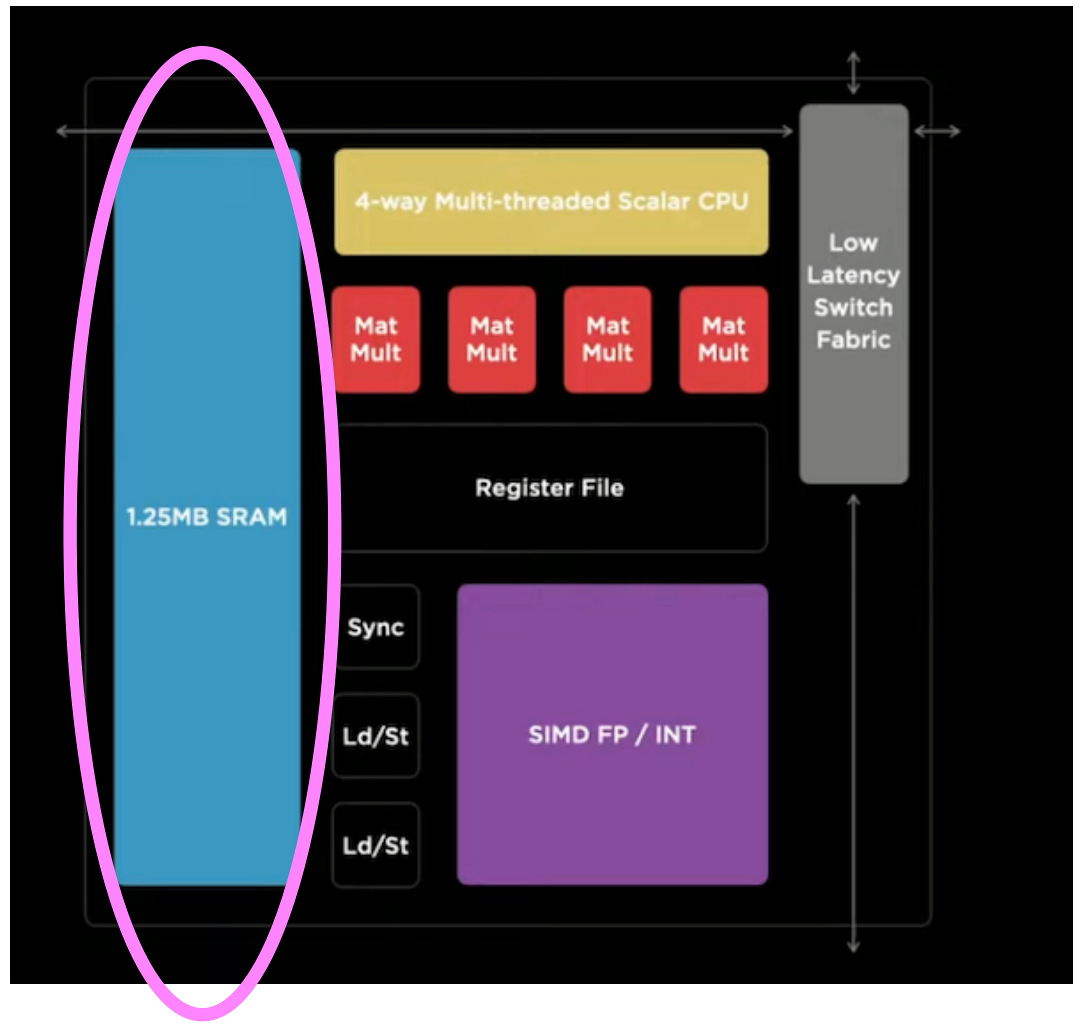
Global In-Memory Computing Market Is Expected to Reach USD 41.53 Billion by 2028 : Fior Markets

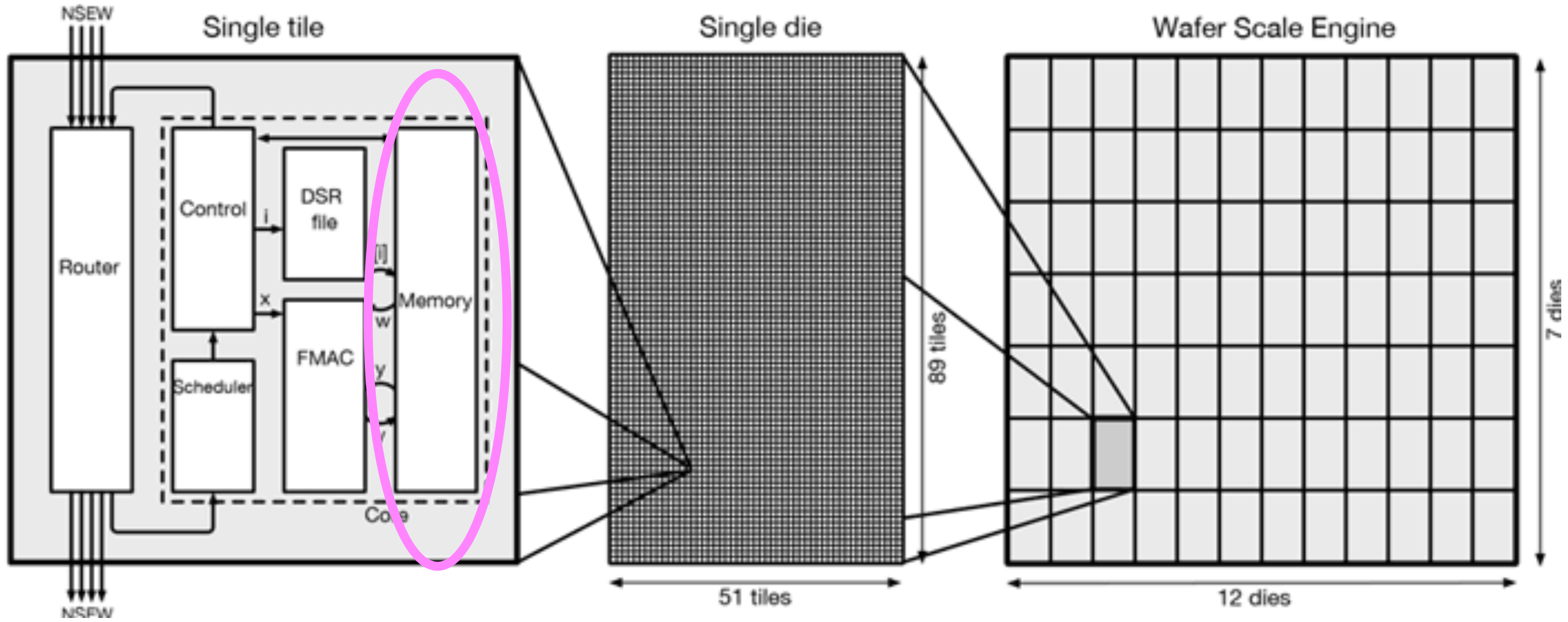


Compute-in-Memory IC

Large Amounts of Local Memory







Memory Chip Different from ASIC

FLASH, DDR, LPDDR, even HBM are high volume chips because

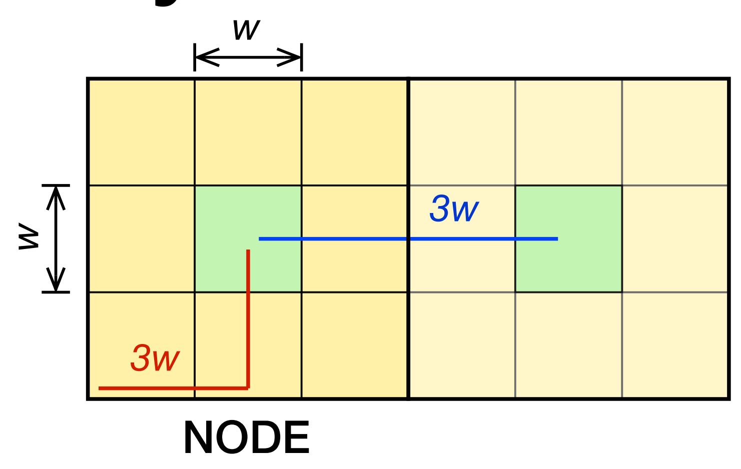
- Same chip used in multiple products
- Many chips in a single product

Memory cell density/flexibility tradeoff

- Dedicated memory fab → lowest cost, inflexible, standard parts
- Embedded logic fab → medium cost, customizable like ASIC

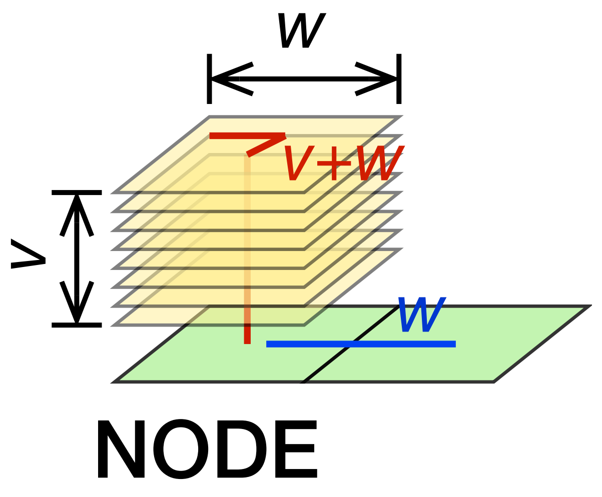
Mix and Match Paradigm

Monolithic in-memory computing chips use a lot of area for memory



- Expensive advanced logic process not fully utilized
- Long wires over memory area between logic islands waste energy

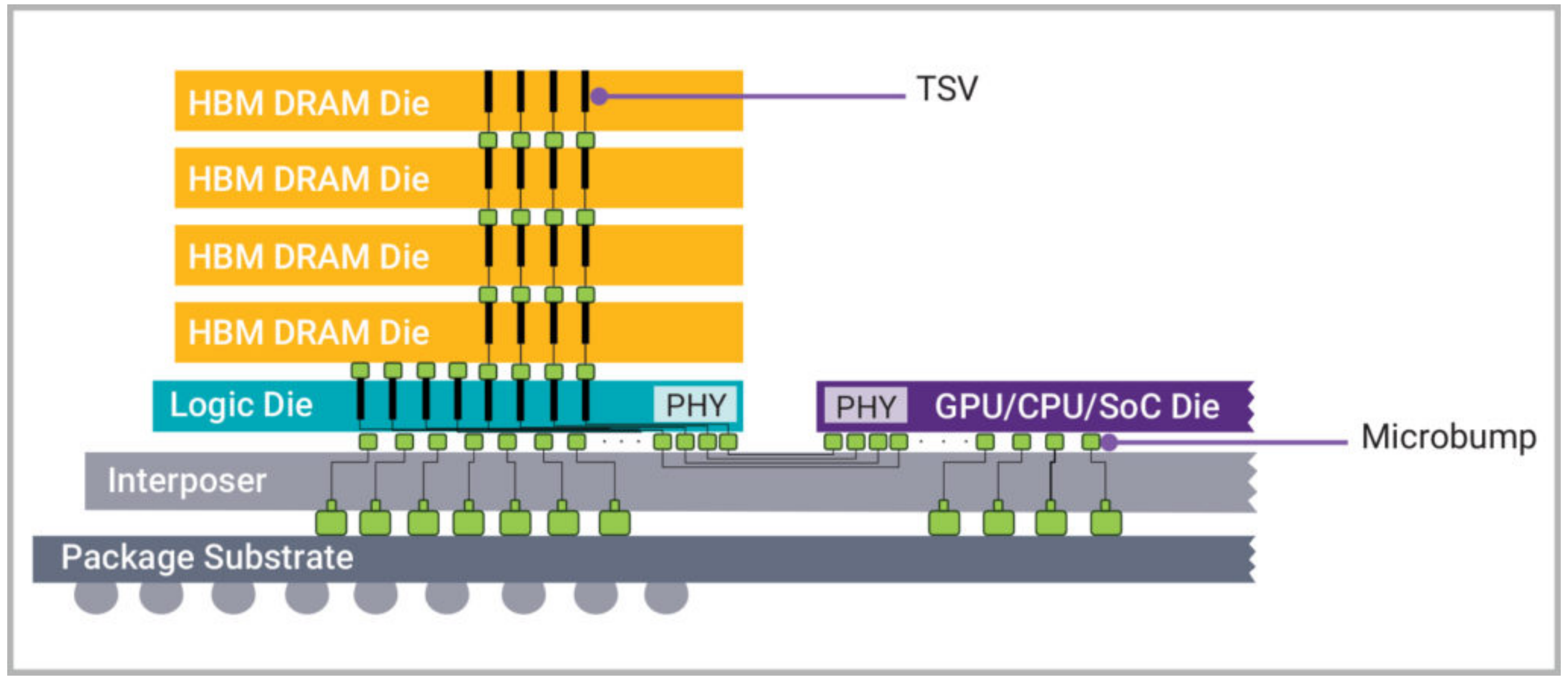
We propose separating memory and logic onto stacked dies



- Differently optimized process for memory and logic gates
- Contiguous logic reduces wire length, improves energy efficiency

Wafer Stacking

- Same size dies, yield = one big die
- Low cost, good for memory chips
- Yield problem for logic die



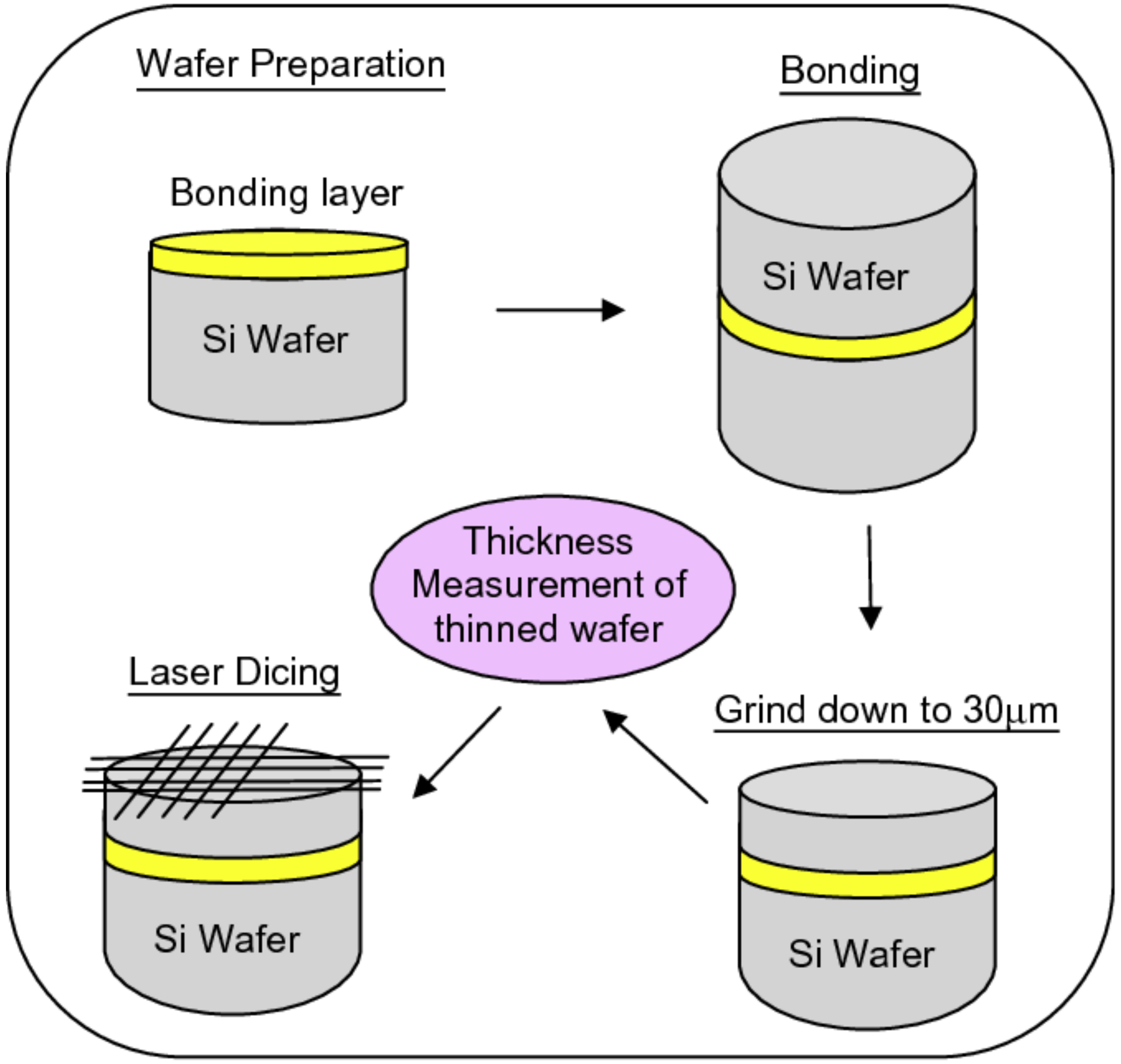
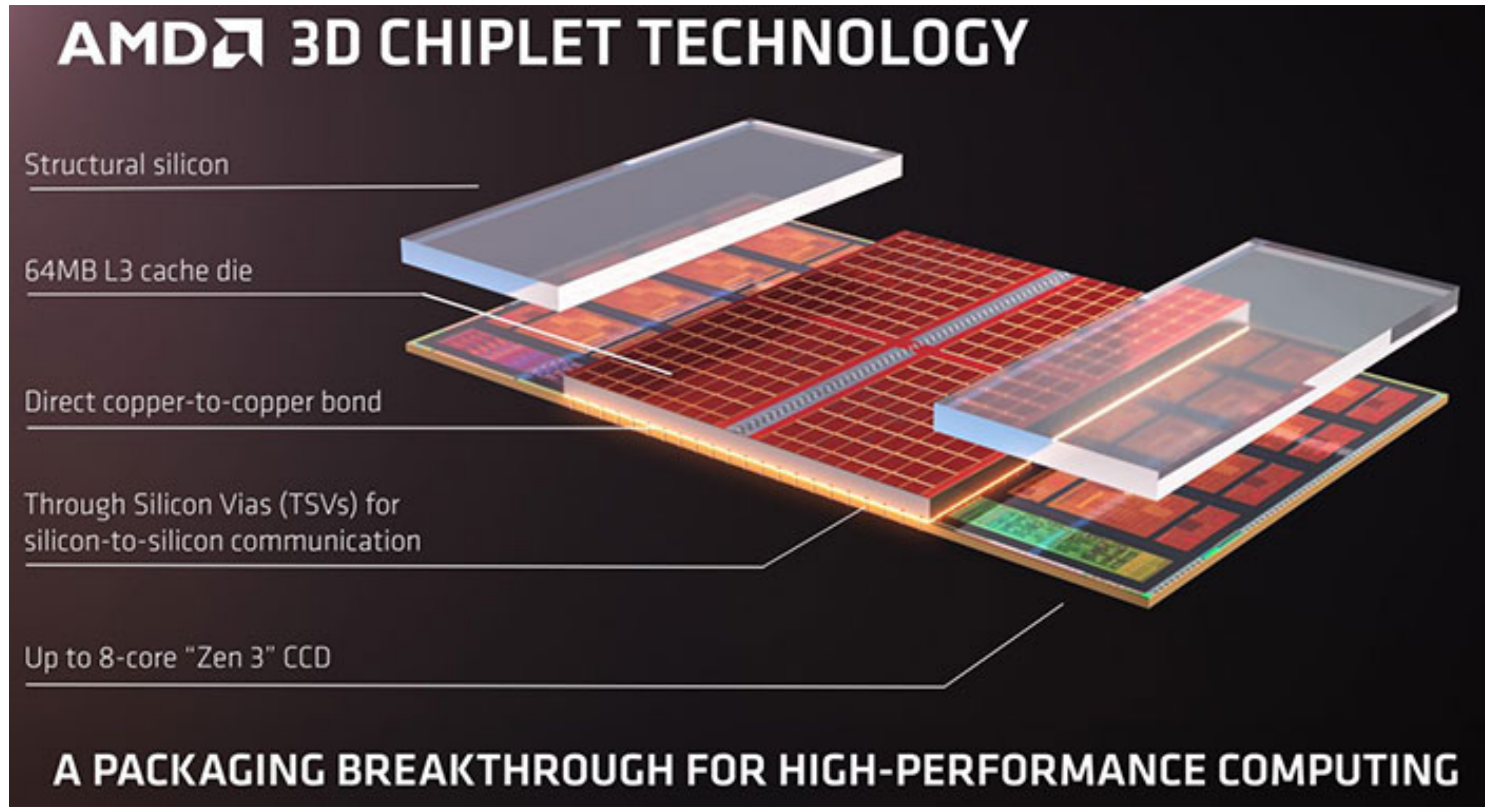
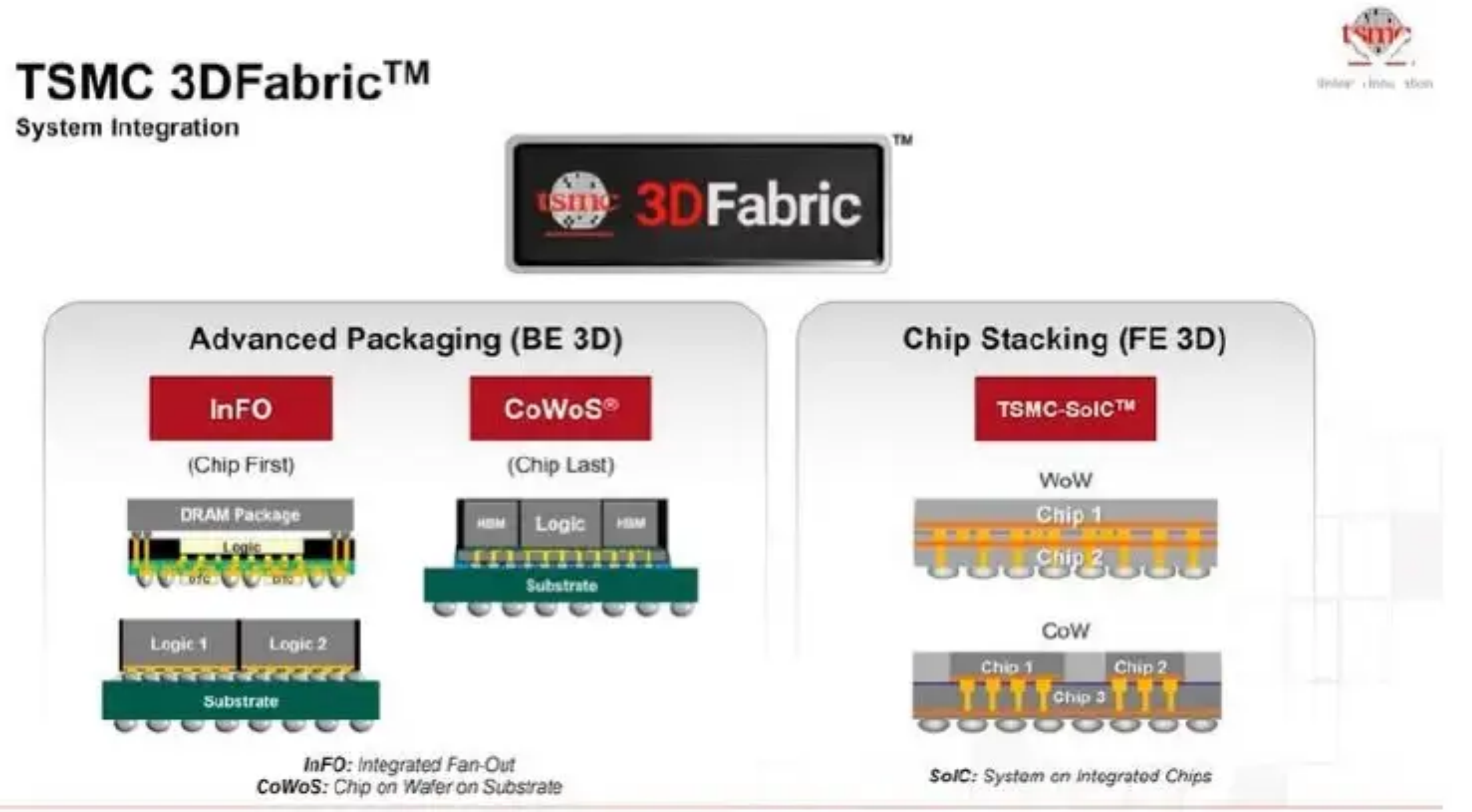


Fig. 1 Process sequence for this experiment

Chip Stacking

- Accommodate different size dies, different processes
- Test before stacking (KGD) → large logic chip is possible
- Short vertical interconnect using TSV → good performance, power
- Higher cost

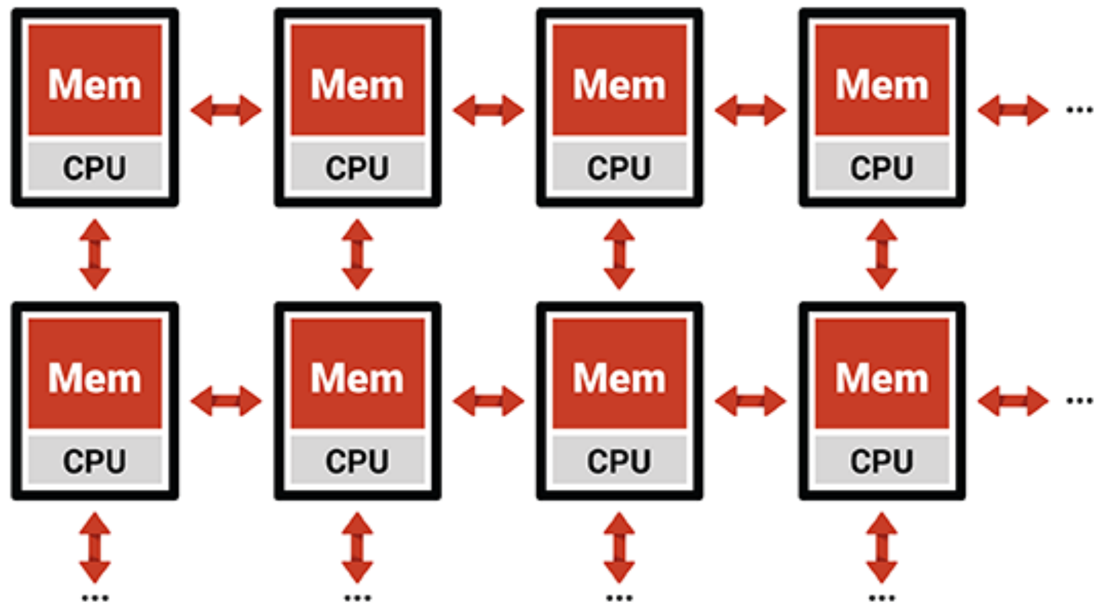

AMD 3D CHIPLET TECHNOLOGY
Structural silicon
64MB L3 cache die
Direct copper-to-copper bond
Through Silicon Vias (TSVs) for silicon-to-silicon communication
Up to 8-core "Zen 3" CCD
A PACKAGING BREAKTHROUGH FOR HIGH-PERFORMANCE COMPUTING



Memory Stack

TSMC 7nm SRAM optimized process ≈ 2 MB/mm²

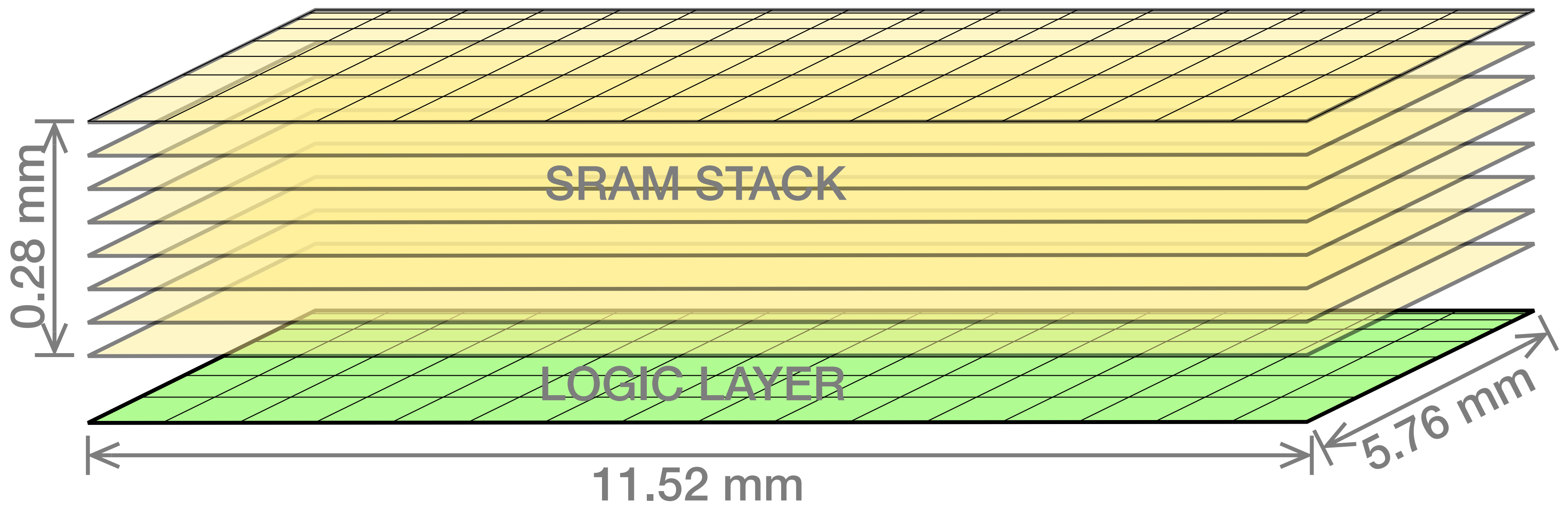
- Chose 66 mm² stack area for reasonable yield
- 1 GB in 8 layers = 531 mm² silicon area
- 4 TB/s bandwidth using 10% TSV area overhead



Compute-in-Memory IC

Multiple stacks on large logic chip

- More capacity, bandwidth



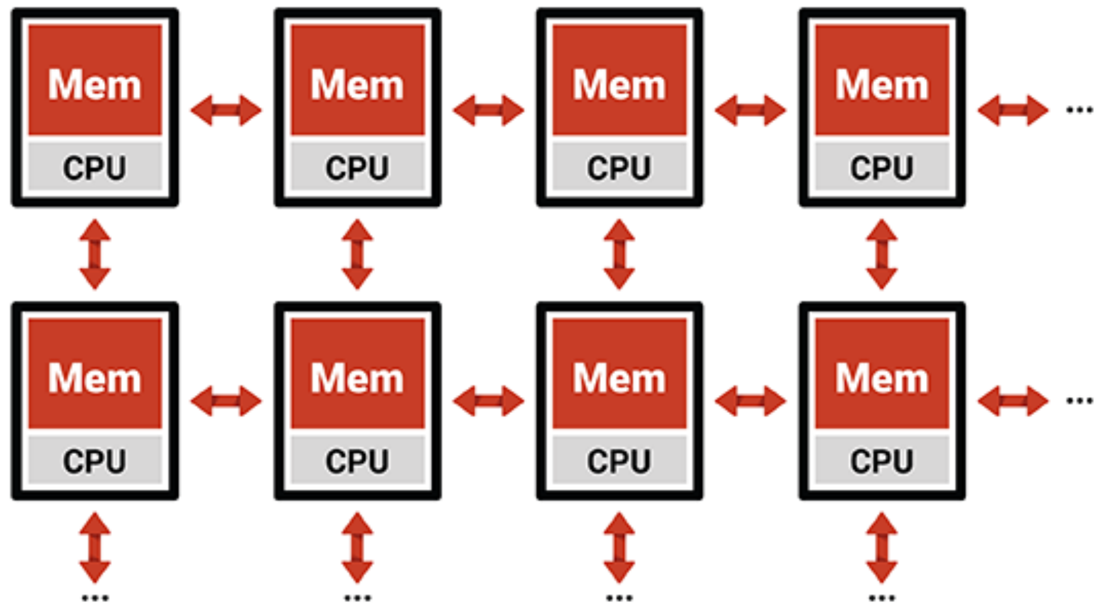
Logic Chip

Mature technology

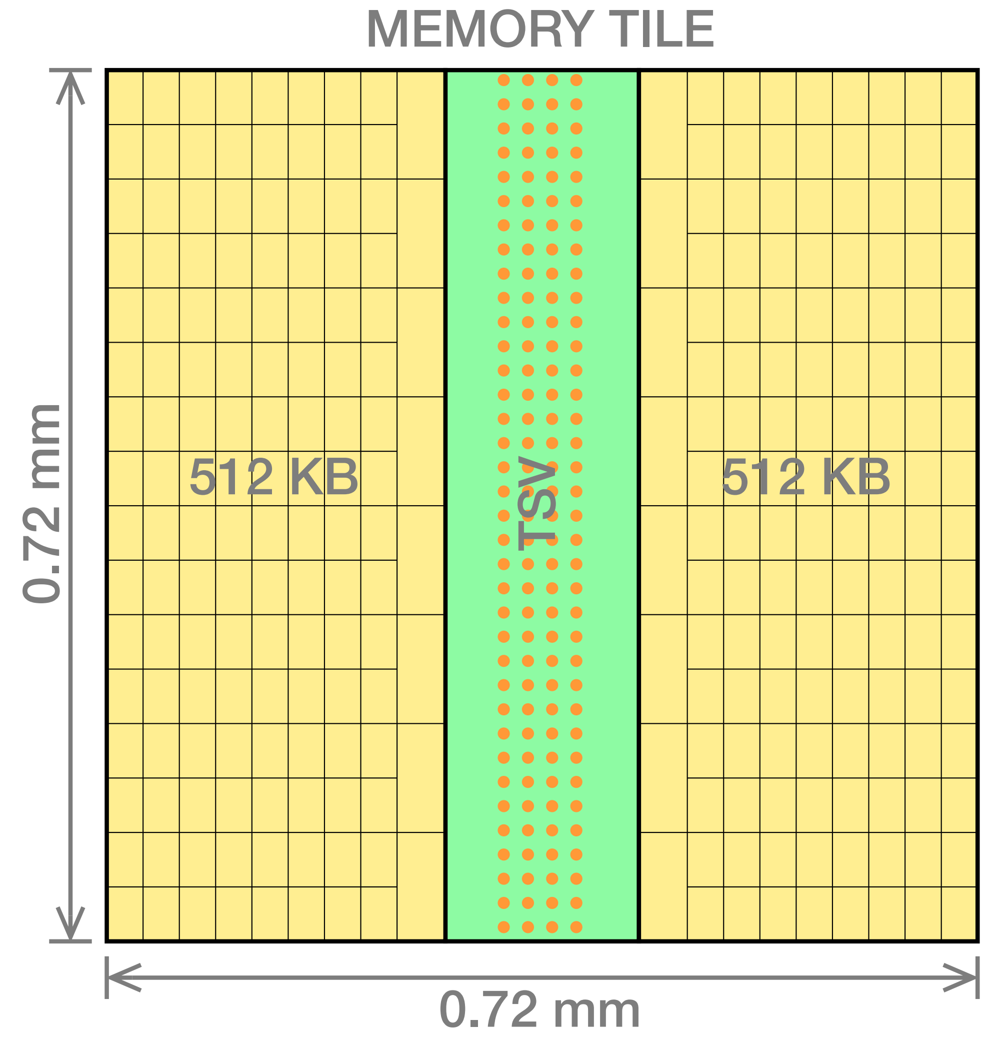
- GlobalFoundries 22FDX, TSMC 22ULP…
- Low CAD tools cost, many IP available

66 mm² stack → 128 logic tiles each 0.52 mm²

- 1.8 MGE (million gate equivalent) per tile
  ≈ 12 IEEE 64-bit FP multiply-add units
- 230 MGE per stack (1500 DP FPMAC)



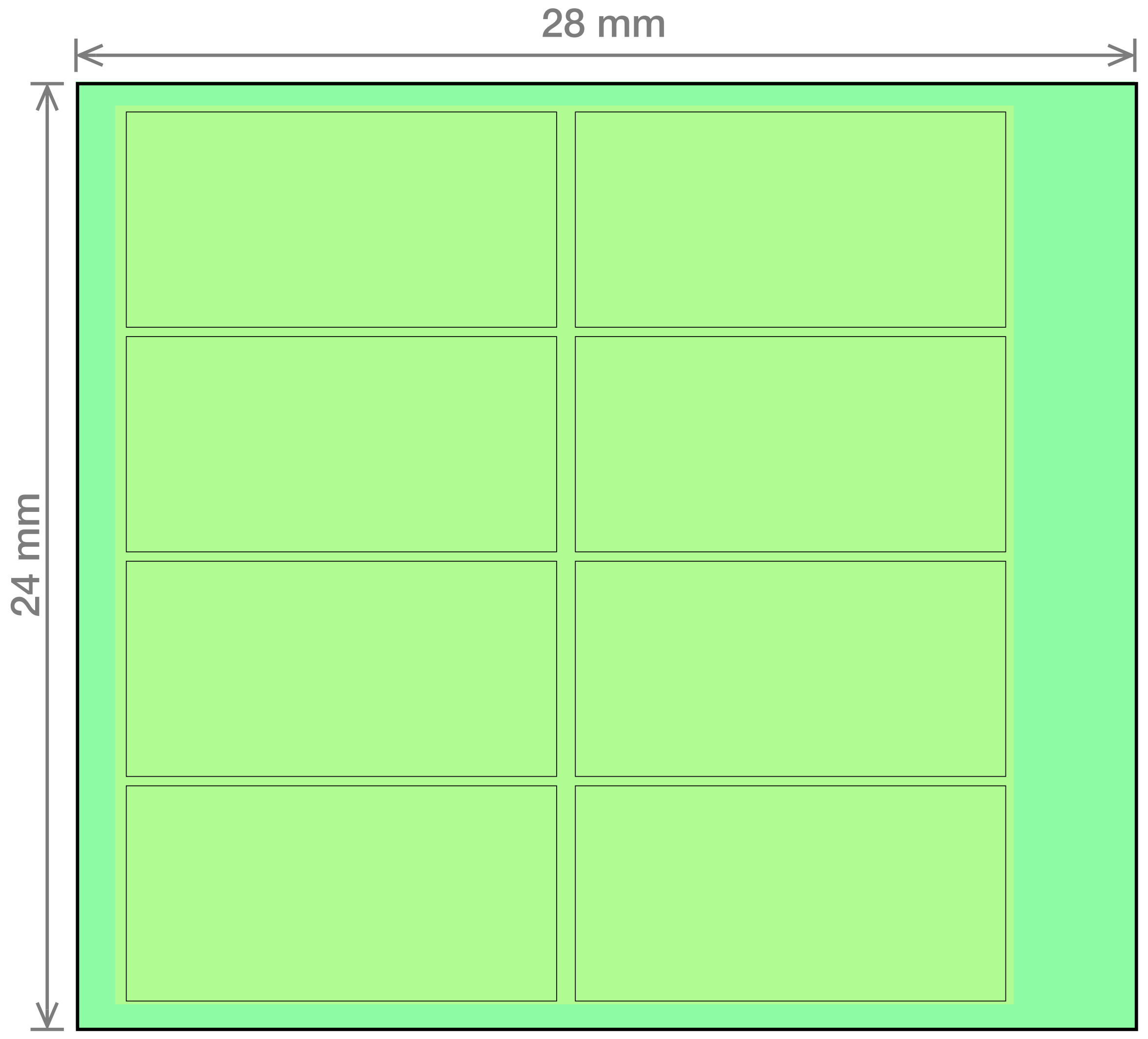
Compute-in-Memory IC

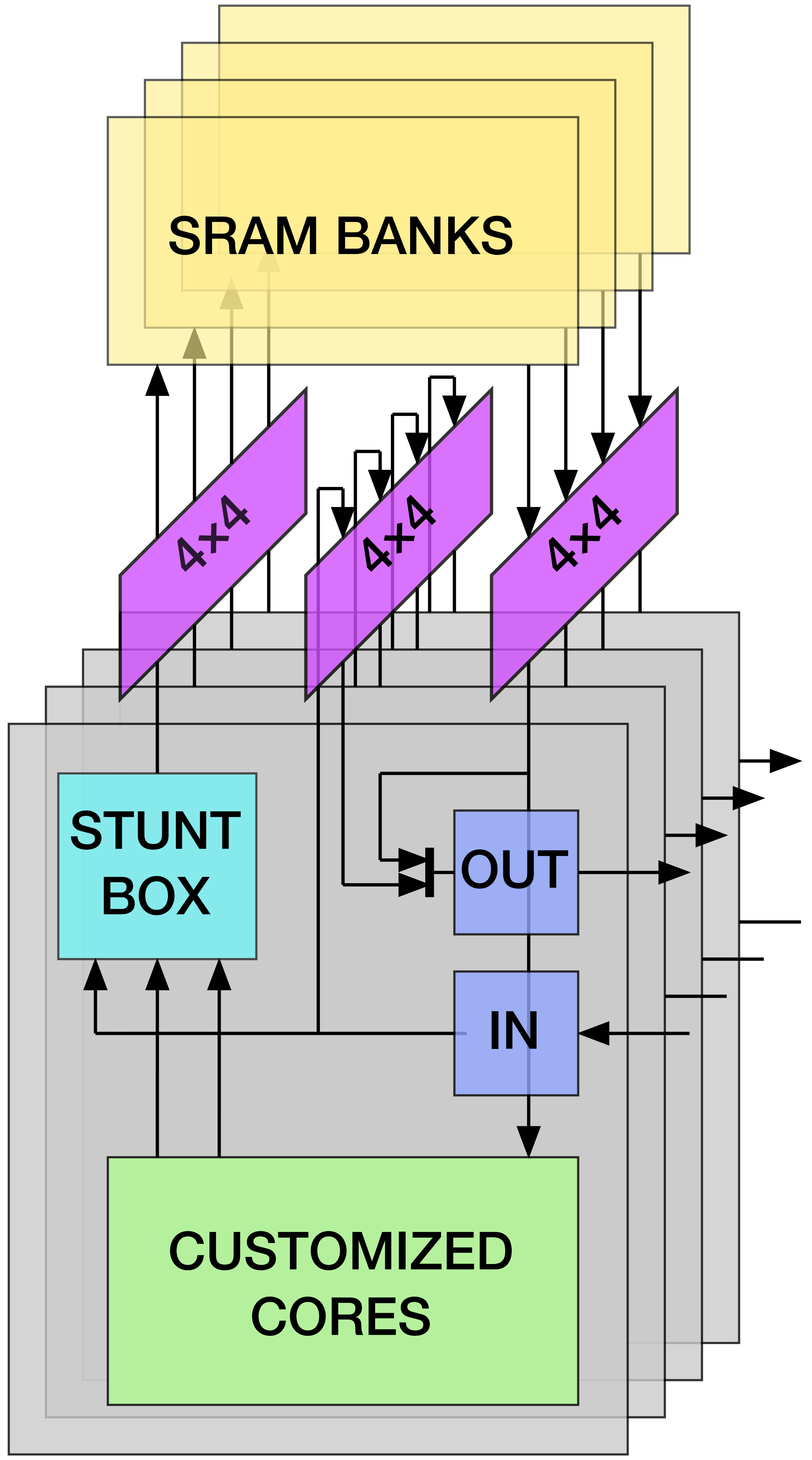


Platform Architecture

8 SRAM stacks on 672 mm² logic chip (24×28 mm)

- 1024 tiles
- 1 GHz logic in 22 nm
- 2D torus NoC, 32-bit links





Tile is a complete computer

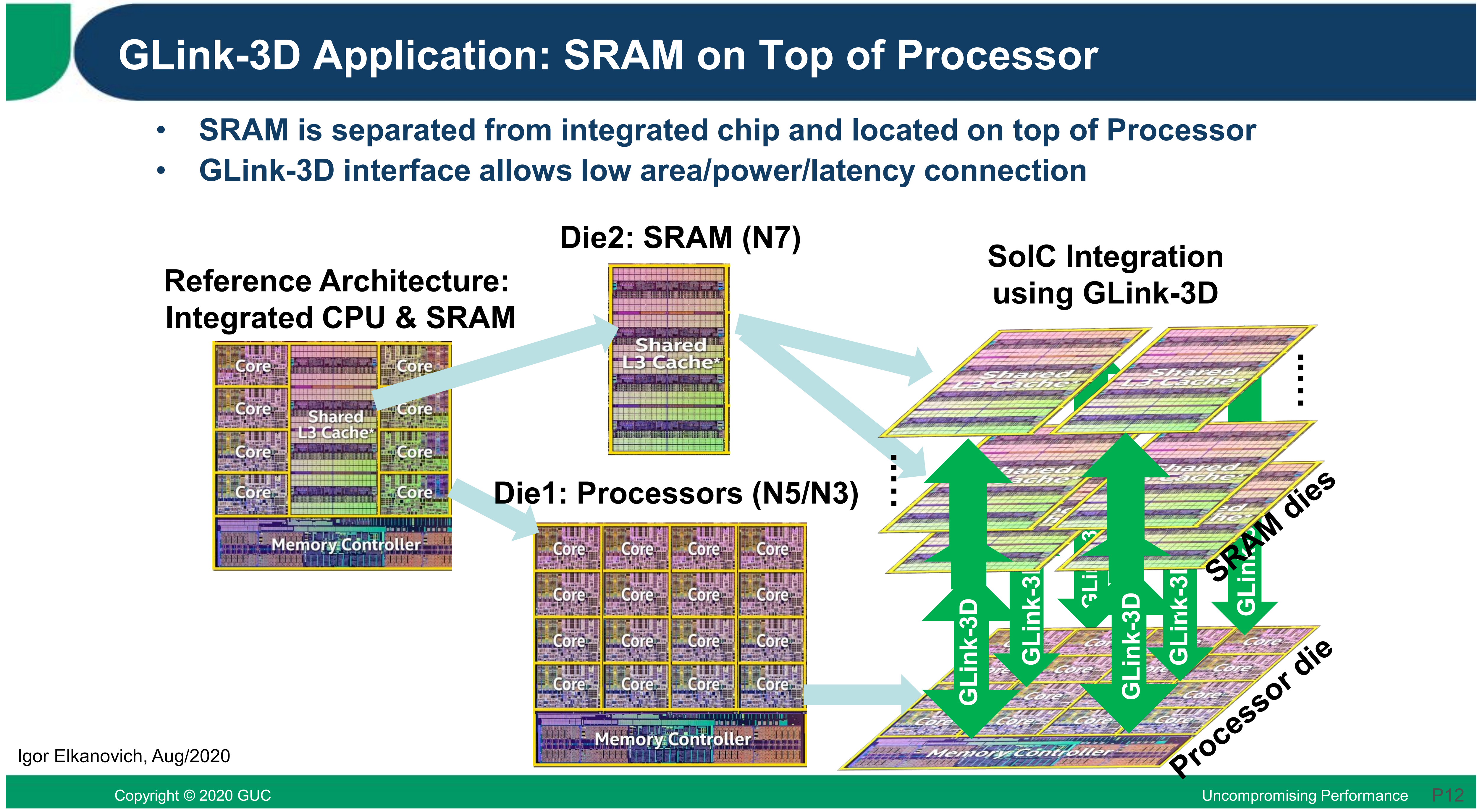
- Shared memory multicore architecture (SMP)
- Eight 32-bit memory accesses per cycle (32 GB/sec)

Manufacturing Ecosystem

- TSMC has been developing stacked memory technology for some time

- CAD tools, TSV PHY already available

- Mixing mature logic chip is a business decision



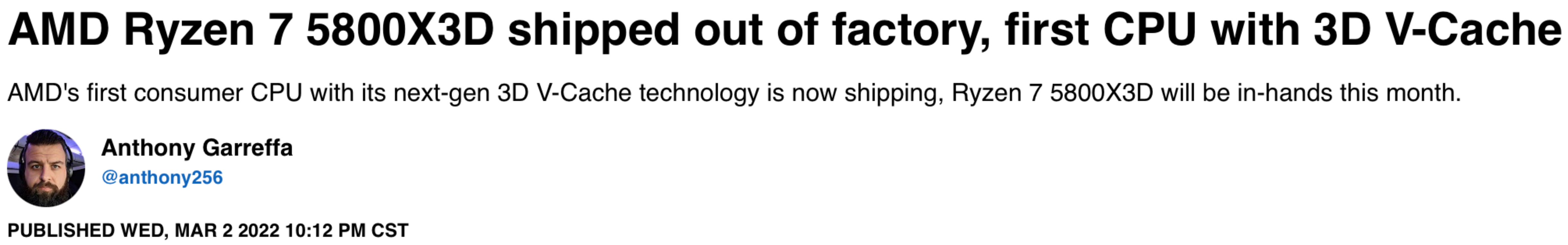
GLink-3D Application: SRAM on Top of Processor

- SRAM is separated from integrated chip and located on top of Processor
- GLink-3D interface allows low area/power/latency connection

Igor Elkanovich, Aug/2020

Copyright © 2020 GUC

Uncompromising Performance P12



AMD Ryzen 7 5800X3D shipped out of factory, first CPU with 3D V-Cache

AMD's first consumer CPU with its next-gen 3D V-Cache technology is now shipping, Ryzen 7 5800X3D will be in-hands this month.

Anthony Garreffa
@anthony256

PUBLISHED WED, MAR 2 2022 10:12 PM CST

Agenda

1. Economics & technology
2. Accelerator architecture
3. Examples
4. Programming model & evaluation
5. Memory technologies
6. Conclusion

We illustrate with examples chosen for simplicity of explanation

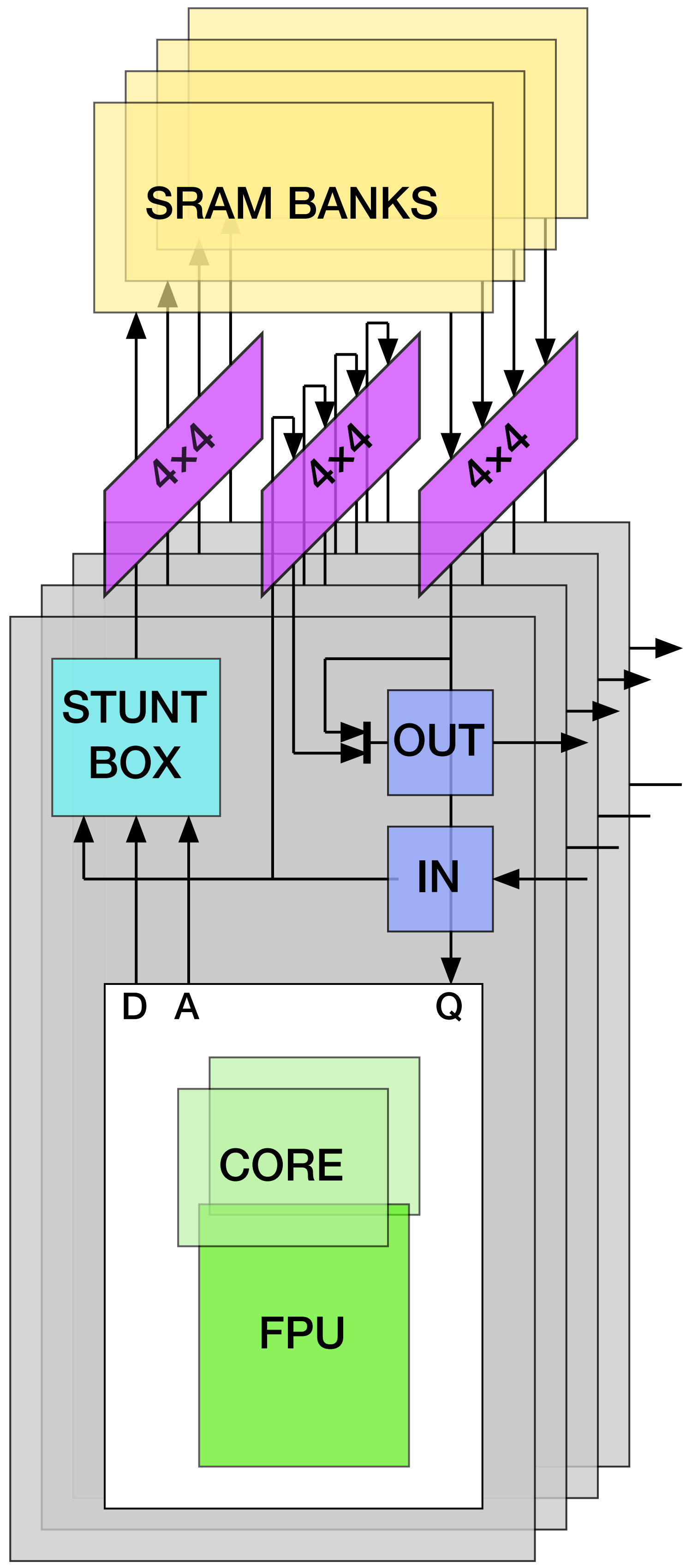
Real commercial designs could do better

HPC Example (SpMV++)

Specialized sparse matrix accelerator

- 8 GB, 4 TFLOPS (DP), 250 W PCIe card
- 8 bytes/FLOP memory bandwidth
- Logic 147.5 W (144 pJ per tile, 22nm)
- SRAM 25.6 W (25.6 pJ per tile, 7nm)

Evaluate using HPCG benchmark



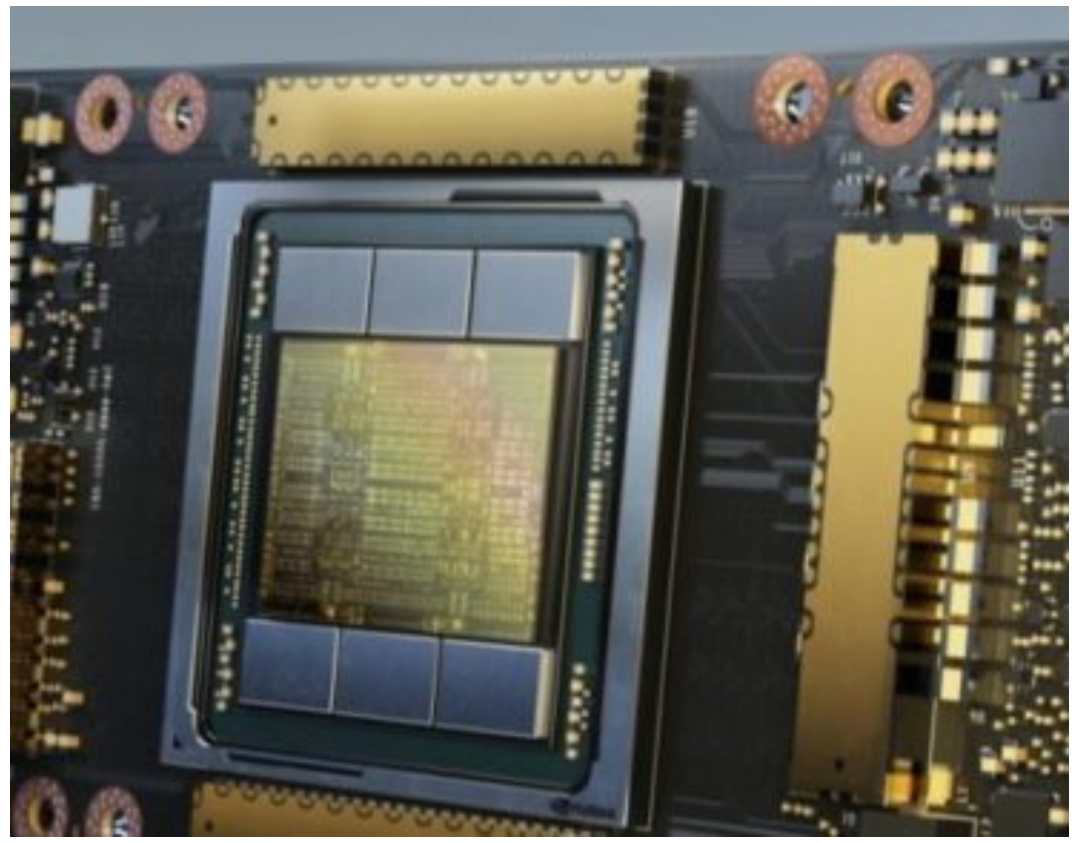
Tile

SpMV++ vs. GPU

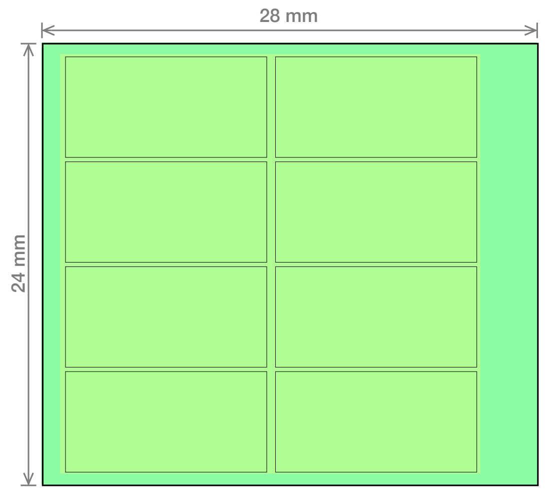
NVIDIA A100 is today's premier HPC accelerator

- 7 nm
- 826 mm^2
- HBM2E
- 250 W

Technology	A100	SpMV++	Improve
Capacity (GB)	80	8	0.1x
Peak TFLOPS (DP)	19.5	4.1	0.21x
Bandwidth (TB/s)	2	32.8	16x
Bytes/FLOP	0.10	8.0	78x
HPCG TFLOPS	0.227	2.9	13x
FPU Utilization*	1.16%	70%	60x
Power (W)	250	250	1x
GFLOPS/W	0.91	11.5	13x
	*estimate for SpMV++		



NVIDIA A100 GPU

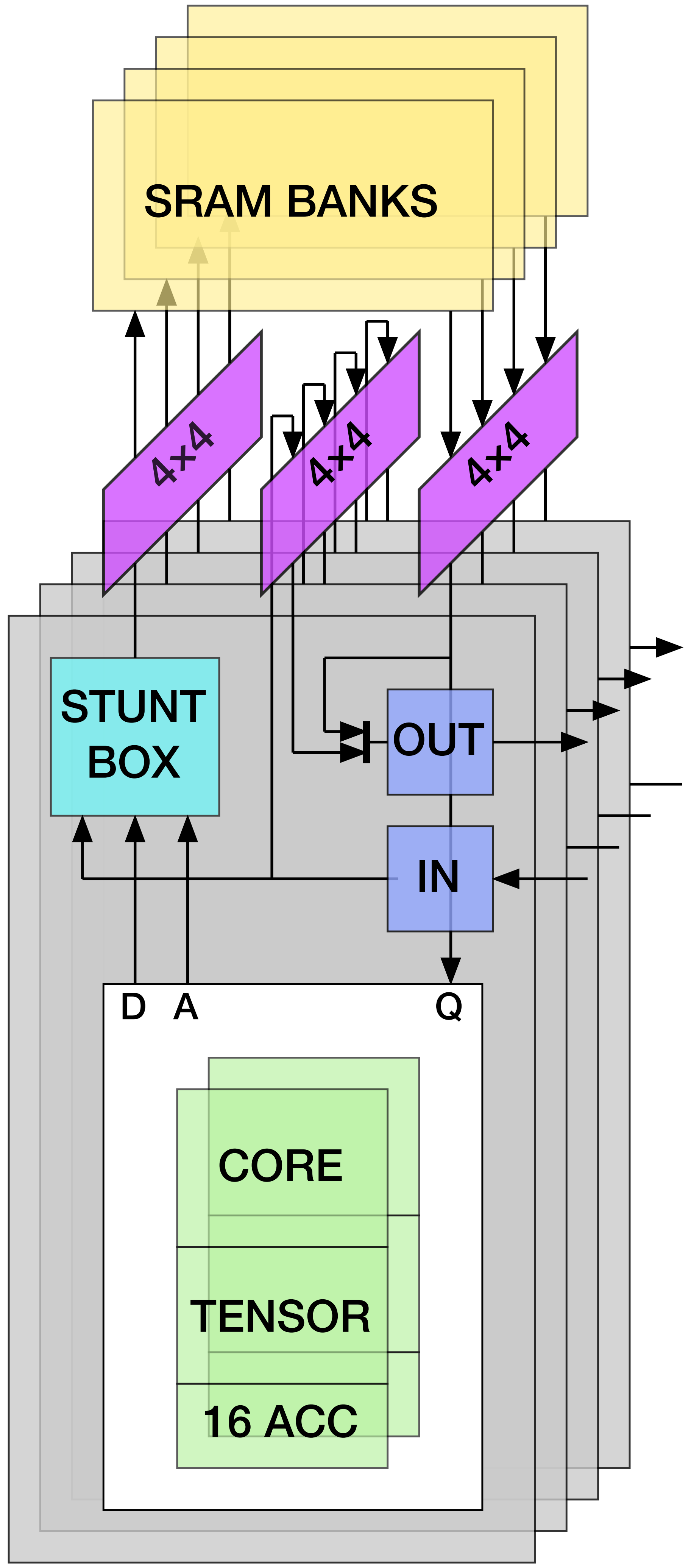


SpMV++

ML Example

Specialized machine learning accelerator

- 4-bit precision (logarithmic numbers)
  "Ultra-Low Precision 4-bit Training of Deep Neural Network," IBM Research
- 8 GB, 262 TOPS
  4-bit multiply, higher precision accumulate
- 500 MHz, low VDD
  0.45V vs. 0.85V for 1 GHz operation
- 3.9 TOPS/W energy efficiency



Mainstream Programming Model

Cluster of computers → array of tiles

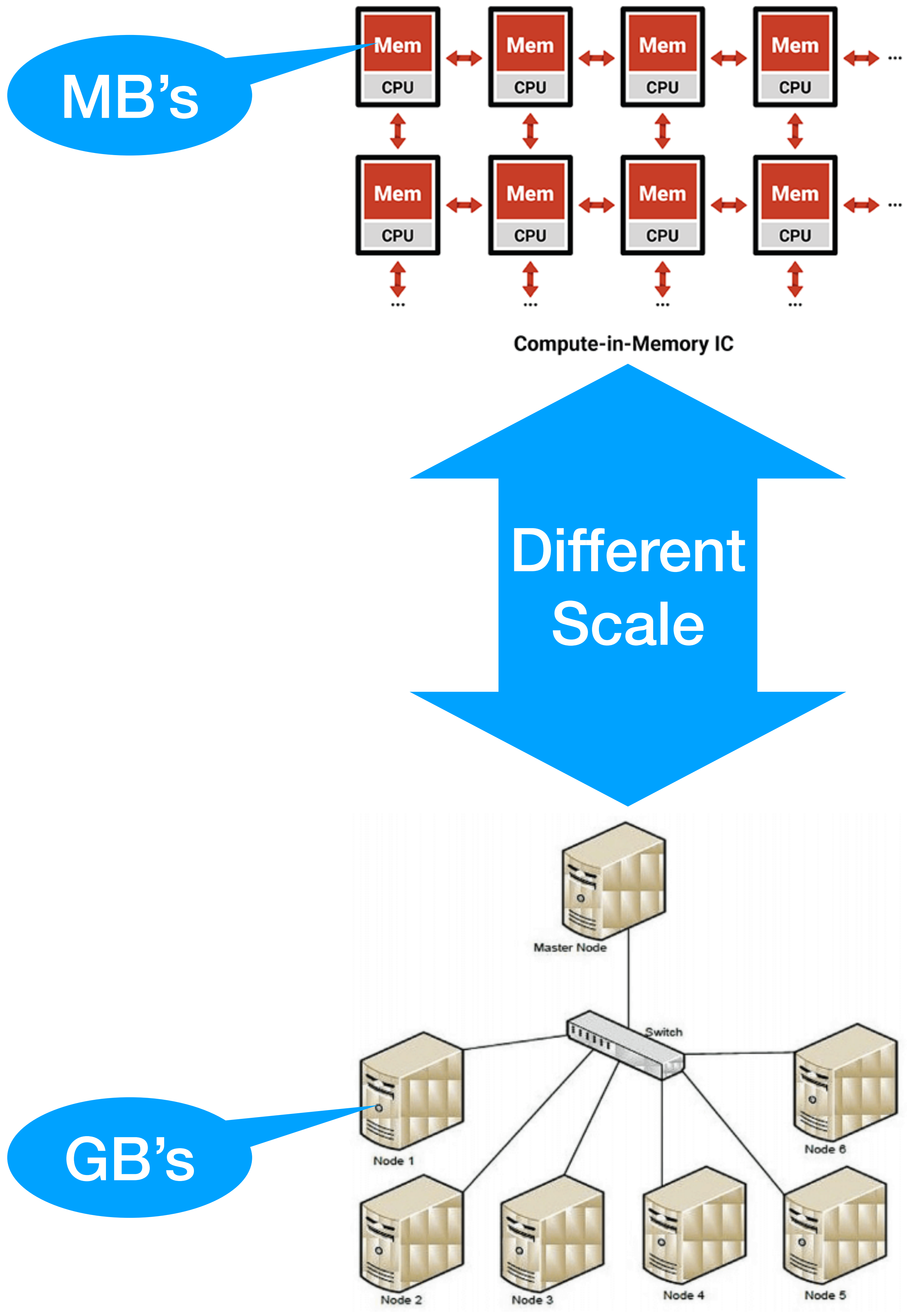
- Multicores with shared coherent memory (SMP)

Network → network on chip (NoC)

- Application specific interconnection topology
- Front-end computers → host processors

Same application programming interface (API)

- Linux threads, sockets, RDMA…



Codesign Methodology

1. Develop algorithm on standard cluster of SMP servers
   - Standard Linux API, but respecting target node memory size
2. Simulate near-memory RISC-V SMP cluster
   - Parallel “thread per core, process per node” simulator (next slide)
3. Develop co-processor with custom instructions
   - Refine codesign and validate performance improvement

Cavatools

Caveat — RISC-V user-mode Linux virtual machine

- Thread-per-core execution-driven simulator, ≈100 MIPS
- Shared memory (eg. OpenMP), multiple nodes (eg. MPI)

Custom instruction definition

- Spec → compiler intrinsic, asm, sim

```
fd += fs3 * load(rs1+imm)
```

fs3	imm[7:2]	op2	rs1	op3	fd	opcode

Open source (this work was partially supported by BSC)

More details in ICS Conference presentation “Cavatools: Parallel Architecture Simulator for RISC-V” (PDF)

Caveat Simulation Paradigm

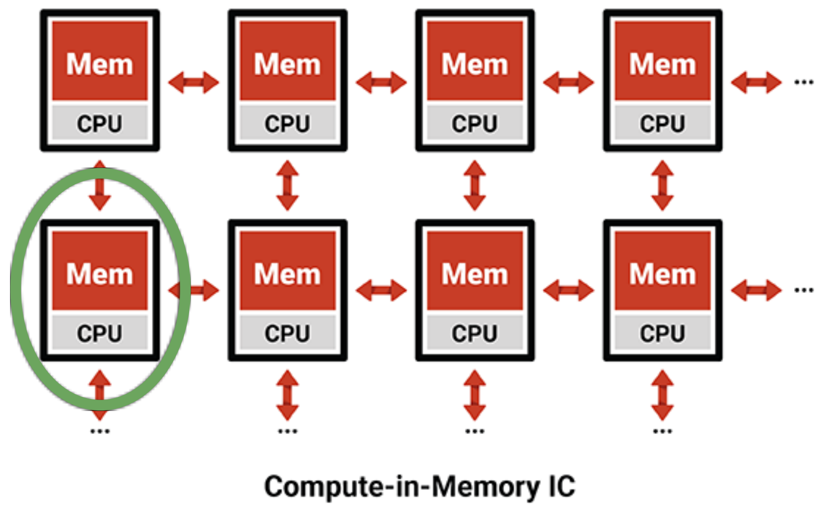
Array of tiles ➞ Linux processes

- SMP cores ➞ Linux threads
- RISC-V AMO ➞ x86 CMPXCHG

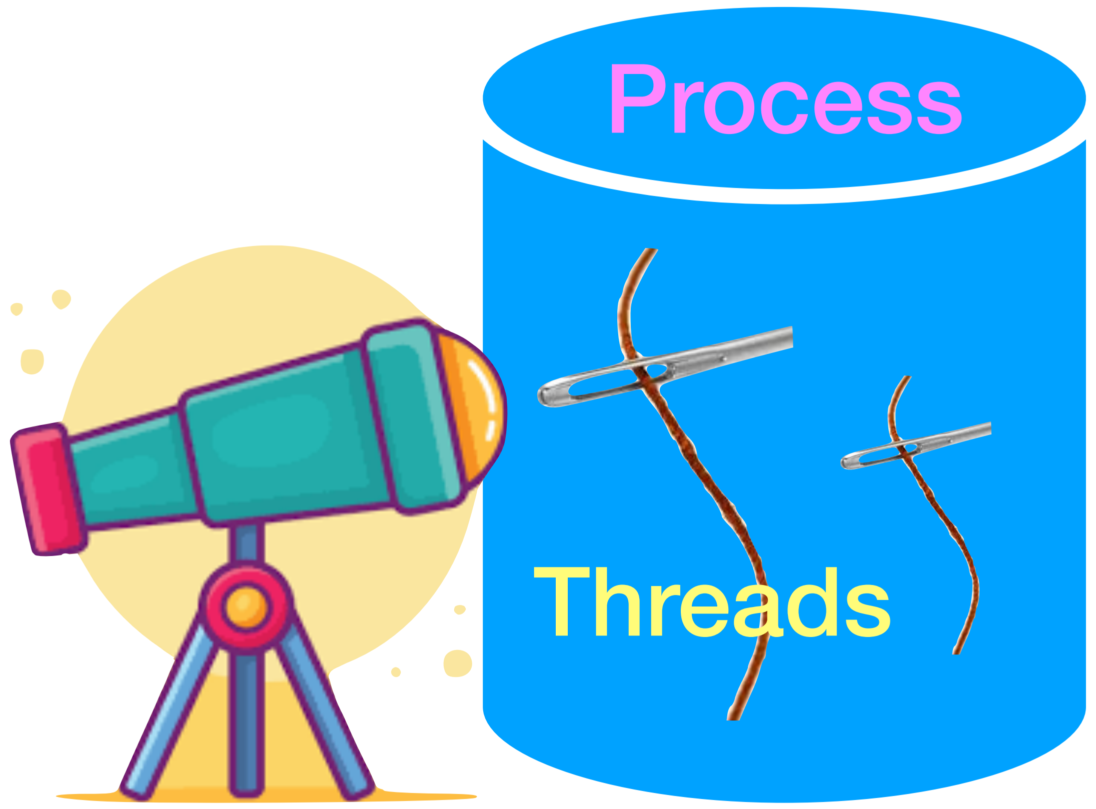
Network on Chip ➞ Linux sockets

- Messages ➞ Linux read(), write()

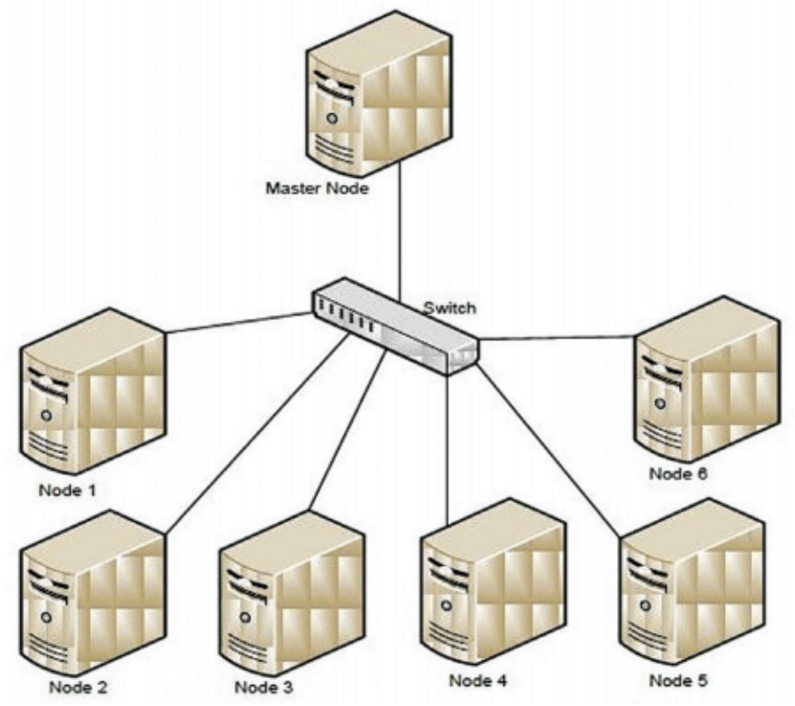
Erised — realtime visualization



Accelerator



Simulator



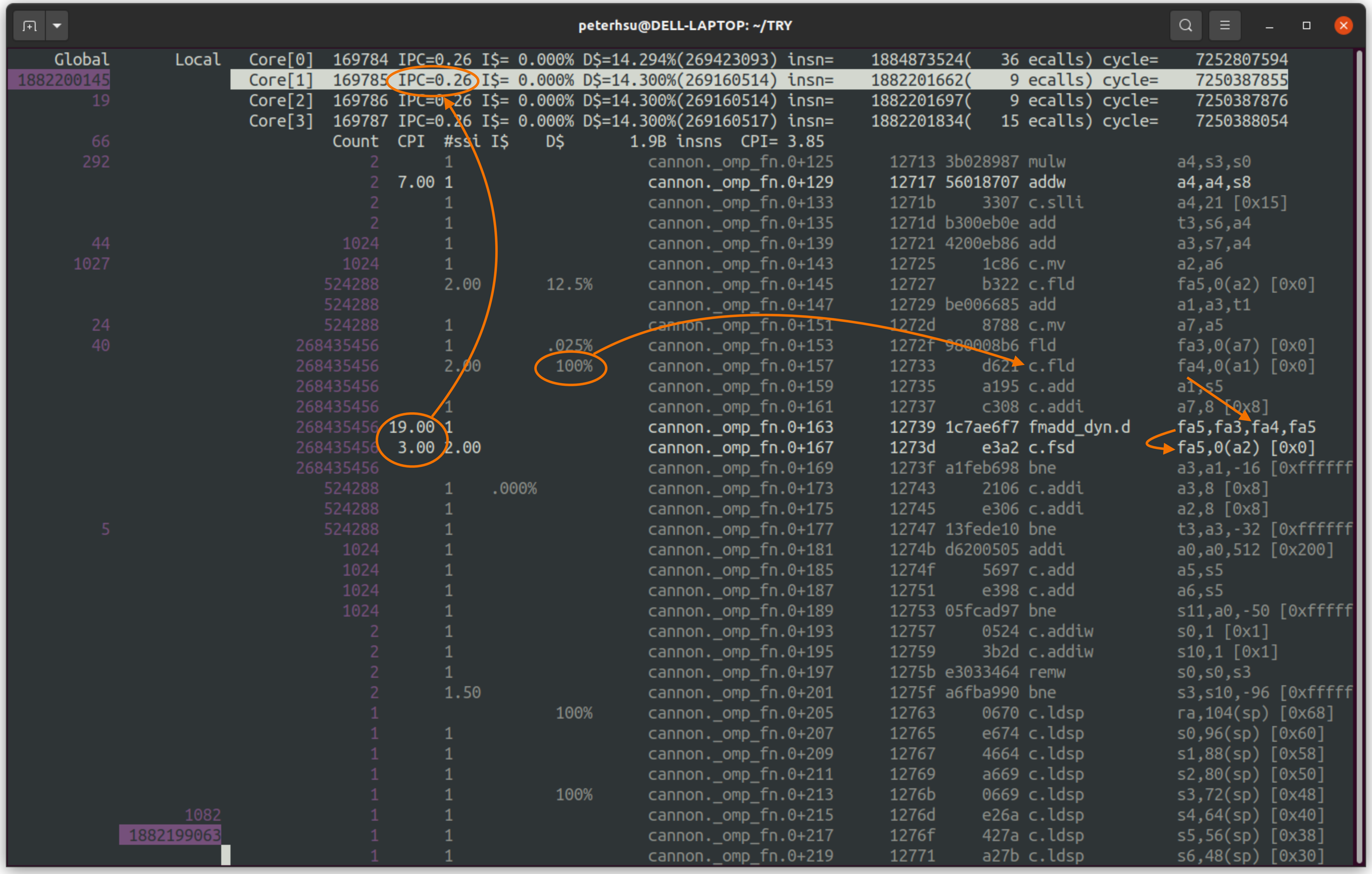
Erised Performance Visualizer

Within a tile

- Pipeline stalls
- Instruction Buffer Misses

Across chip

- System calls (IPC)
- Message queues



```
     Global       Local  Core[0]  169784 IPC=0.26 I$= 0.000% D$=14.294%(269423093) insn=    1884873524(  36 ecalls) cycle=    7252807594
1882200145               Core[1]  169785 IPC=0.26 I$= 0.000% D$=14.300%(269160514) insn=    1882201662(   9 ecalls) cycle=    7250387855
        19               Core[2]  169786 IPC=0.26 I$= 0.000% D$=14.300%(269160514) insn=    1882201697(   9 ecalls) cycle=    7250387876
                         Core[3]  169787 IPC=0.26 I$= 0.000% D$=14.300%(269160517) insn=    1882201834(  15 ecalls) cycle=    7250388054
        66                         Count  CPI  #ssi I$    D$       1.9B insns  CPI= 3.85
       292                             2       1                       cannon._omp_fn.0+125     12713 3b028987 mulw            a4,s3,s0
                                       2  7.00 1                       cannon._omp_fn.0+129     12717 56018707 addw            a4,a4,s8
                                       2       1                       cannon._omp_fn.0+133     1271b     3307 c.slli          a4,21 [0x15]
                                       2       1                       cannon._omp_fn.0+135     1271d b300eb0e add             t3,s6,a4
        44                          1024       1                       cannon._omp_fn.0+139     12721 4200eb86 add             a3,s7,a4
      1027                          1024       1                       cannon._omp_fn.0+143     12725     1c86 c.mv            a2,a6
                                  524288       2.00         12.5%      cannon._omp_fn.0+145     12727     b322 c.fld           fa5,0(a2) [0x0]
                                  524288                               cannon._omp_fn.0+147     12729 be006685 add             a1,a3,t1
        24                        524288       1                       cannon._omp_fn.0+151     1272d     8788 c.mv            a7,a5
        40                     268435456       1            .025%      cannon._omp_fn.0+153     1272f 980008b6 fld             fa3,0(a7) [0x0]
                               268435456       2.00          100%      cannon._omp_fn.0+157     12733     d621 c.fld           fa4,0(a1) [0x0]
                               268435456                               cannon._omp_fn.0+159     12735     a195 c.add           a1,s5
                               268435456       1                       cannon._omp_fn.0+161     12737     c308 c.addi          a7,8 [0x8]
                               268435456 19.00 1                       cannon._omp_fn.0+163     12739 1c7ae6f7 fmadd_dyn.d     fa5,fa3,fa4,fa5
                               268435456  3.00 2.00                    cannon._omp_fn.0+167     1273d     e3a2 c.fsd           fa5,0(a2) [0x0]
                               268435456                               cannon._omp_fn.0+169     1273f a1feb698 bne             a3,a1,-16 [0xffffff
                                  524288       1     .000%             cannon._omp_fn.0+173     12743     2106 c.addi          a3,8 [0x8]
                                  524288       1                       cannon._omp_fn.0+175     12745     e306 c.addi          a2,8 [0x8]
         5                        524288       1                       cannon._omp_fn.0+177     12747 13fede10 bne             t3,a3,-32 [0xffffff
                                    1024       1                       cannon._omp_fn.0+181     1274b d6200505 addi            a0,a0,512 [0x200]
                                    1024       1                       cannon._omp_fn.0+185     1274f     5697 c.add           a5,s5
                                    1024       1                       cannon._omp_fn.0+187     12751     e398 c.add           a6,s5
                                    1024       1                       cannon._omp_fn.0+189     12753 05fcad97 bne             s11,a0,-50 [0xfffff
                                       2       1                       cannon._omp_fn.0+193     12757     0524 c.addiw         s0,1 [0x1]
                                       2       1                       cannon._omp_fn.0+195     12759     3b2d c.addiw         s10,1 [0x1]
                                       2       1                       cannon._omp_fn.0+197     1275b e3033464 remw            s0,s0,s3
                                       2       1.50                    cannon._omp_fn.0+201     1275f a6fba990 bne             s3,s10,-96 [0xfffff
                                       1                     100%      cannon._omp_fn.0+205     12763     0670 c.ldsp          ra,104(sp) [0x68]
                                       1       1                       cannon._omp_fn.0+207     12765     e674 c.ldsp          s0,96(sp) [0x60]
                                       1       1                       cannon._omp_fn.0+209     12767     4664 c.ldsp          s1,88(sp) [0x58]
                                       1       1                       cannon._omp_fn.0+211     12769     a669 c.ldsp          s2,80(sp) [0x50]
                                       1       1             100%      cannon._omp_fn.0+213     1276b     0669 c.ldsp          s3,72(sp) [0x48]
                      1082             1       1                       cannon._omp_fn.0+215     1276d     e26a c.ldsp          s4,64(sp) [0x40]
            1882199063                 1       1                       cannon._omp_fn.0+217     1276f     427a c.ldsp          s5,56(sp) [0x38]
                                       1       1                       cannon._omp_fn.0+219     12771     a27b c.ldsp          s6,48(sp) [0x30]
```

Agenda

1. Economics & technology
2. Accelerator architecture
3. Examples
4. Programming model & evaluation
5. Memory technologies
6. Conclusion

Optimized SRAM process is less expensive than ASIC, but 7nm SRAM main memory is still quite expensive

We need a path to more affordable memory technology

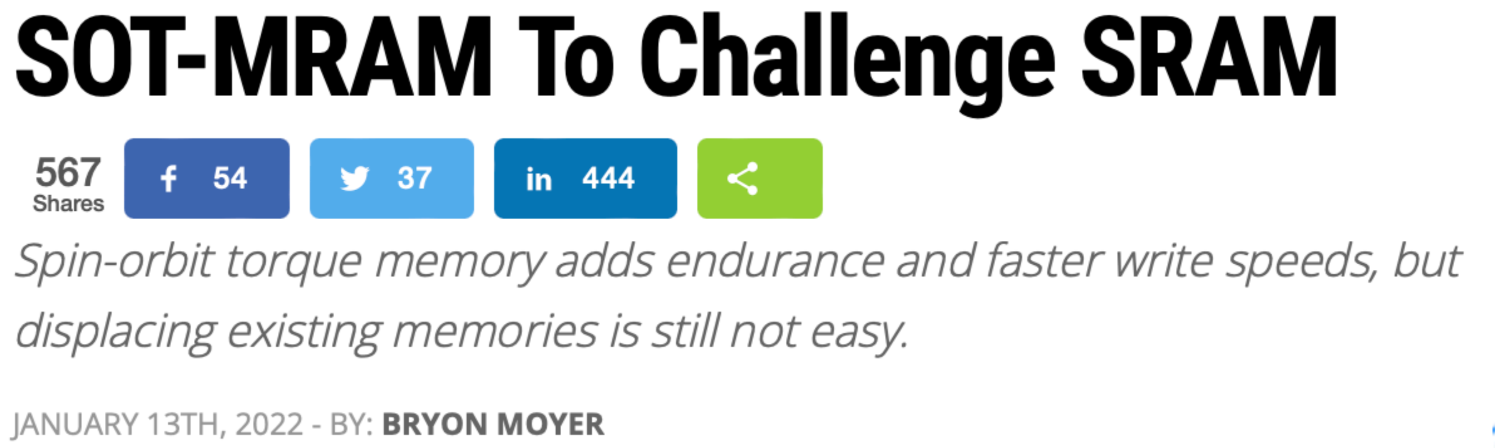
Magnetoresistive Memory

Leverage mature process for low cost

- MTJ extra module in standard CMOS
- Similar to RF, Analog process modules

Attractive characteristics

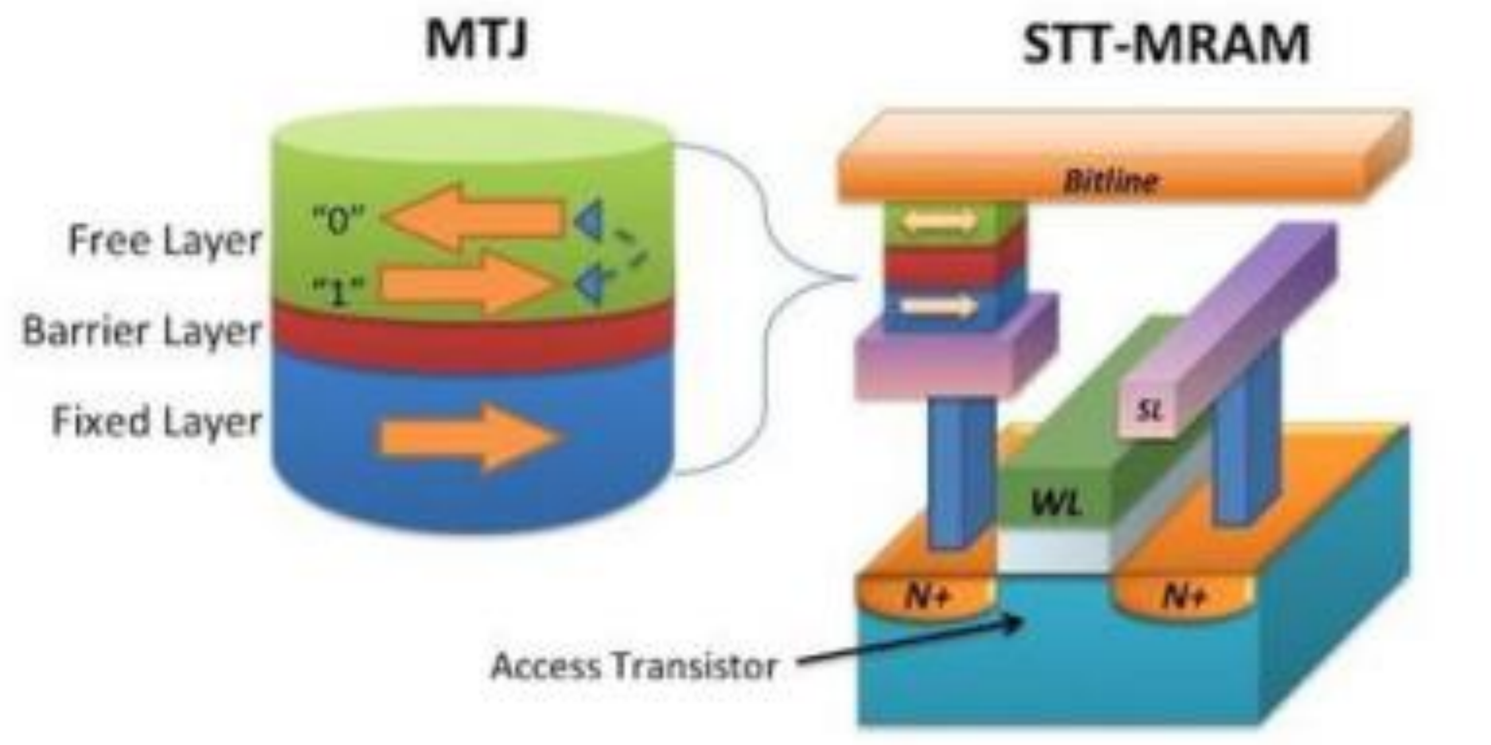
- Read time, energy ≈ SRAM (but write ↑)
- Wear-out is no longer a problem

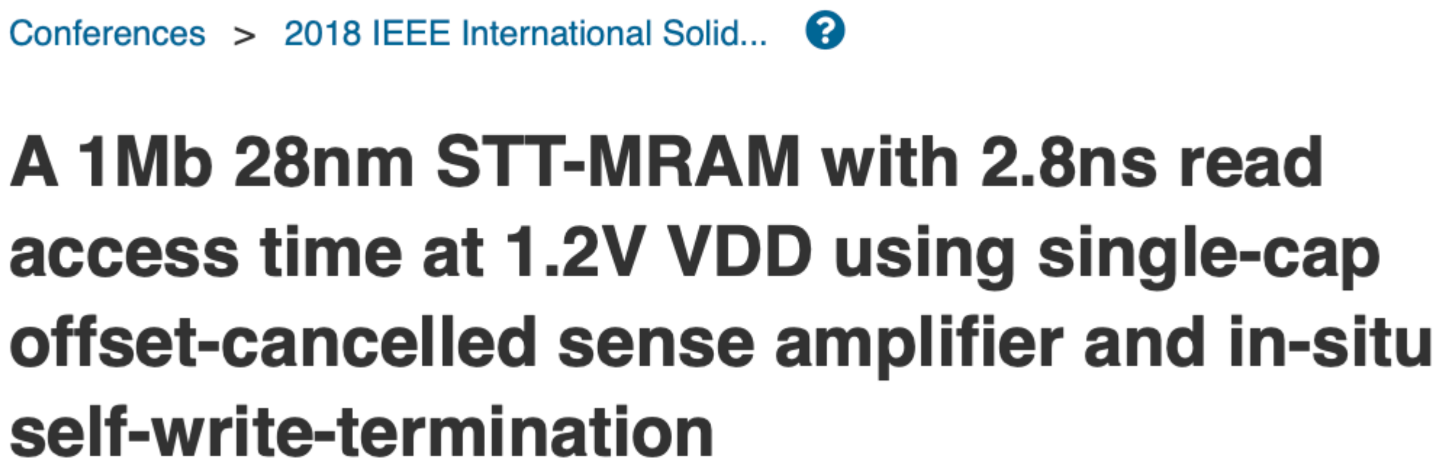


SOT-MRAM To Challenge SRAM

Spin-orbit torque memory adds endurance and faster write speeds, but displacing existing memories is still not easy.

JANUARY 13TH, 2022 - BY: **BRYON MOYER**





Conferences > 2018 IEEE International Solid...

A 1Mb 28nm STT-MRAM with 2.8ns read access time at 1.2V VDD using single-cap offset-cancelled sense amplifier and in-situ self-write-termination

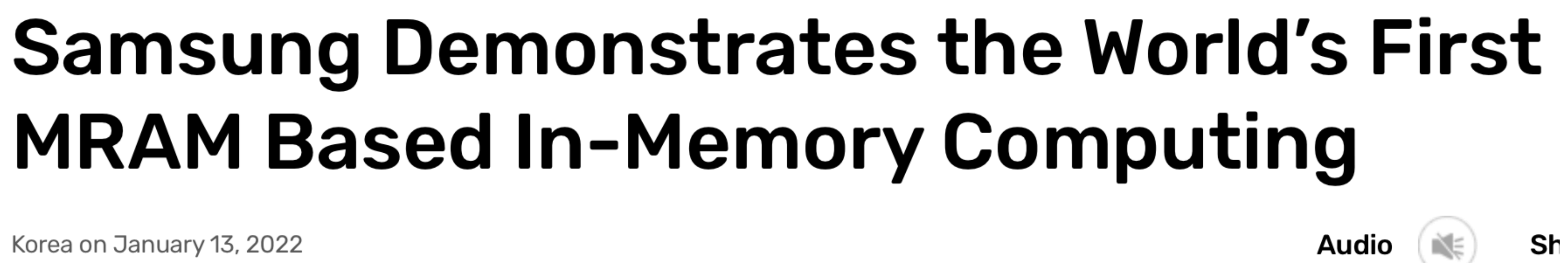
Reducing Memory Cost

SRAM 7nm ≈ 2 MB/mm² (TSMC)

- Scaling less than logic gates

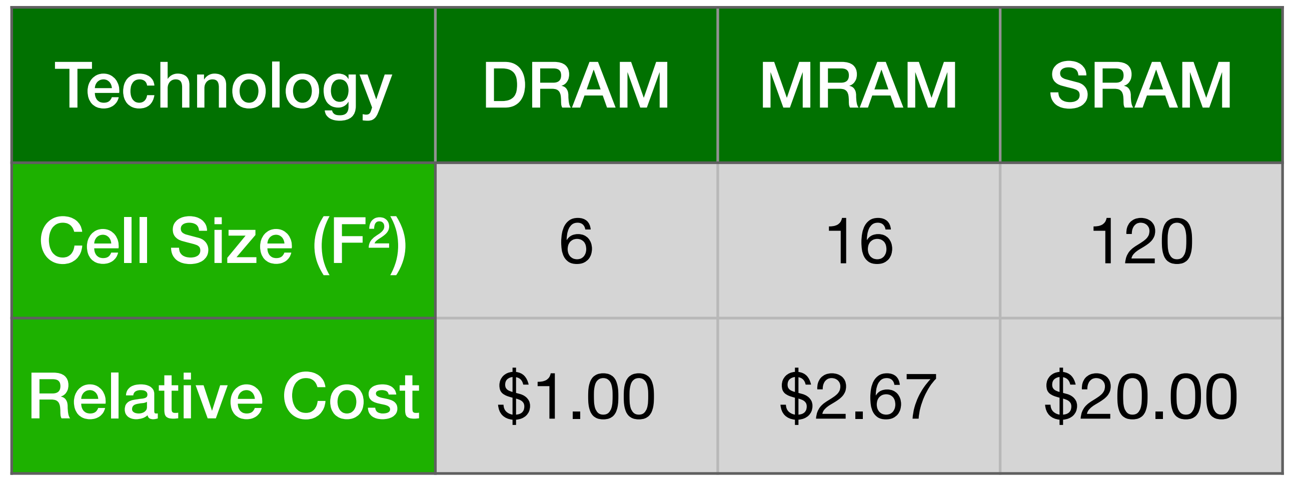
MRAM 28nm > 1 MB/mm² today

- Nonvolatile → mobile devices
- 3D (like XPoint™) in future

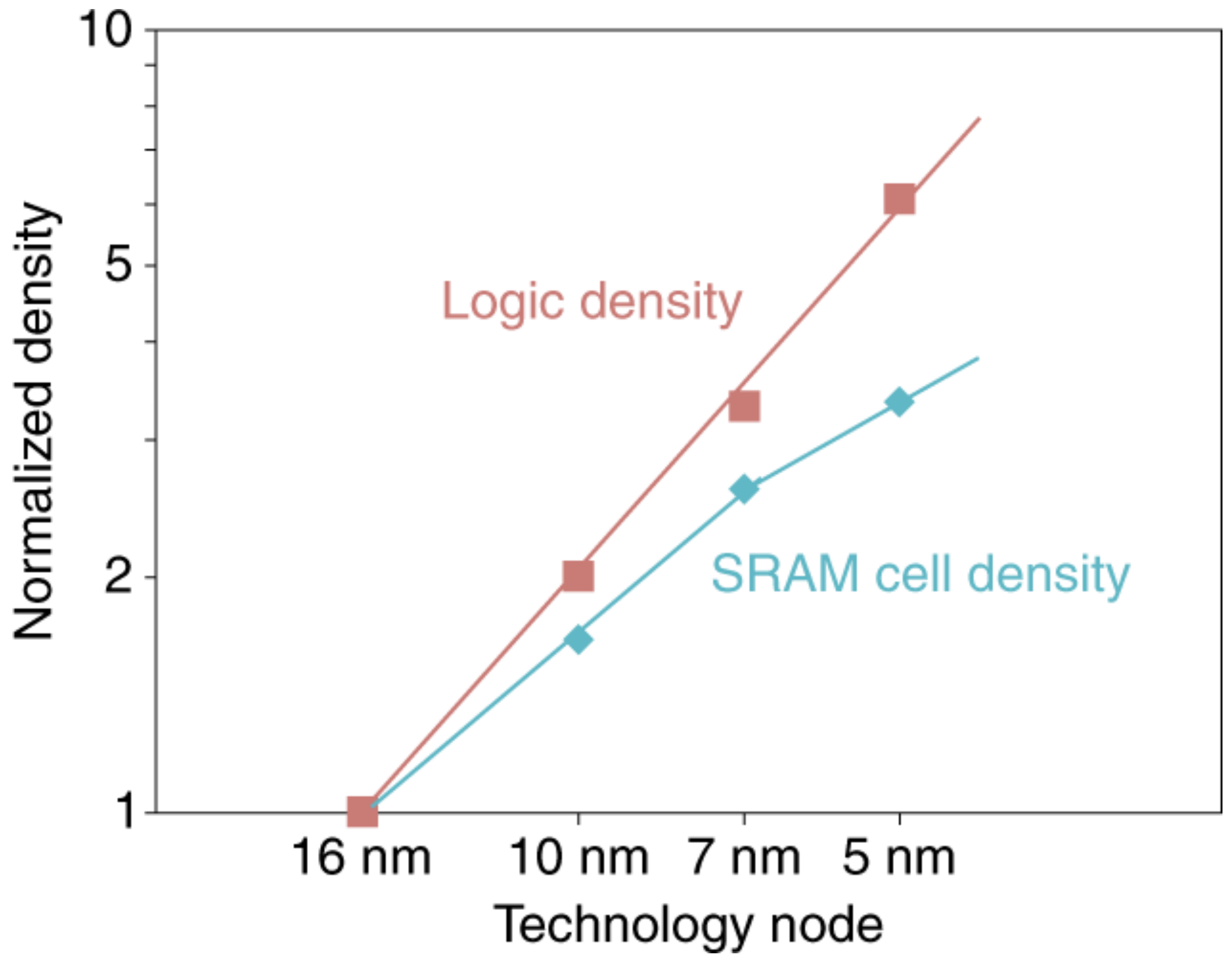


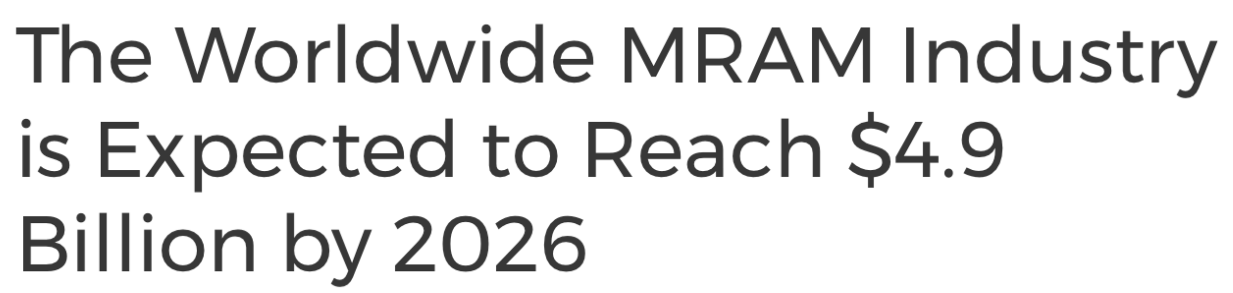
Samsung Demonstrates the World's First MRAM Based In-Memory Computing

Korea on January 13, 2022

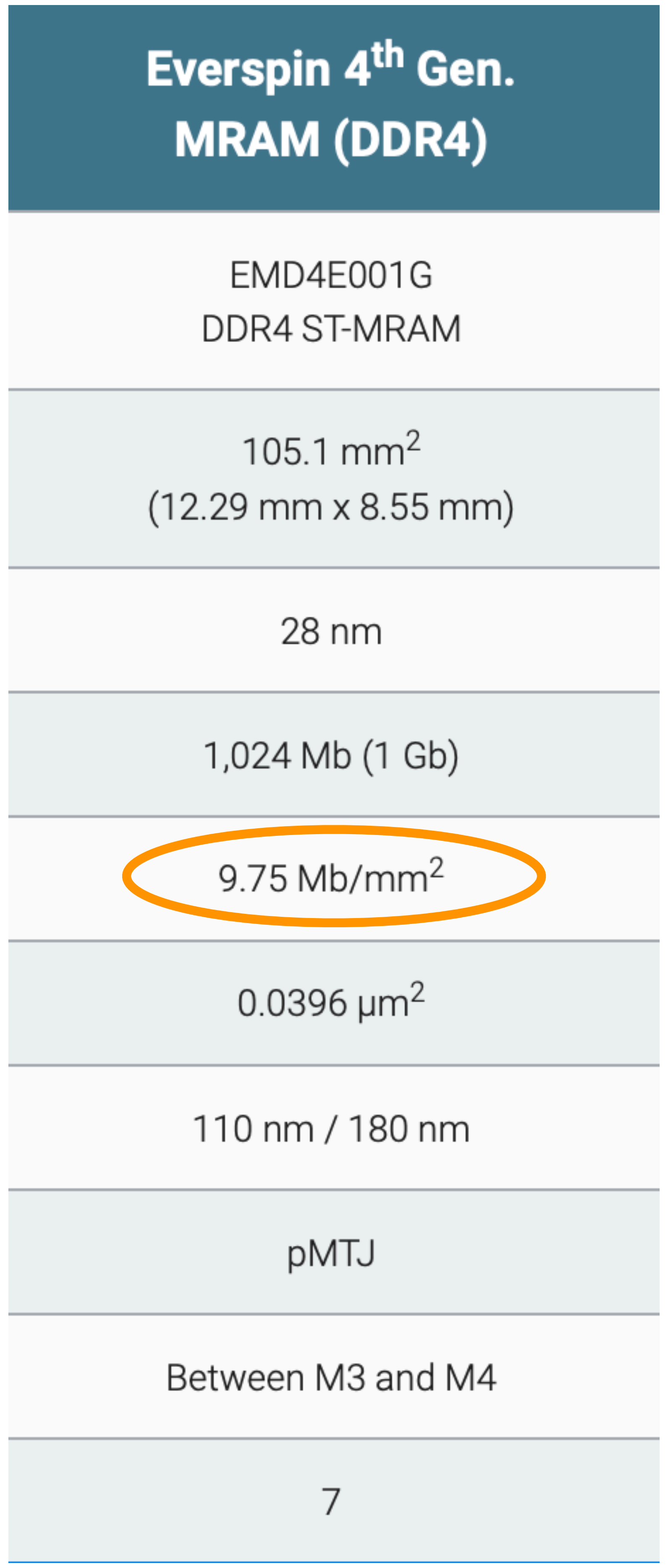


Technology	DRAM	MRAM	SRAM
Cell Size (F²)	6	16	120
Relative Cost	$1.00	$2.67	$20.00





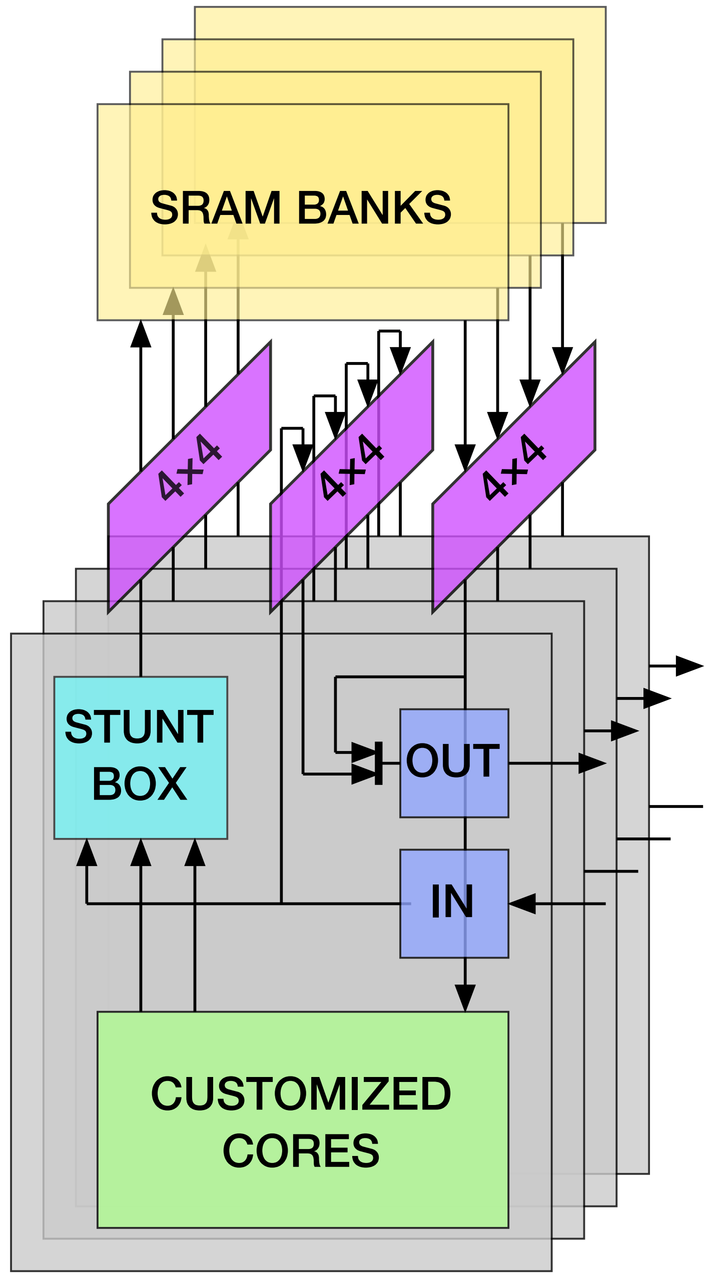
The Worldwide MRAM Industry is Expected to Reach $4.9 Billion by 2026

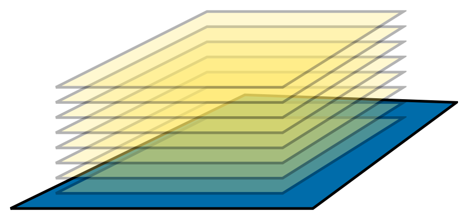


Everspin 4th Gen. MRAM (DDR4)
EMD4E001G DDR4 ST-MRAM
105.1 mm² (12.29 mm x 8.55 mm)
28 nm
1,024 Mb (1 Gb)
9.75 Mb/mm²
0.0396 μm²
110 nm / 180 nm
pMTJ
Between M3 and M4
7

Summary

We provide RISC-V platform architecture
High bandwidth, low latency SRAM memory
Standard Linux programming environment
Simulation tools to validate performance





You design arithmetic, interconnect, software for your application

Conclusion

- RISC-V enables hardware innovation but development cost limits creativity
- Proposed paradigm for near-memory accelerator using state-of-the-art memory with mature logic technology
- Enable commercially competitive accelerators based on novel ideas by smaller organizations with limited resources

RISC-V

Thank You

Abstract: The RISC-V architecture has opened new opportunities for many people to innovate in computer design. However to design a chip that can compete in the marketplace against veteran industry computer designers with their vast resources is still a formidable challenge. We propose a solution for specialized accelerators with near-memory processing architectures. We observe the critical technology is the embedded memory because it consumes most of the silicon area and determines the power/bandwidth of the chip. If instead memory is stacked on top of the logic chip, then a less dense, lower cost mature technology can be used for the logic. Communication wire power will be lower because the through-silicon via (TSV) interconnect traverses a much smaller distance, offsetting the lower power efficiency of mature logic technology. Design cost of the re-useable hi-tech memory chip is amortized across multiple accelerators. We believe this approach can help smaller organizations with limited resources design commercially competitive novel accelerators.

Bio: Peter Hsu received his Ph.D. from University of Illinois Urbana-Champaige. He started work at IBM T. J. Watson Research Center on the 801 Project. He joined SGI in 1990 as architect of MIPS R8000 TFP microprocessor; the chip powered fifty TOP500 supercomputers in 1994. Peter co-founded ArtX in 1997 to develop the Nintendo GameCube video game console. He joined Oracle Labs in 2011 as Architect and built a fifty thousand core parallel SQL database accelerator. Peter moved to Europe in 2018 and was visiting professor at EPFL University in Switzerland, then senior researcher at the Barcelona Supercomputing Center. In 2022 he started a consulting company in Spain, Peter Hsu & Associates, S.L.U.