

Are You Ready for 1000-Way Parallelism on a Single Chip?

Andreas Olofsson

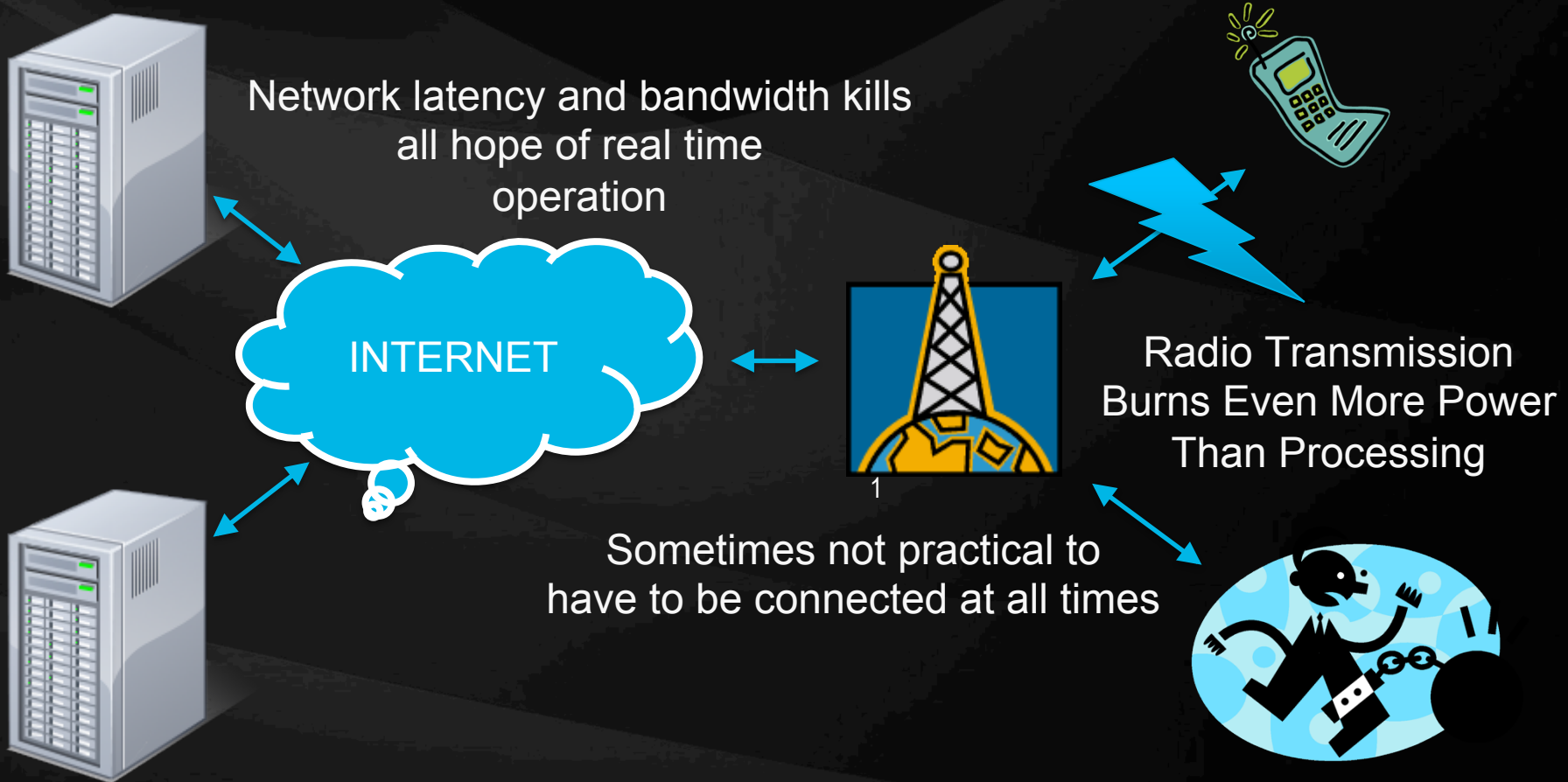
CODEMESH 2013



Who needs more Performance?

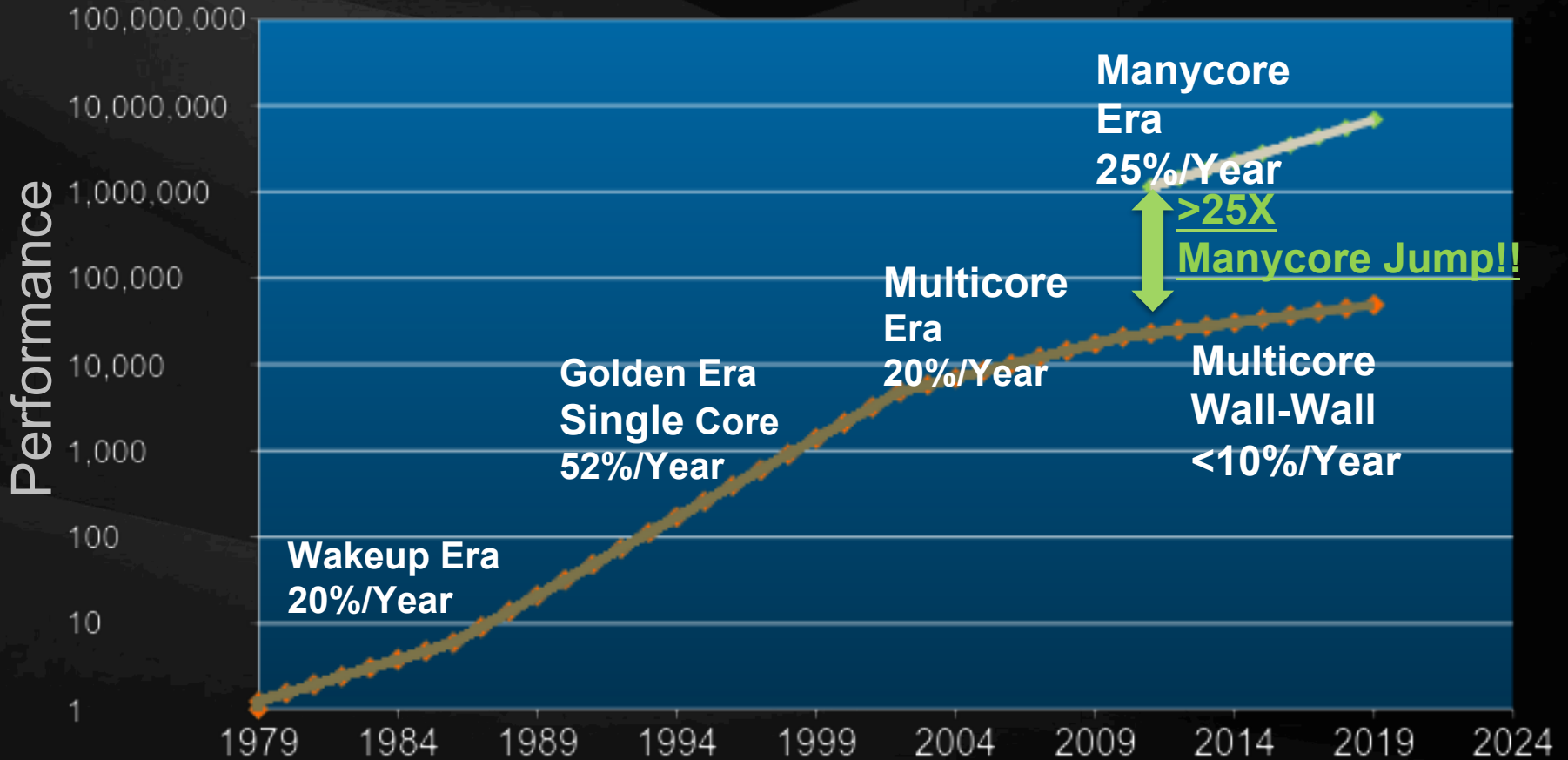


The Magical Cloud is NOT The solution!

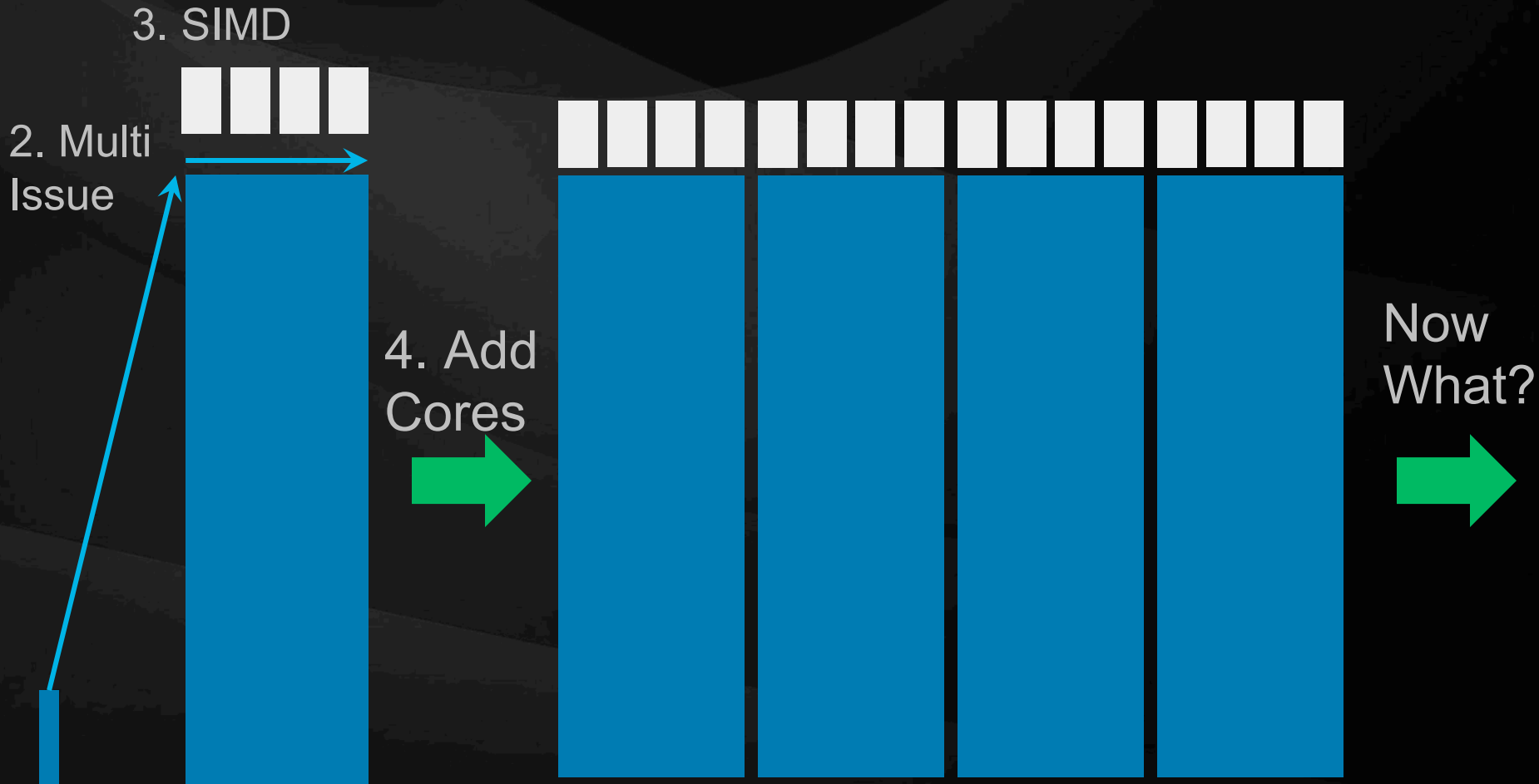


Locality, locality, locality...

The Free Lunch is Over!



Running out of HW Tricks..



1. Scale
Frequency

4. Add
Cores

Now
What?

10 Trends that Will Shape the Future of Computing

Power Consumption

Latency Wall

Memory Bottlenecks

Yield Issues

Frequency Wall

Time to Market

Wiring

Software Complexity

Thermal Density

Amdahl's Law

Any Reason to Think the Future of Computing is NOT Parallel?

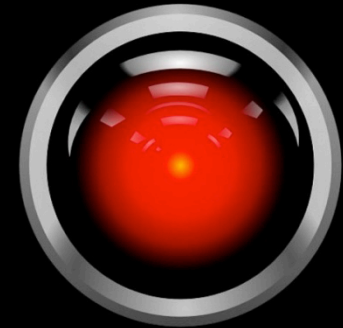
No Computing

Parallel Computing

No Electronic Computing
-1943

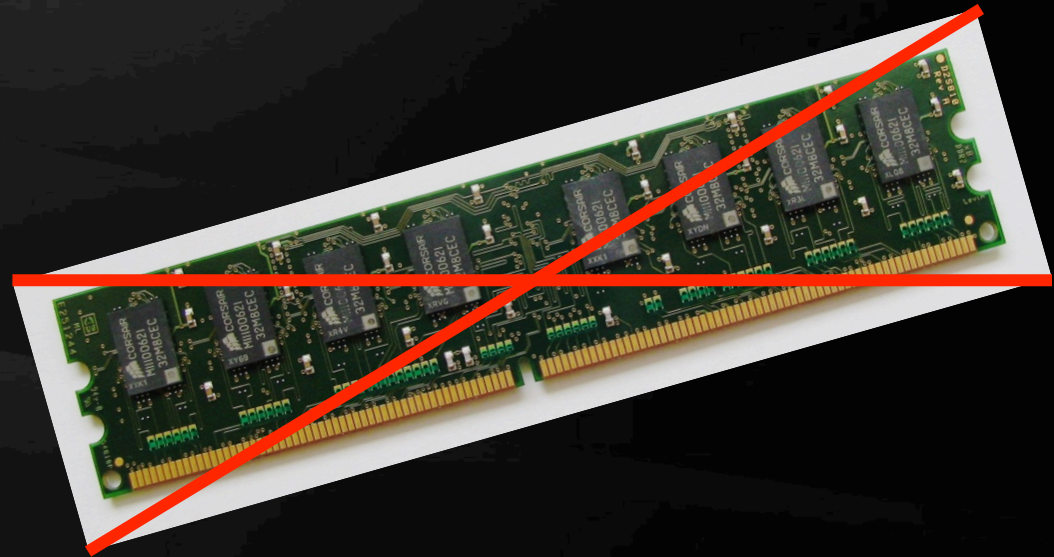
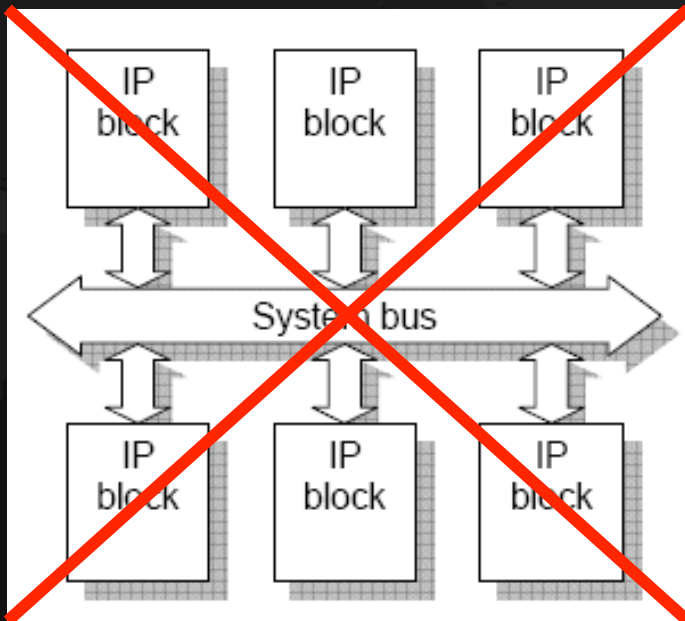
“Von Neumann Age”
Serial Computing
1943-2013?

Parallel Computing
2013-??

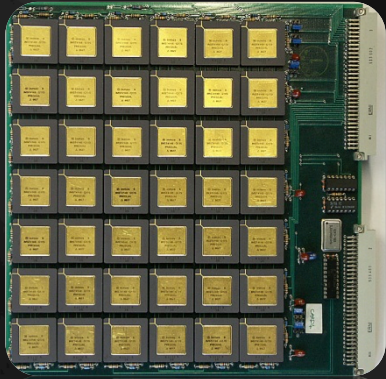


How to Scale?

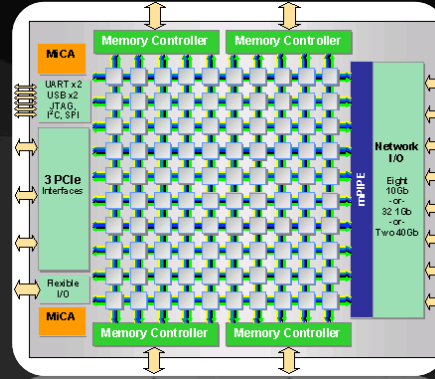
Reduce shared and critical resources to zero
(True for SW and HW)



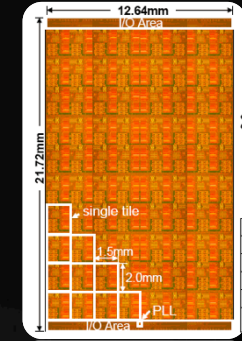
A Brief History of Parallel Computing



Transputer
(1984)



Tiler
(2006)

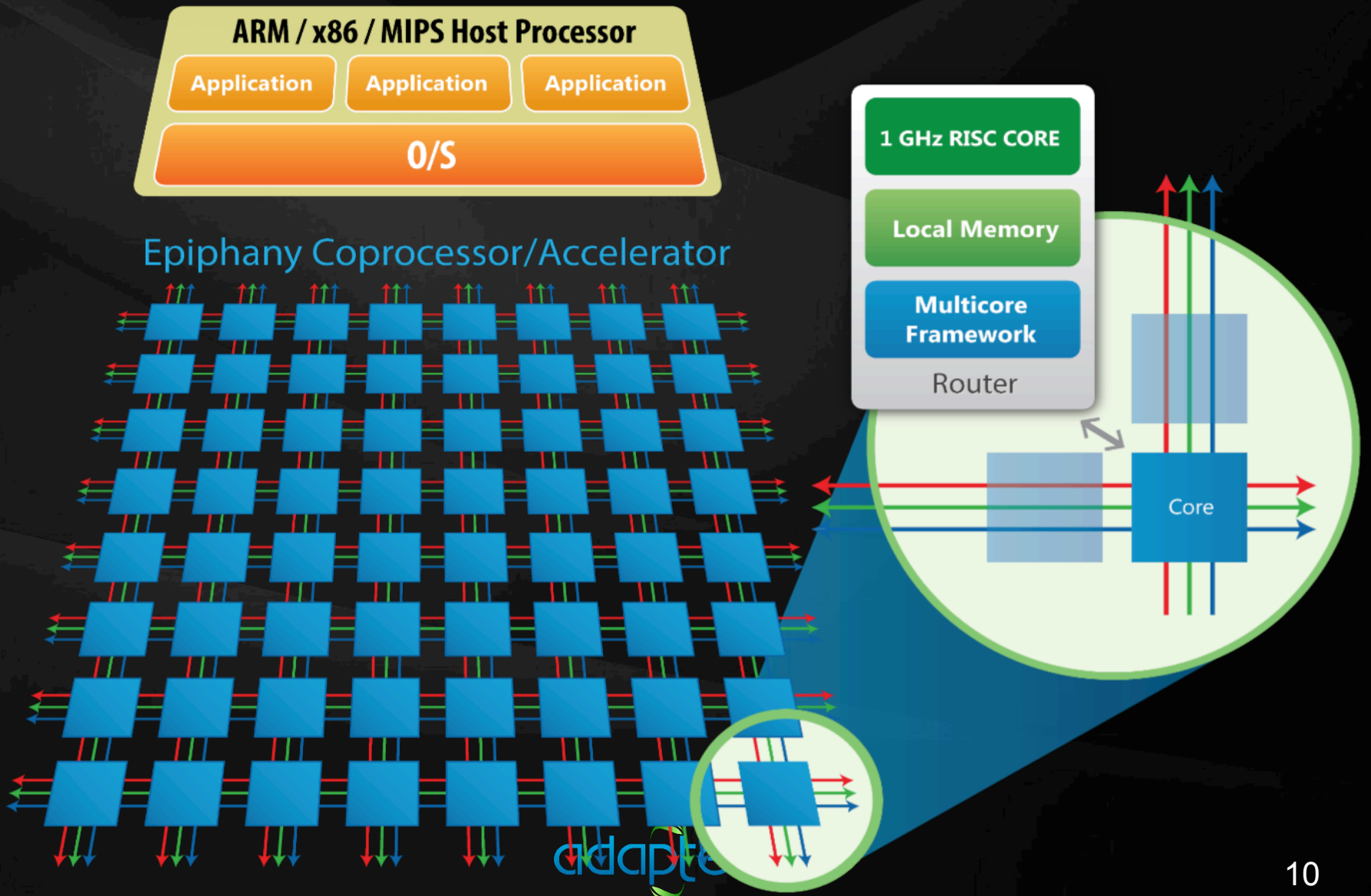


Teraflop (Intel)
(2007)

- Ambric
- Cognivue
- Icera
- PACT
- SiCortex
- Asocs
- Coherent Logix
- Intrinsity
- Picochip
- Silicon Hive
- Aspex
- CPUtech
- IP-flex
- Plurality
- Spiral Gateway
- Axis Semi
- Cradle
- Mathstar
- Quicksilver
- Stream Processors
- BOPS
- Cswitch
- Morphics
- Rapport
- Stretch
- Boston Circuits
- ElementCXI
- Movidius
- Recore
- Venray
- Brightscale
- Greenarrays
- Octasic
- Sandbridge
- Xelerated
- Chameleon
- Inmos
- SiByte
- XMOS
- Clearspeed
- Intelliasys
- SiCortex
- Zililabs



The Epiphany Multicore Architecture



Pragmatic Architecture Tradeoffs

IN

- Shared memory architecture
- Dual issue RISC processors
- 64 entry register file
- 32-128KB per core memory
- Multi-banked local memory
- Packet based Mesh NOC
- 32 Bit IEEE float / int
- Memory protection
- Timers, Interrupts, DMAs

OUT

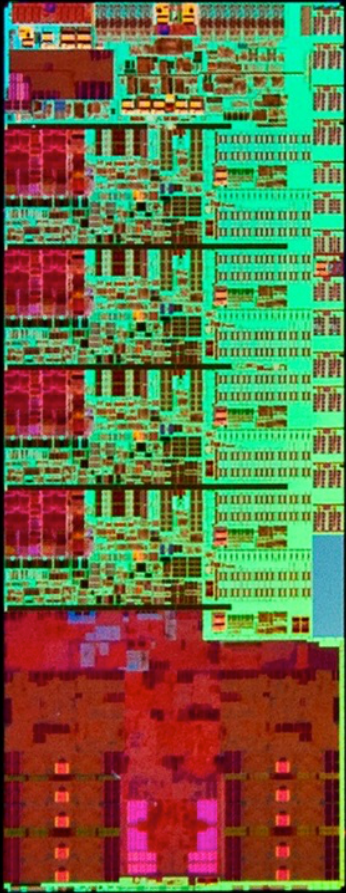
- Special purpose instructions
- Hardware caching
- SIMD
- Optimized read accesses
- Memory management unit
- Strict memory ordering

Key Epiphany Software Considerations

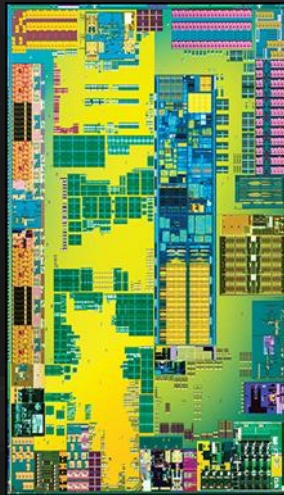
- Critical code and data must sit in core's local memory (<32KB)
- Optimized (but not restricted) to be a coprocessor
- Flat globally shared memory (NUMA) map (SRAM not cache)
- Row, column based "physical world" address mapping
- Remote writes MUCH faster than remote reads
- Off-chip bandwidth is VERY expensive
- On-chip core-to core communication is cheap and plenty
- Beware of read/write remote memory access races

Apples and Oranges...

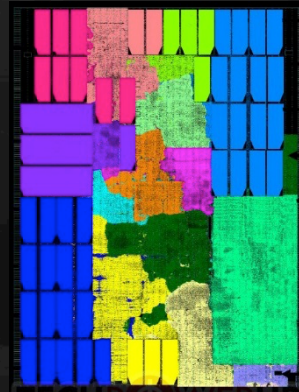
100 Epiphany CPU cores
fit in the space of one
Haswell CPU core!



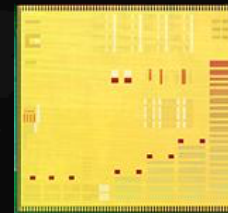
Intel Haswell
14.5mm²



Intel Atom
5.6mm²



AMD Jaguar
3.1mm²



ARM A15
1.62mm²



ARM A7
0.45mm²



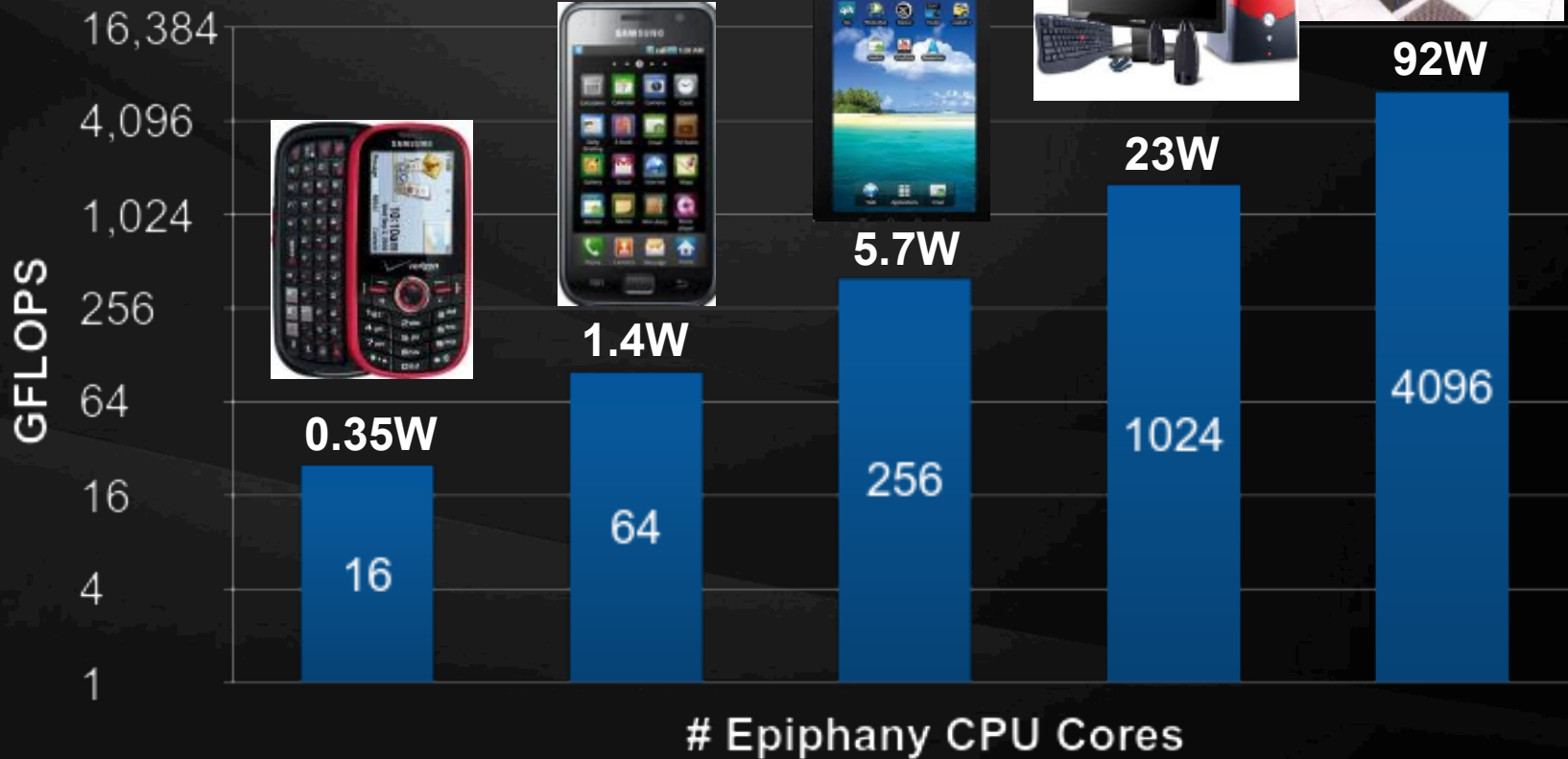
Epiphany
0.13mm²

Processor Comparison

Technology	FPGA	DSP	GPU	CPU	Manycore	Manycore
Process	28nm	40nm	28nm	32nm	28nm	28nm
Programming	VHDL	C++/Asmbly	CUDA	C/C++	C/C++	C/C++
Area (mm ²)	590	108	294	216	10	130
Price	\$6900	\$200	\$499	\$285	TBD	TBD
Chip Power (W)	40	22	135	130	2	25
CPUs	n/a	8	32	4	64	1024
Max GFLOPS	1500	160	3000	115	102	2048
Max GMACS	3000	320	n/a	n/a	51.2	1024
GHz * Cores	n/a	12	16	14.4	51.2	1024
L1 Memory	6MB	512KB	2.5MB	256KB	2MB	32MB
L2 Memory	n/a	4MB	512KB	1024KB	n/a	n/a

Epiphany: A Truly Scalable Architecture

A Single Unified Instruction Set Architecture!



Think 32KB is too small...no problem?

Memory	IEEE	Cores	Array Size	Frequency
32KB	SPF	16	2.45mm ²	800MHz
64KB	SPF	16	3.2mm ²	700MHz
64KB	DPF	16	3.5mm ²	600MHz
128KB	DPF	16	4.9mm ²	600MHz

How the \$#@% Do
We Program This
Thing?

Parallel Possibilities with Epiphany

- Run different program on each core
- Run the same program on each core (SPMD)
- Look for natural data tiling (in images, packets etc)
- Pipeline functions across different cores (send/receive)
- Keep the data in place and move functions around the chip
- Use a host send programs to the Epiphany coprocessor
- Use a framework like OpenCL
- Fork/join threads from an Epiphany core

Parallel Programming Frameworks

Erlang	SystemC	Intel TBB	Co-Fortran	Lisp	Janus
Scala	Haskell	Pragmas	Fortress	Hadoop	Linda
Smalltalk	CUDA	Clojure	UPC	PVM	Rust
Julia	OpenCL	Go	X10	Posix	XC
Occam	OpenHMPP	ParaSail	APL	Simulink	Charm++
Occam-pi	OpenMP	Ada	Labview	Ptolemy	StreamIt
Verilog	OpenACC	C++Amp	Rust	Sisal	Star-P
VHDL	Cilk	Chapel	MPI	MCAPI	??

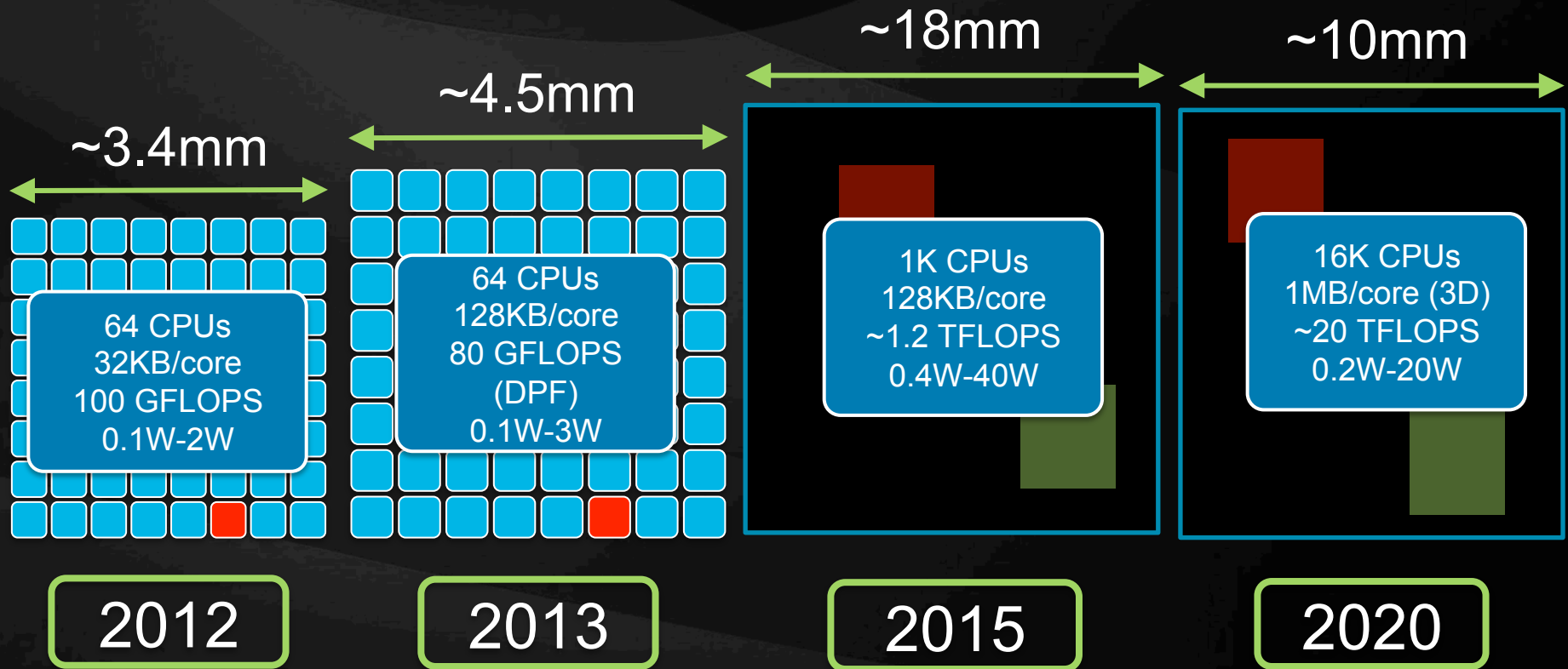
The Parallella Project

“Raspberry Pi for parallel”

- 16/64-core Epiphany CPU
- Dual-core A9 ARM SoC with FPGA
- 1GB RAM
- Ethernet, USB, HDMI, etc
- Linux/Android OS
- Credit card form factor
- 5 Watts!
- Open Source SDKs
- ~6,500 backers/customers
- \$99-\$199
- ..visit parallella.org



Epiphany Roadmap



Recommendations/Predictions

- No easy fix, need to rewrite whole stack for massive parallelism
- Get ready for explicit memory management
- Software should be processor agnostic
- Know where your bits are stored
- Hardware will fail, plan accordingly
- 1K core chips possible today and we will have 16K cores with 1MB/core by 2020 ...
get ready!