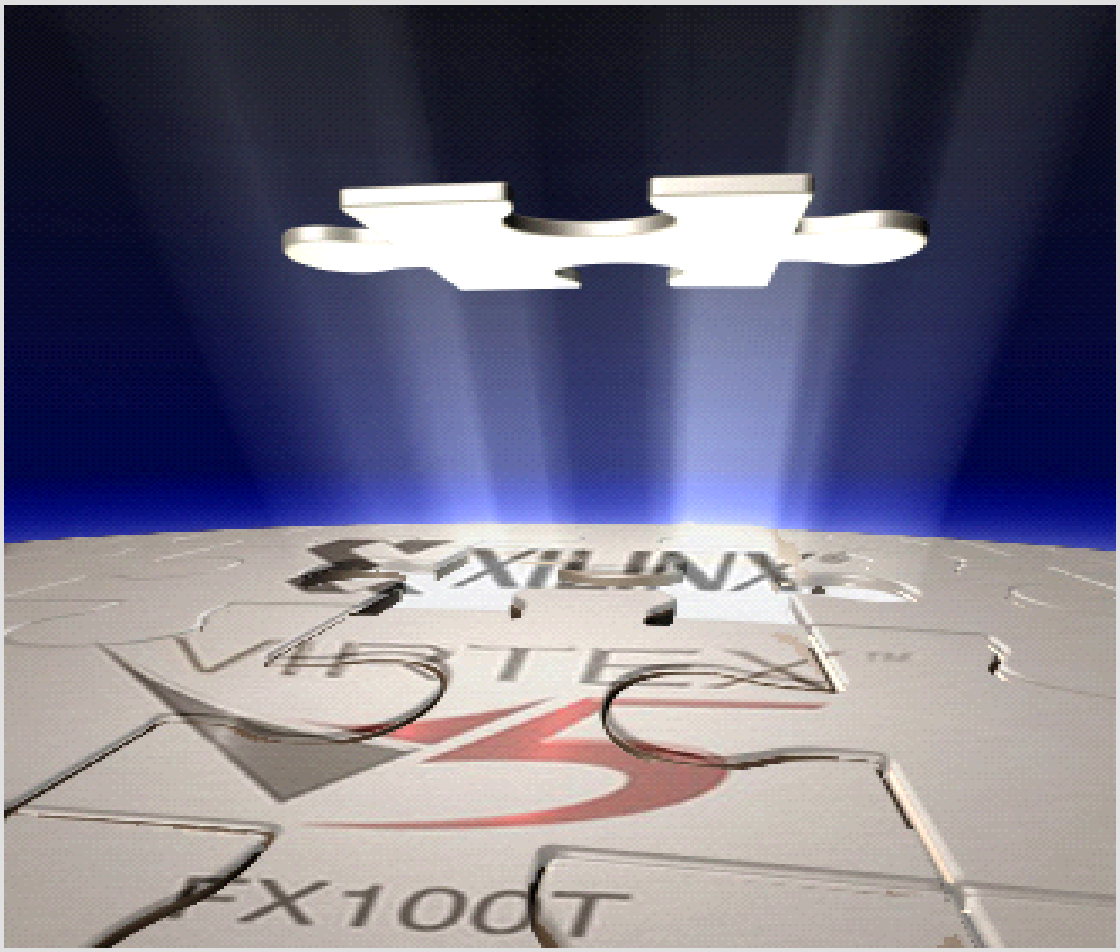
Are FPGAs ready for accelerating datacenters?



Dirk Koch, The University of Manchester, UK (dirk.koch@manchester.ac.uk)

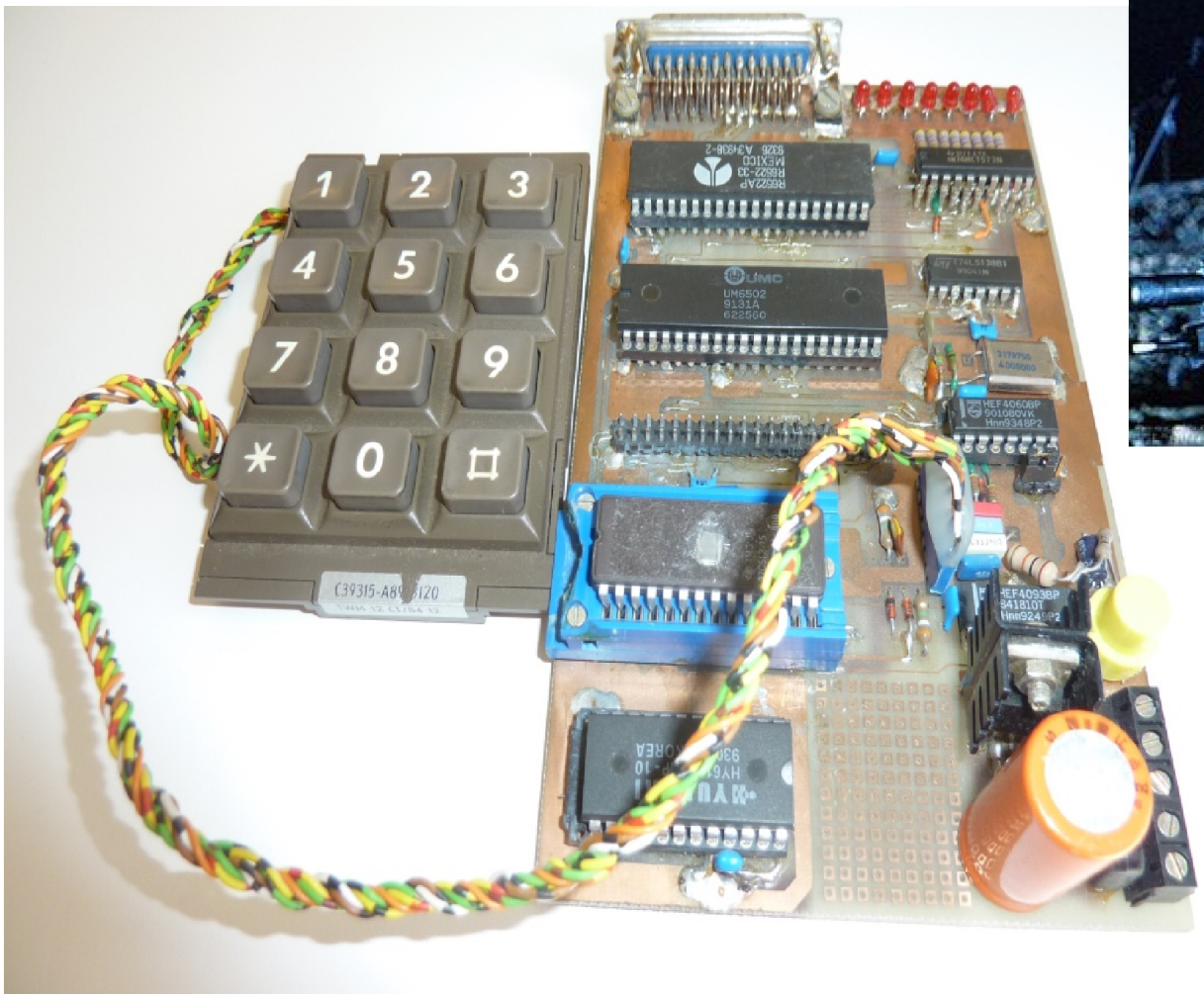
Let's build a 1,000,000,000,000,000 FLOPS Computer

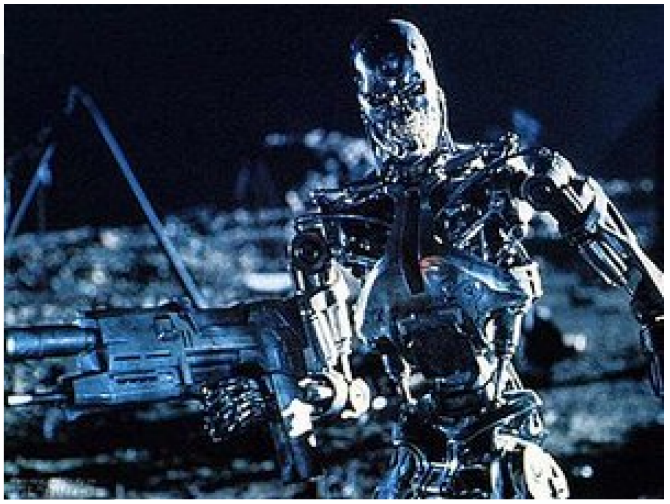
(Exascale computing:
10^{18} FLOPS = one quintillion or a billion billion
floating-point calculations per sec.)

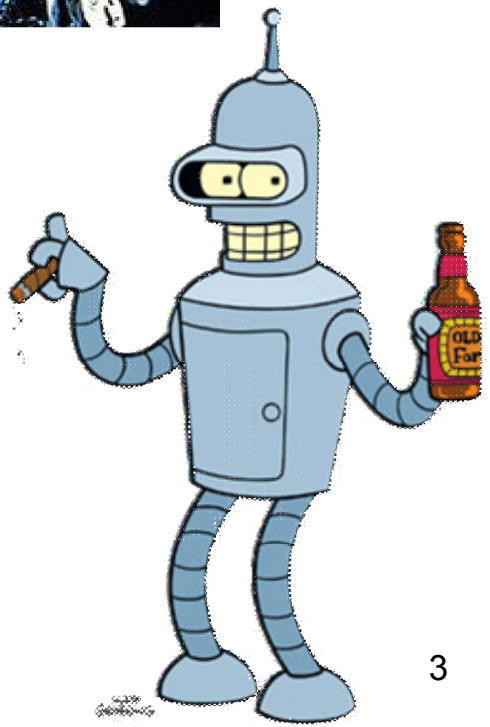
1,000,000,000,000,000,000 FLOPS

- 10,000,000,000,000,000,00 FLOPS

  1975: MOS 6502 (Commodore 64, BBC Micro)



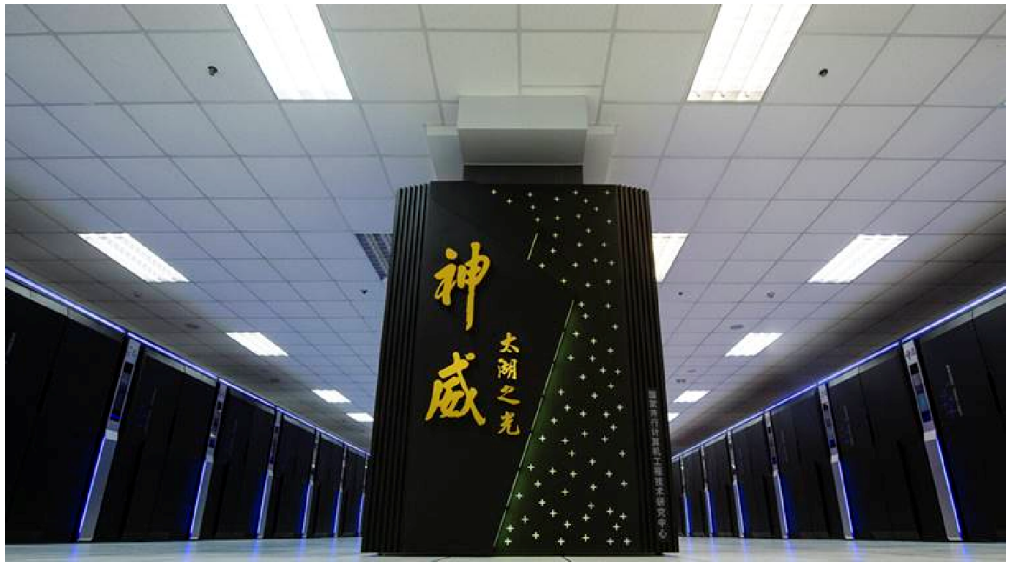




Sunway TaihuLight Supercomputer

- 2016 (fully operational)
- 12,543,6000,000,000,000,00 FLOPS (125.436 petaFLOPS)
- Architecture Sunway SW26010 260C (Digital Alpha clone) 1.45GHz 10,649,600 cores
- Power "The cooling system for TaihuLight uses a closed-coupled chilled water outfit suited for 28 MW with a custom liquid cooling unit"*

  *https://www.nextplatform.com/2016/06/20/look-inside-chinas-chart-topping-new-supercomputer/
- Cost US$ ~$270 million



TaihuLight for Exascale Computing?

We need 8x the worlds fastest supercomputer:

- Architecture — Sunway SW26010 260C (Digital Alpha clone) @1.45GHz: > 85M cores
- Power — 224 MW (including cooling)
  costs ~ US$ 40K/hour, US$ 340M/year
  from coal: 2,302,195 tons of CO2 per year
- Cost — US$ 2.16 billion

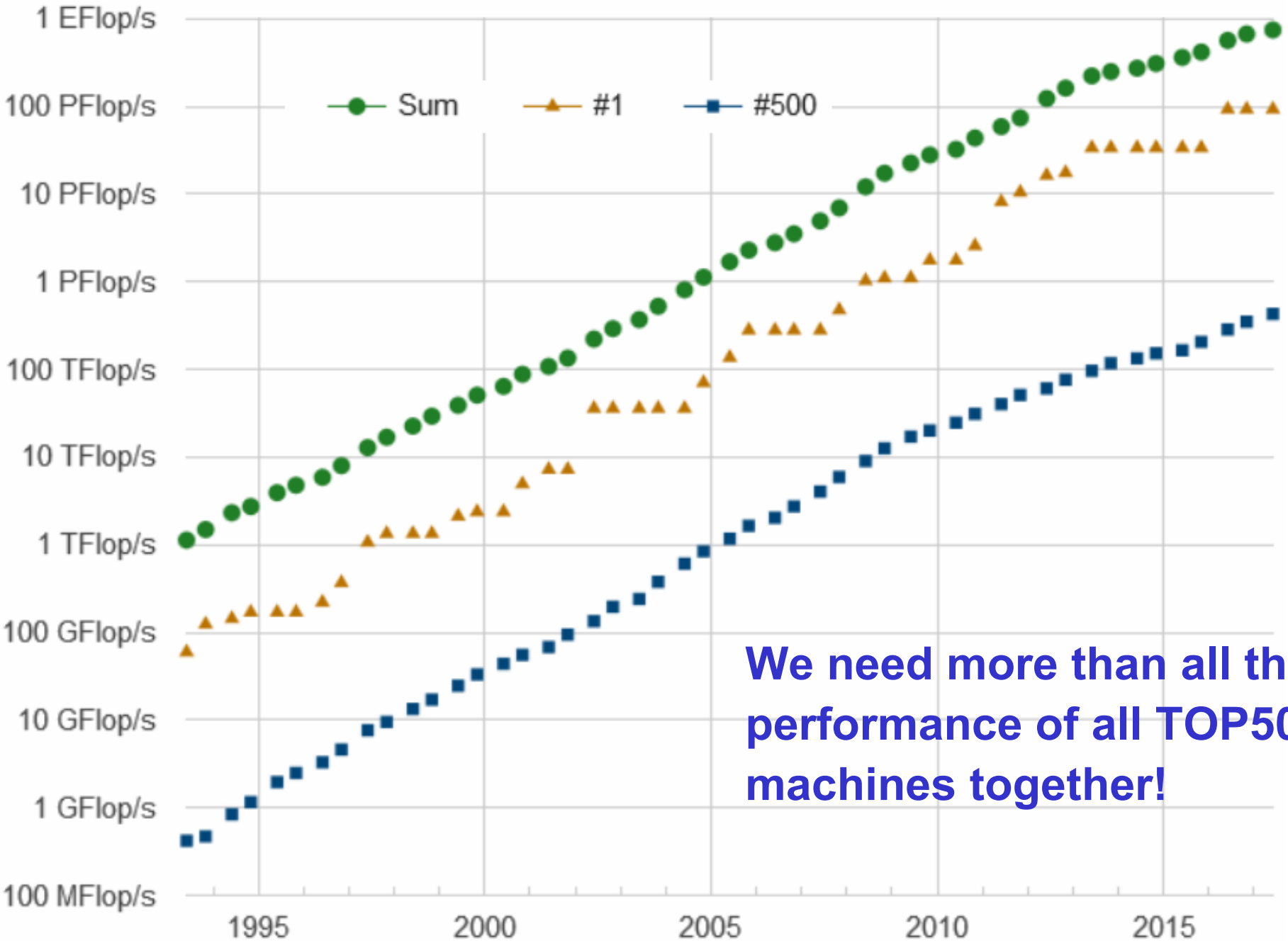
We have to get at least

10x better in energy efficiency

2-5x better in cost

Also: scalable programming models

TOP500 Performance Development



Alternative: Green500

Shoubu supercomputer (#1 Green500 in 2015):

- **Cores:** 1,181,952
- **Theoretical Peak:** 1,535.83 TFLOPS/s
- **Memory:** 82 TB
- **Processor:** Xeon E5-2618Lv3 8C 2.3GHz
- PEZY-SC accelerators (GPU-like; uses OpenCL) (theoretical 6-7 GFLOPS/W)

- → **150 MW for Exascale** (very optimistic)

- Good, but not good enough

GPUs?

- Energy efficiency is the key for performance computing
  → high integration → high memory / I/O throughput
  → There is strong need to process more data faster at less power!
- Energy efficiency is the most important technology driver (from mobile to datacenters)
- Many people consider GPUs for energy efficient computing, **but**

M2050 GPU

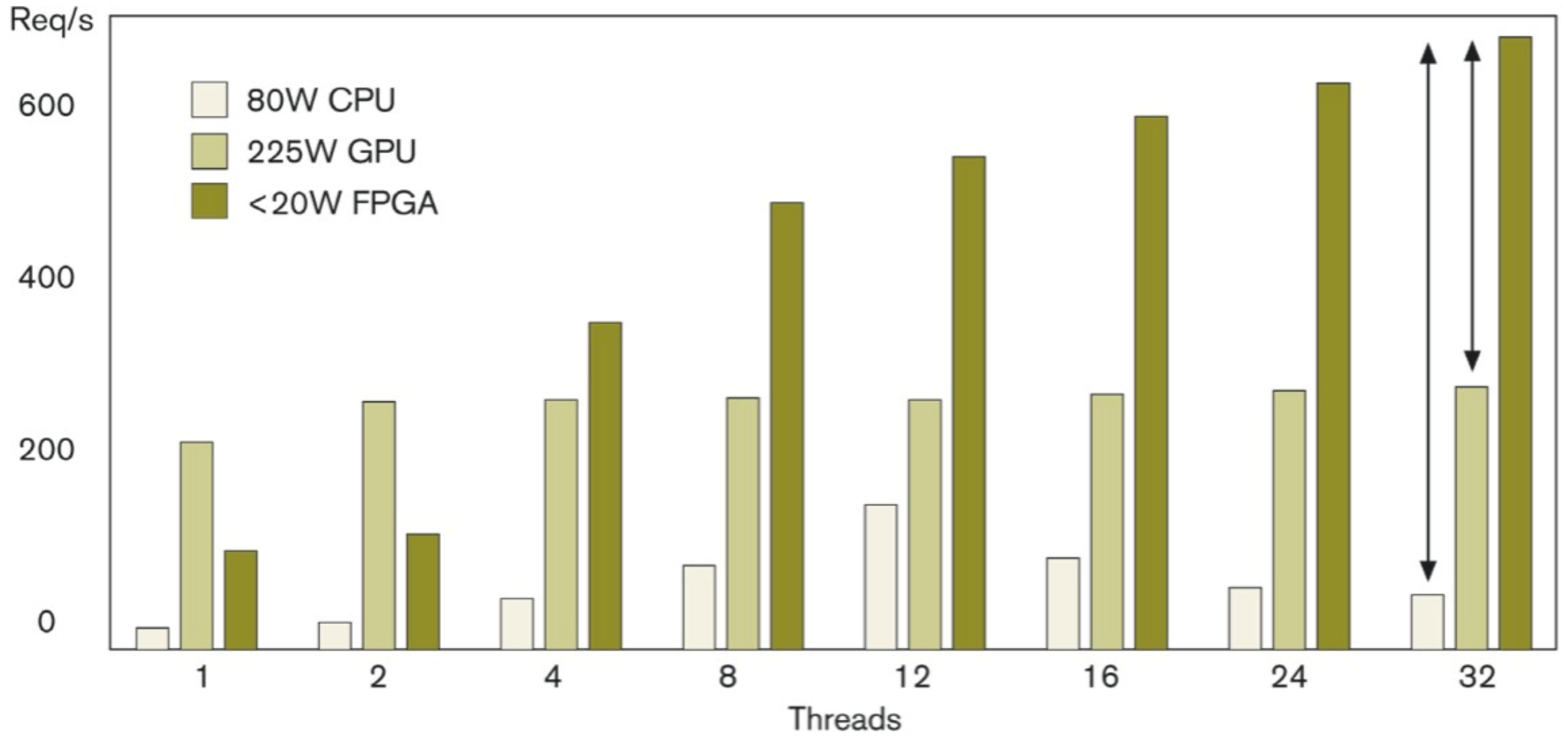
TPD: 225 W (5.4 kWh/day)

Equipment cost: 2400 US$
(energy cost exceeds in less than 5 years considering 0.25 US$/kWh)

1 TFLOPs peak
→ **225 MW at Exascale** (very optimistic)

Alternative: FPGAs

Baidu's analysis on predictive search algorithms:



- FPGA ~500 GFLOPS @ 20W
- **40 MW at Exascale** (optimistic)
  → close to the **green** zone

- 10 x lower power
- 2 x more performance

→ 20 x more compute/unit

CPUs vs. GPUs vs. FPGAs

- **CPUs are ideal for control flow dominated tasks (e.g., an OS, compilation)**
- **GPUs are ideal for double precision number crunching**
- **FPGAs are ideal for number crunching problems that fit dataflow processing model**
- **FPGA advantages**
  - **Customized processing**
  - **Optimized data movement**
  - **High integration (entire system including memory, mass storage, networking, acceleration and CPU)**
  - **High performance at low power**

We are at the end of CMOS scaling

- **Compute demand still exploding**
  (Health, Climate, AI, Big data)
- **No alternatives (optical, quantum computing) at the horizon!**
  → We have to make the best out of CMOS!
- **Issue is not the number of transistors but energy efficiency**
  → Energy efficiency translates into compute performance!
- **Objectives are hard to meet with CPUs or GPUs**
  → Custom Computing using FPGAs!
- **Needs rethinking**
  - **Programming models**
  - **Operating systems (runtime environments)**
  - **The "ecosystem"**

Also: Embedded Supercomputing!


Are we having a golden era?

How golden is the FPGA era?

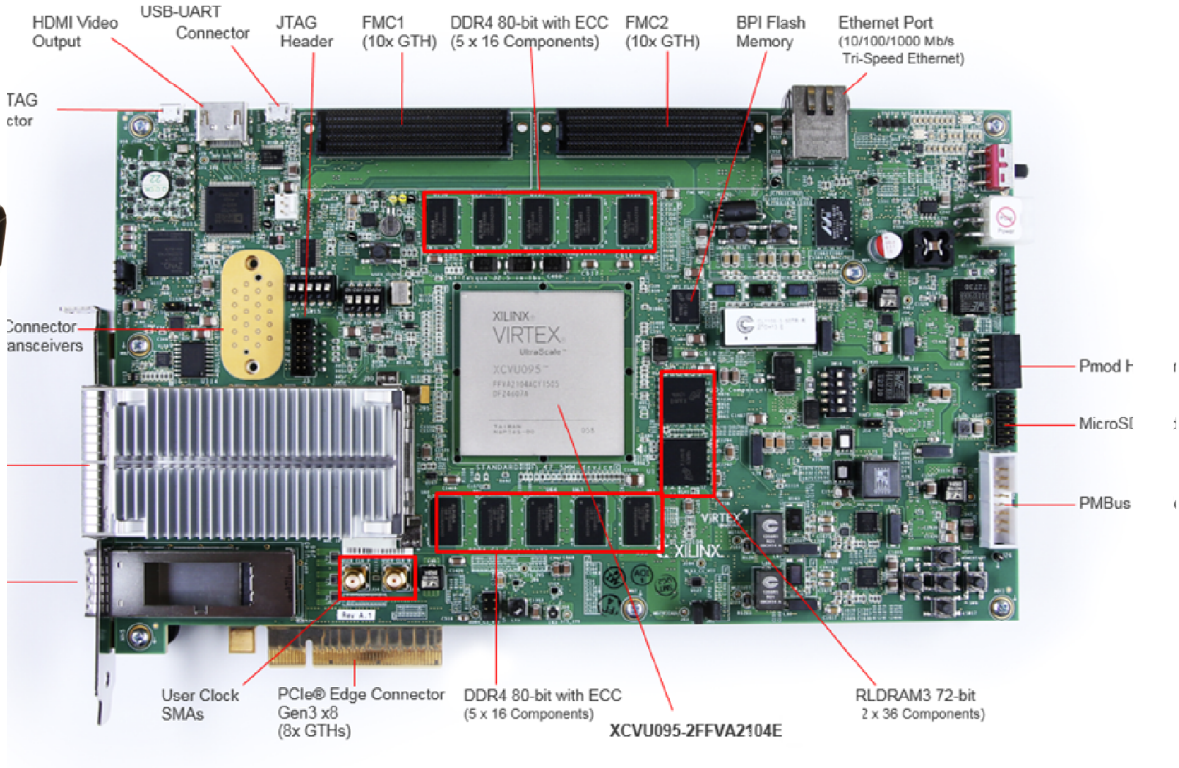
- Technology is driven by applications earning the money
  - Is AI / machine learning enough?
    GPUs are there because gaming paid for it!
- Applications have to get implemented
  - Programming models and tools
  - Devices and platforms
- Hard to get it right!
  Think of Cell Processor!
  - 100 GFLOPS on
    Linpack in 2008!
  - IBM's RoadRunner
  - All good on paper, but
  - Too hard to program

How golden is the FPGA era?

- **FPGA vendors are slow in adopting technology/markets**
- **FPGA vendors don't invest enough (incl. academia)**
- **FPGA vendors are still OEMs (→ fragmented ecosystem)**

TESSLA

VC108 (Xilinx Virtex UltraScale)



How golden is the FPGA era?

- **Applications and demand are there**
  →good community effort on mapping tools
  but are we running behind the trend?

- **Programming models: they are there and**
  HLS gets mature. But truth is:
  big installations use handcrafted RTL

- **Platforms: fragmented (no golden choice)**
  (Alibaba , Amazon, Azure, IBM CAPI, Maxeler, …)
  Some standards (AXI), little intercompatibility

- **Management and APIs**
  FPGA virtualization insufficient,
  no common API (heterogeneity ?)

Sorry, no time to lay back!!!

How golden is the FPGA era?

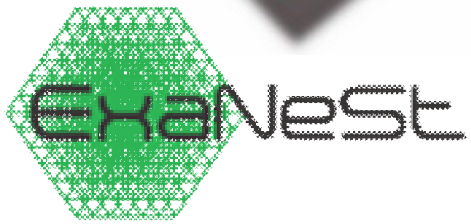
- **We have a once-in-a-lifetime chance to leave our niche**
- **We have to deliver yesterday rather than today**
- **This is a community effort (academia and industry)**

HPC FPGA Research in Manchester

Manchester participates in three joined H2020 projects:

- **ECOSCALE** `http://www.ecoscale.eu/`
  (Scalable programming environment and hardware architecture tailored to current and future HPC applications)

- **ExaNeSt** `http://www.exanest.eu`
  HPC interconnection networks, storage and cooling

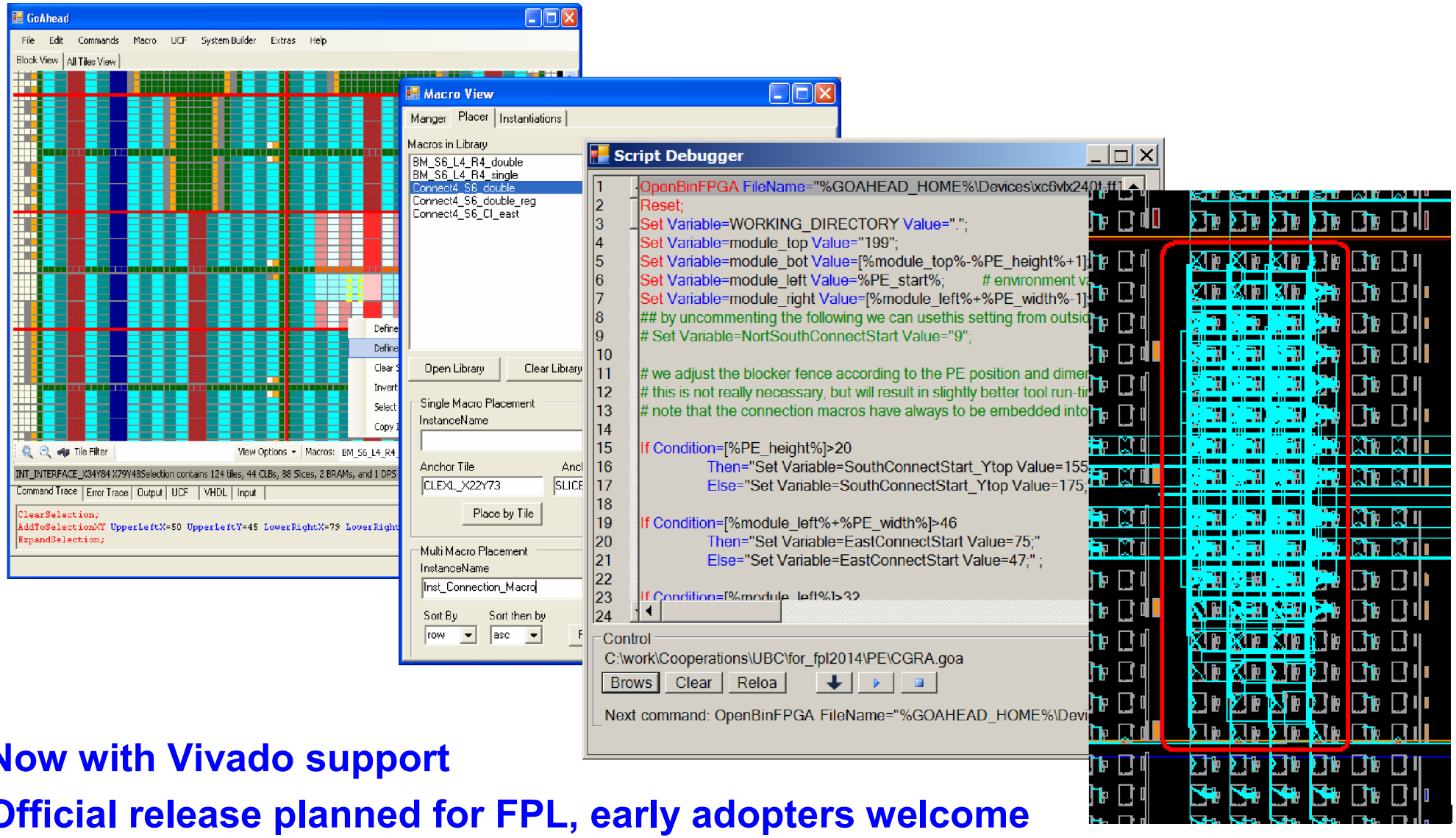


- **ExaNode** `http://www.exanode.eu`
  highly integrated, high-performance, heterogeneous System-on-a-Chip (SoC) aimed towards exascale computing

- **Budget: over 12M, ~2M for Manchester**
- **Follow-up project EuroEXA (total: 20M, over 6M for Manchester)**

Tool – GoAhead

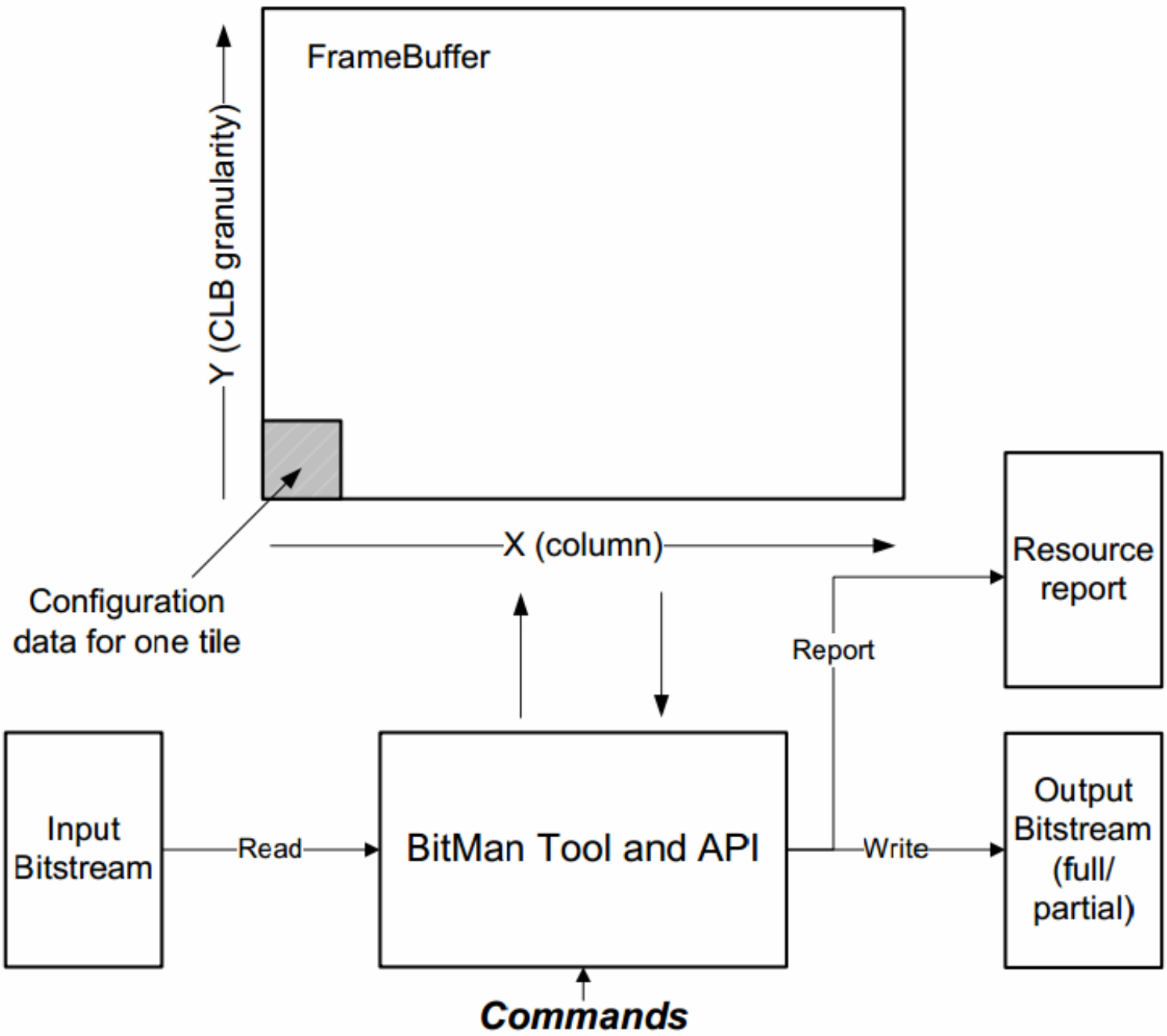
FPGA design tools for implementing partial reconfiguration (most advanced tools currently available)



Now with Vivado support

Official release planned for FPL, early adopters welcome

Tool – BitMan


FrameBuffer
Y (CLB granularity)
X (column)
Configuration data for one tile
Resource report
Report
Input Bitstream
Read
BitMan Tool and API
Write
Output Bitstream (full/ partial)
Commands

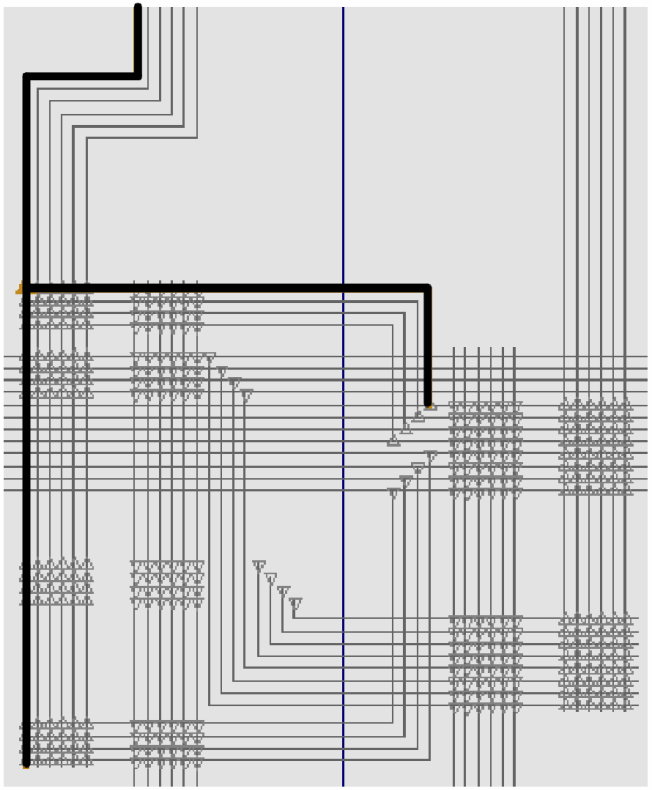
Tool – BitMan

High-level APIs	Functionality

`replace_FPGA_region(X0, Y0, X1, Y1, X2, Y2)`
 	
`duplicate_FPGA_region(X0, Y0, X1, Y1, X2, Y2)`
	Replace/duplicate a rectangular FPGA region bounded by bottom-left (X_0, Y_0) and top-right (X_1, Y_1) to a new region which starts at (X_2, Y_2).
	
	Replace will cl
	data in the old

`reroute_wire(X, Y, input, output)`
 	
`reroute_clock(X, Y, input, output)`
	Change configu
	switch box/cloc
	Y) to connect

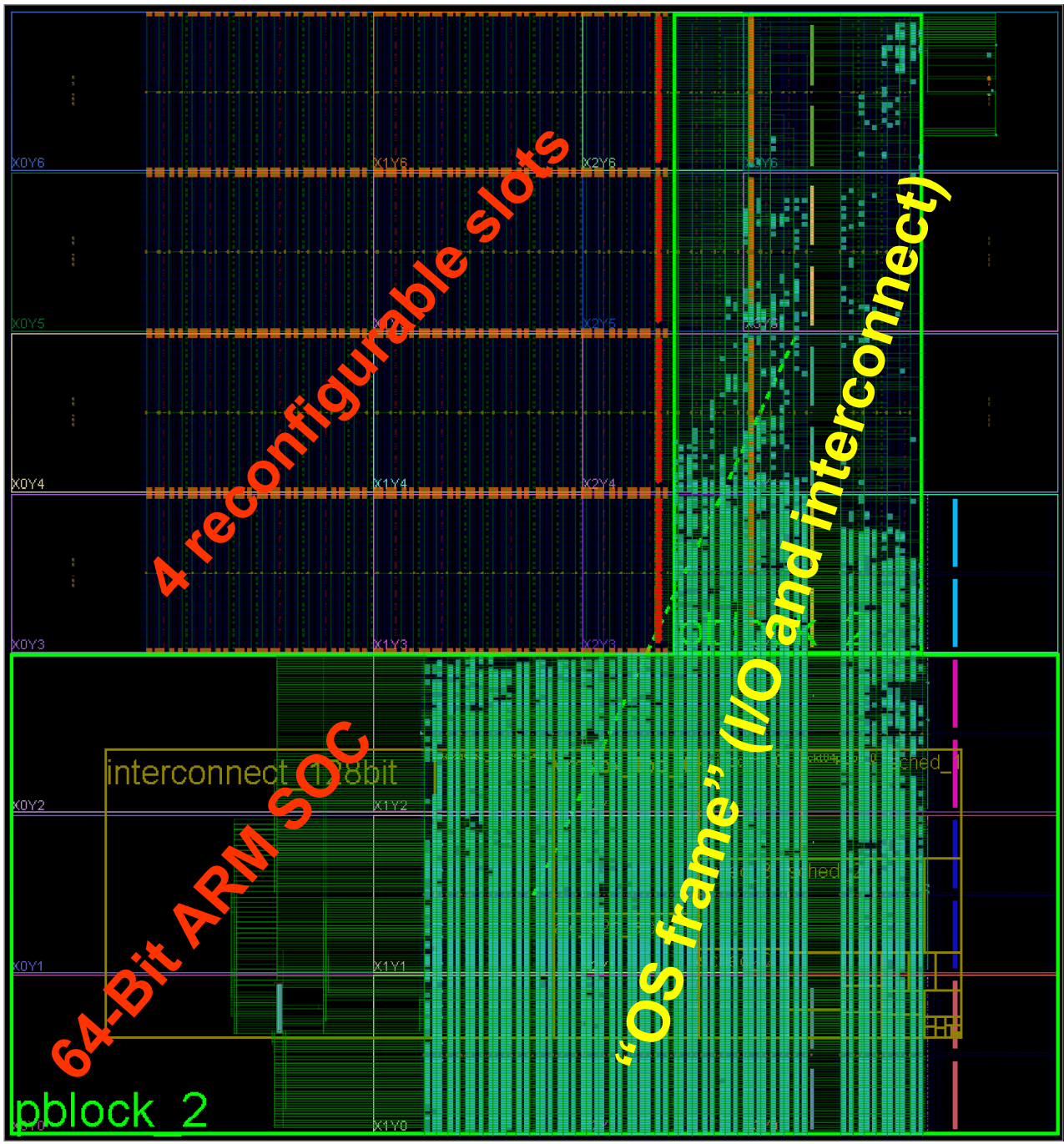
`change_LUT_content(X, Y, LUT, new_config)`
 	
`change_BRAM_content(X, Y, new_config)`
	Change the cor
	(LUT)/BRAM
	new_config



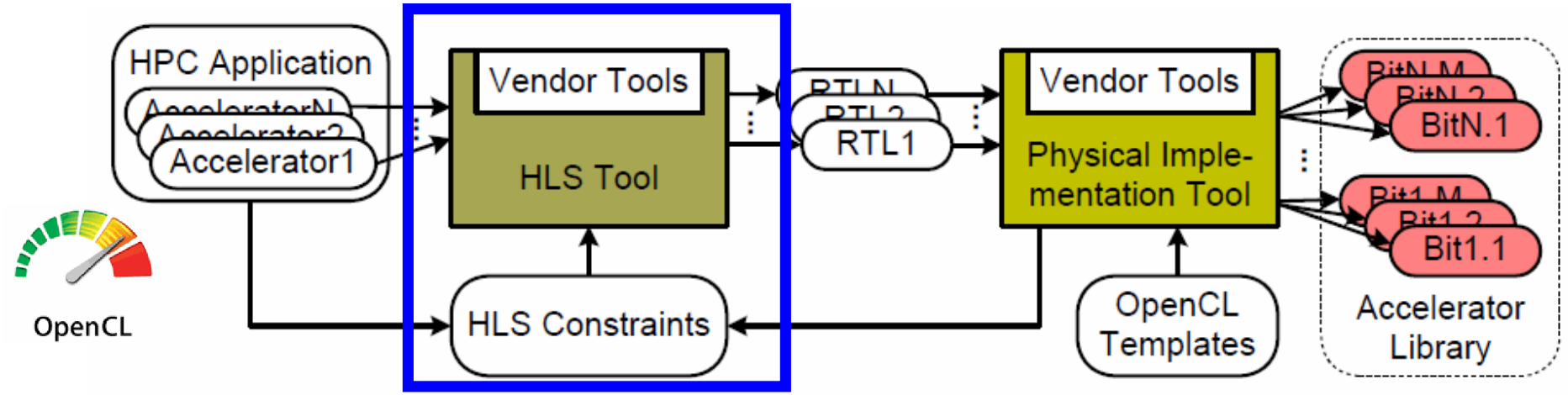
HCLK_L_X32Y78/HCLK_CK_OUTIN_L7
→ HCLK_LEAF_CLK_B_TOPL5
CLB col 01 row 75
Frame 01 : 00 08 09 B3 00 00 00 00

ECOSCALE Prototype

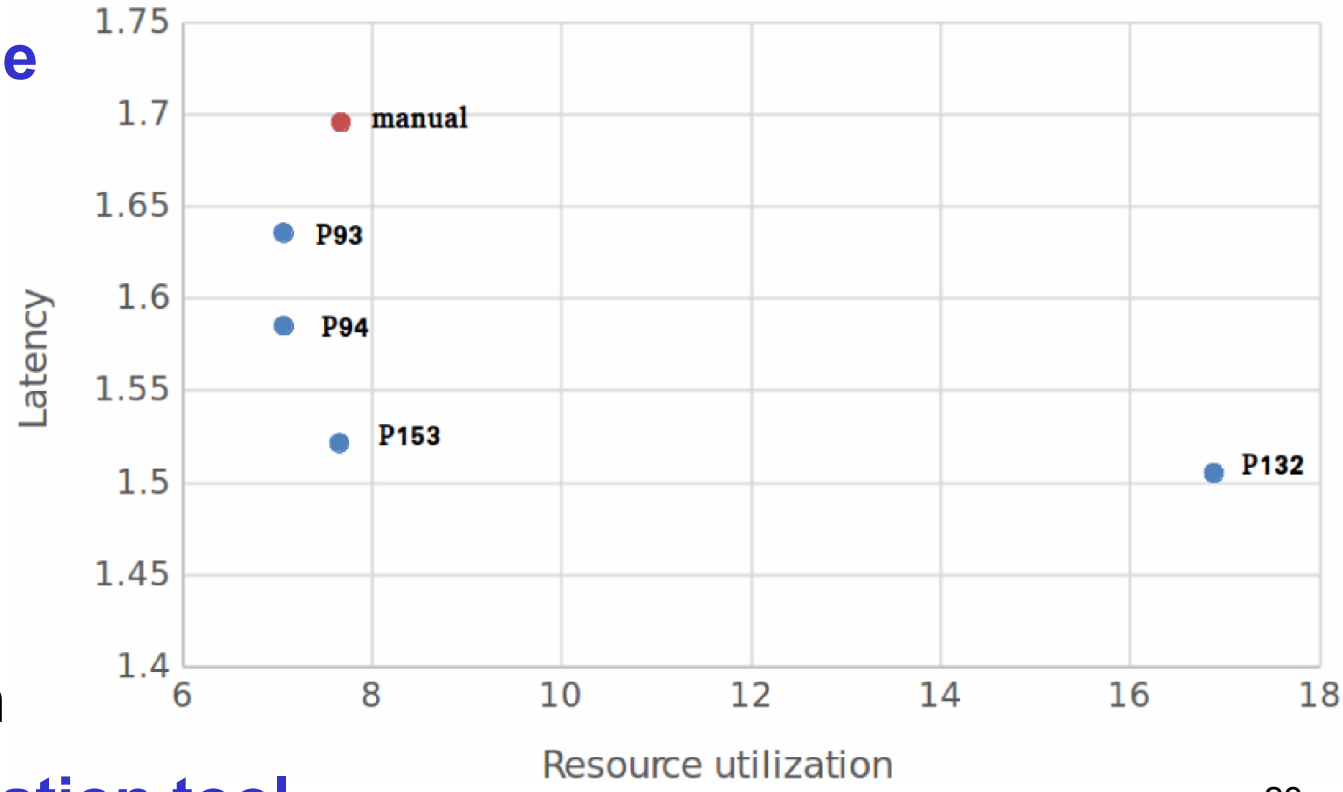
- **Static system with 4 slots (64 bit ADR) (Kintex ZYNQ UltraScale+)**
- **Hosts up to 4 kernels**
- **Modules may occupy multiple consecutive slots**
- **Special features**
  - **Modules implemented independently from the HW OS**
  - **Supports moving compute to data**



Automatic Tool Flow (OpenCL –> Bitstream)



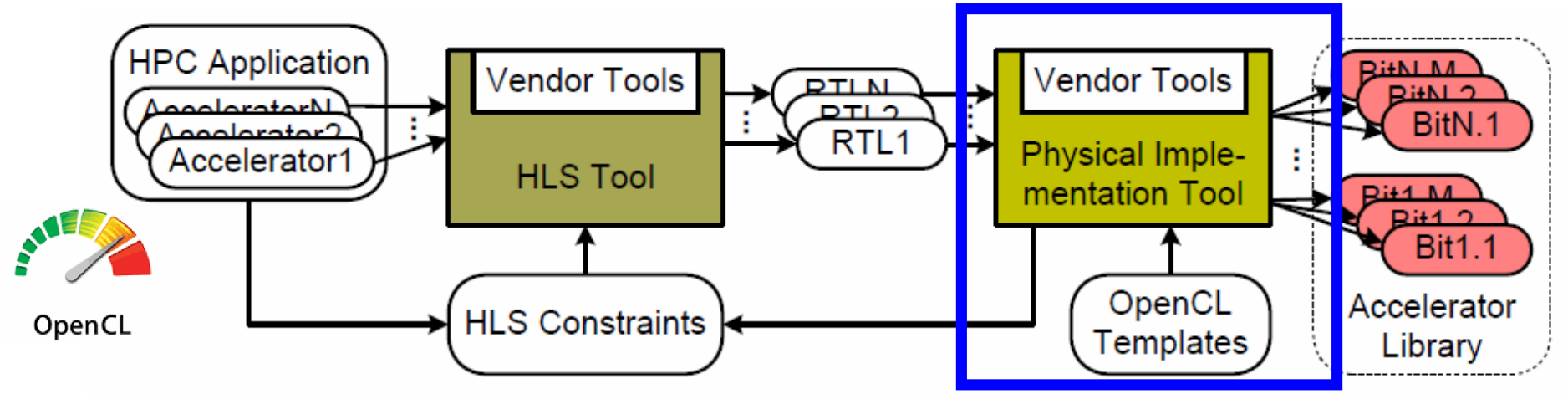
- **OpenCL is not performance portable**
- **Many knobs for trading performce for resources**
- **FPGA implement. for partial reconfig. is hard!**



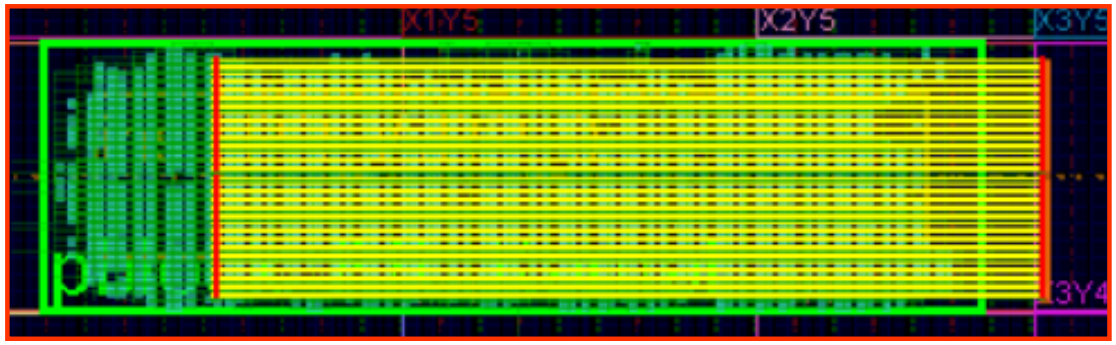
→**DSE integrated with**

→**Physical implementation tool**

Automatic Tool Flow (OpenCL –> Bitstream)



- **DSE output is passed to the physical implementation tool**
- **Decides for how many resources (slots)**
- **Implements relocatable modules to be loaded by runtime system**

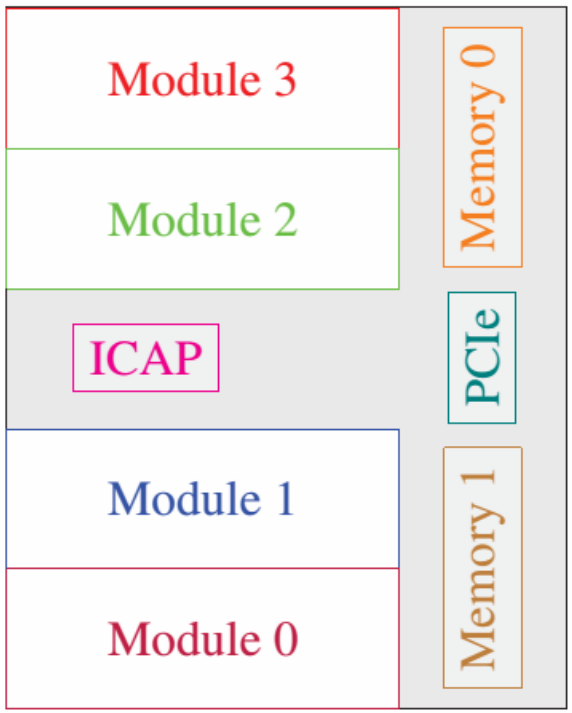


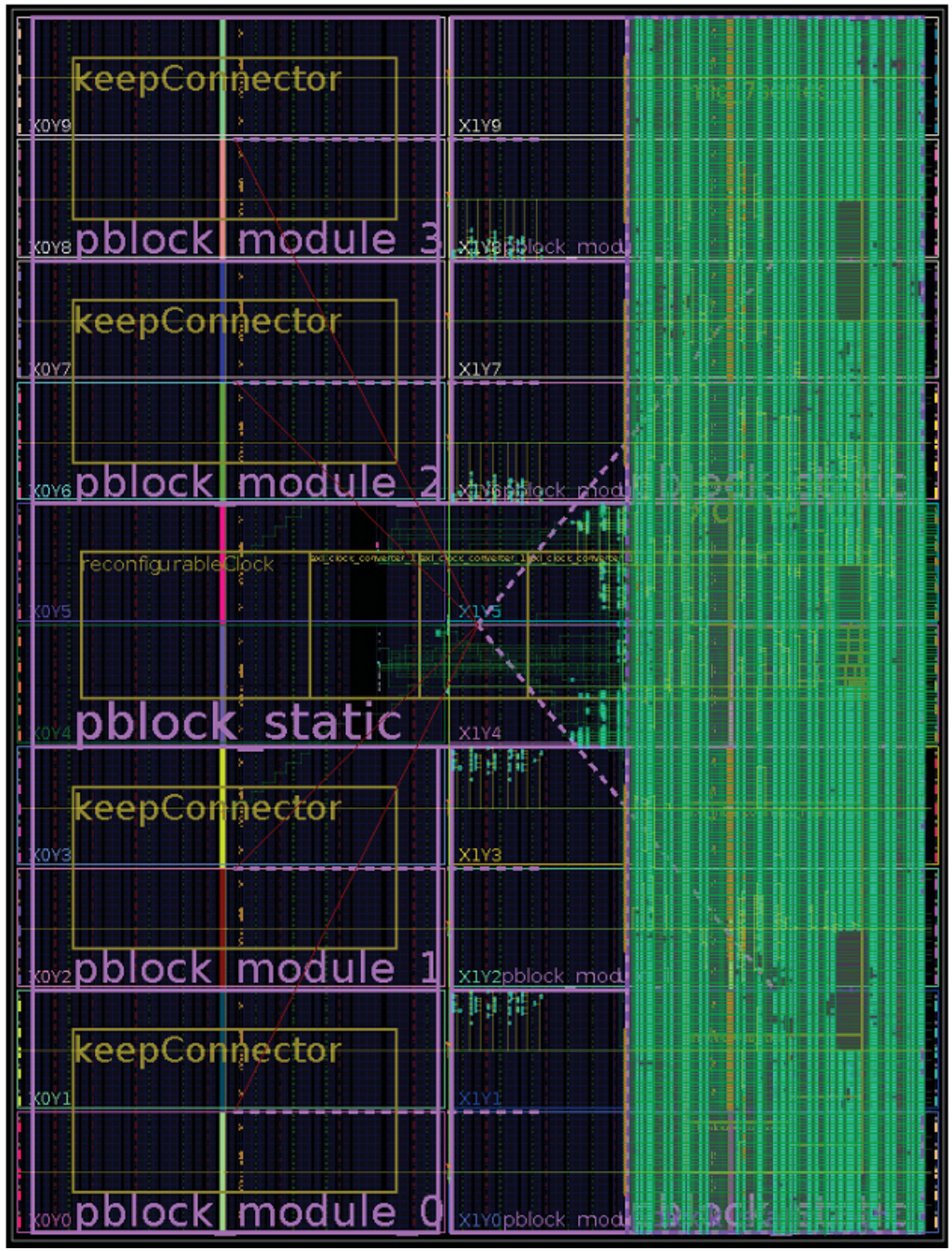
- **Implements mitigation strategies if implementation fails**

Goal: bulletproof compilation flow for non FPGA experts

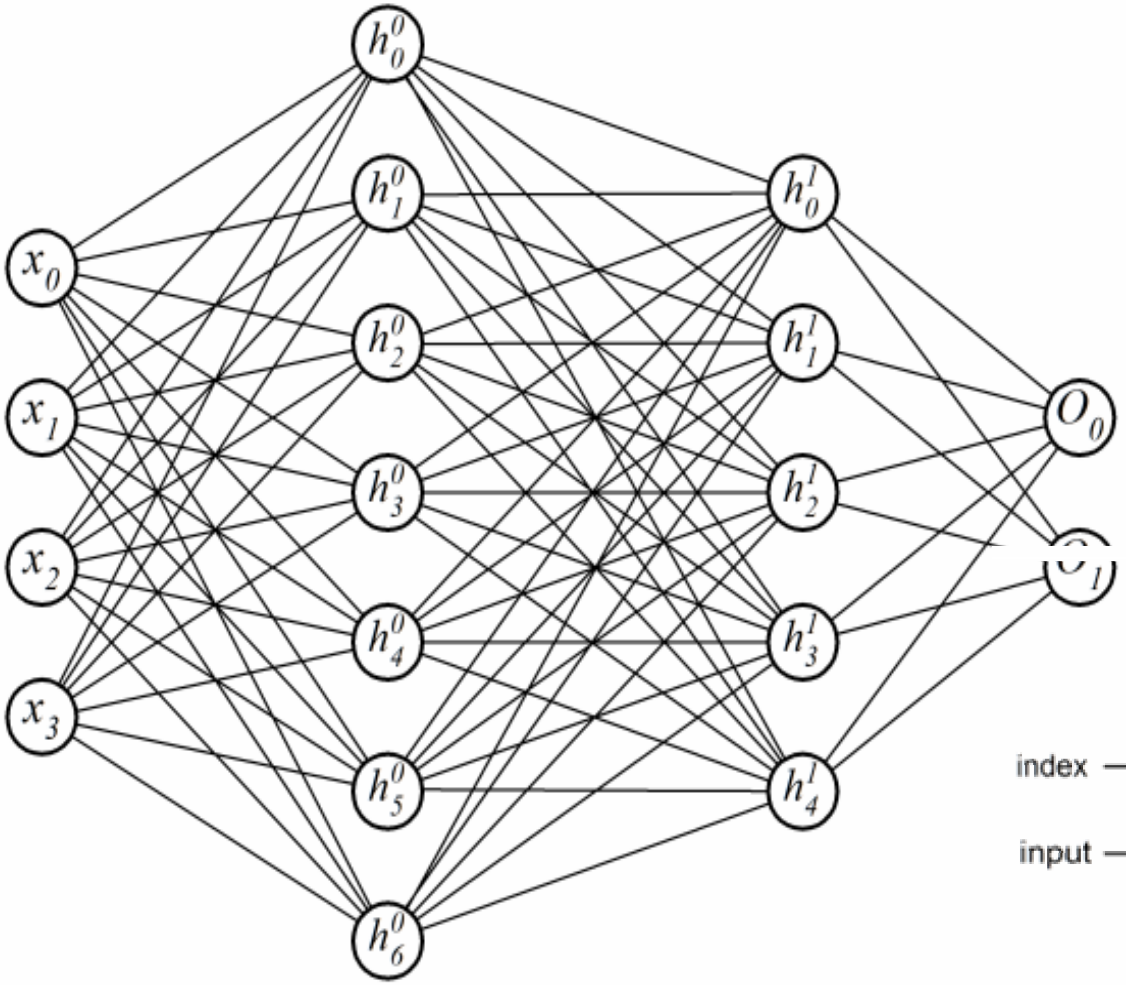
VC709 Infrastructure (Jetstream & PCIeHLS)

- **Infrastructure for integrating OpenCL kernels using PCIe**
- **2 DDR-3 (up to 2 x 8 GB)**
- **Gimmics (e.g., variable clock)**
- **Multi-FPGA support (direct FPGA-2-FPGA PCIe links)**

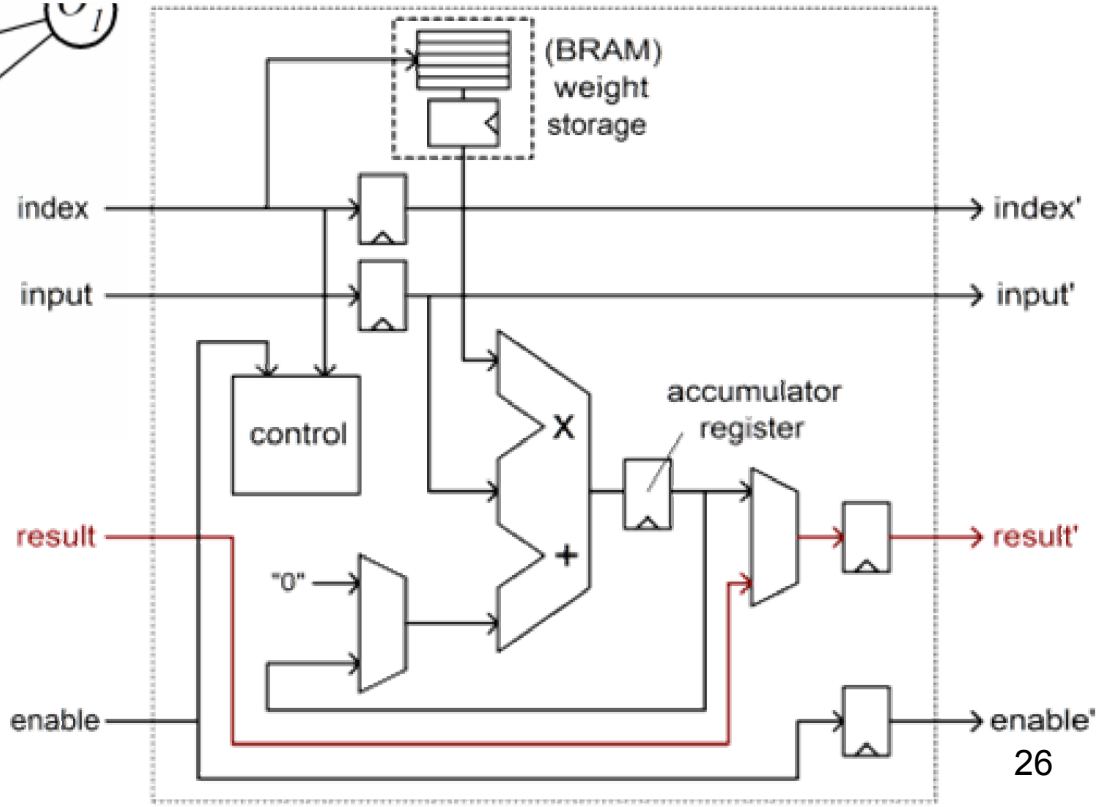




Multi-FPGA Infrastructure for ANNs

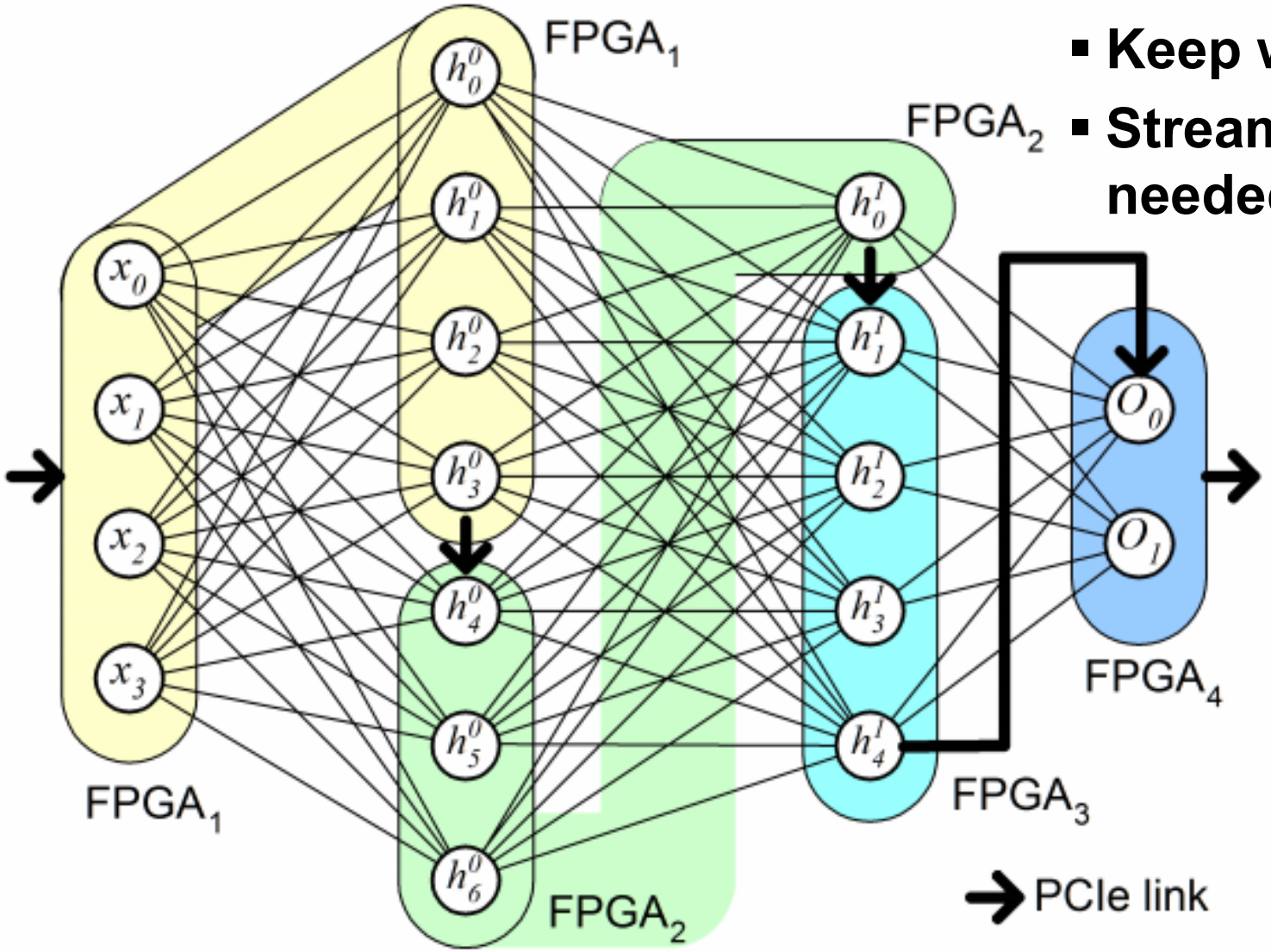


- **Input arrives as a stream**
  → **implement ANN as 1D systolic array**
- **Keep weightset on-chip**
- **Stream to next chip if needed**



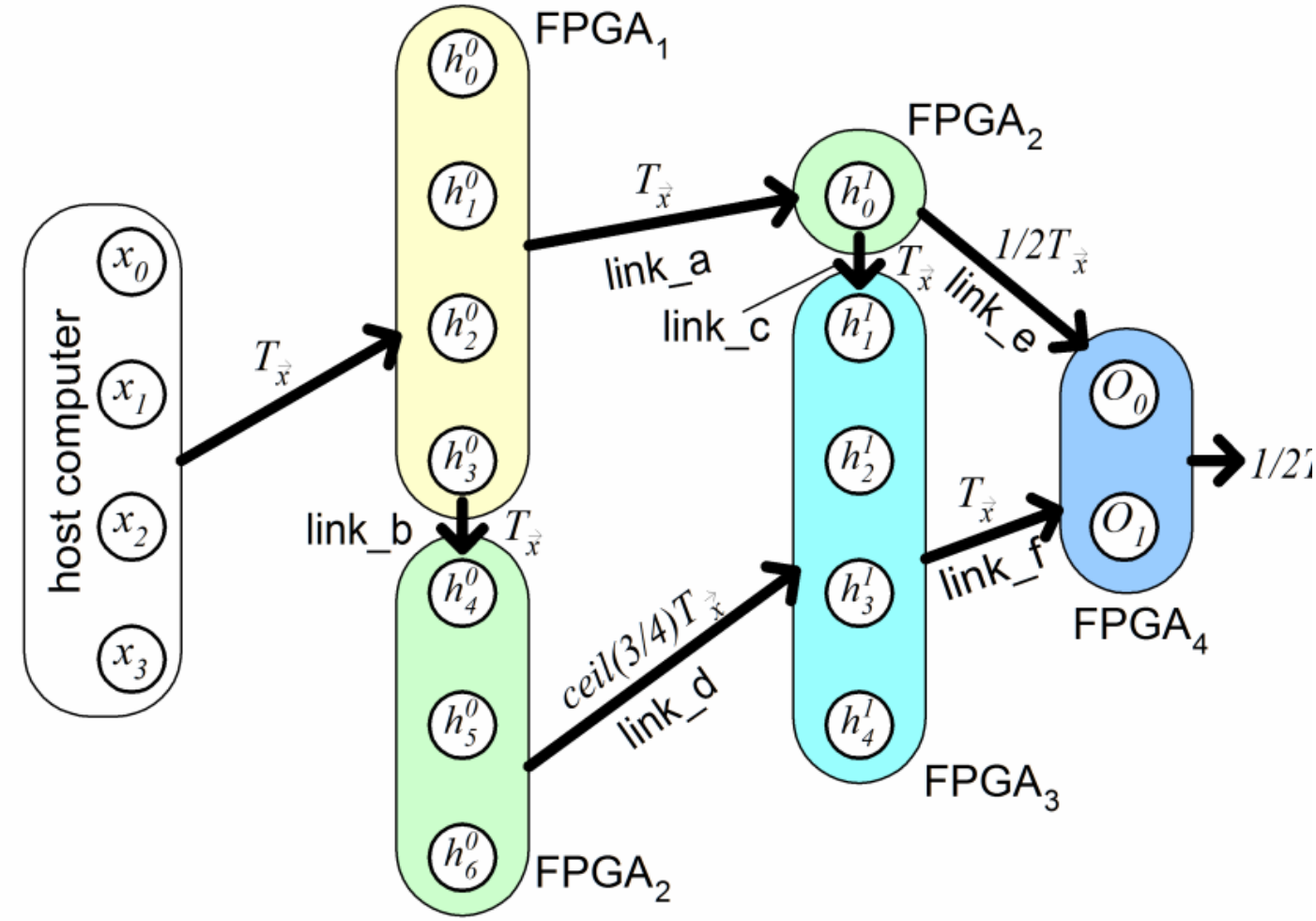
Multi-FPGA Infrastructure for ANNs

- **Input arrives as a stream**
  → **implement ANN as 1D systolic array**
- **Keep weightset on-chip**
- **Stream to next chip if needed**



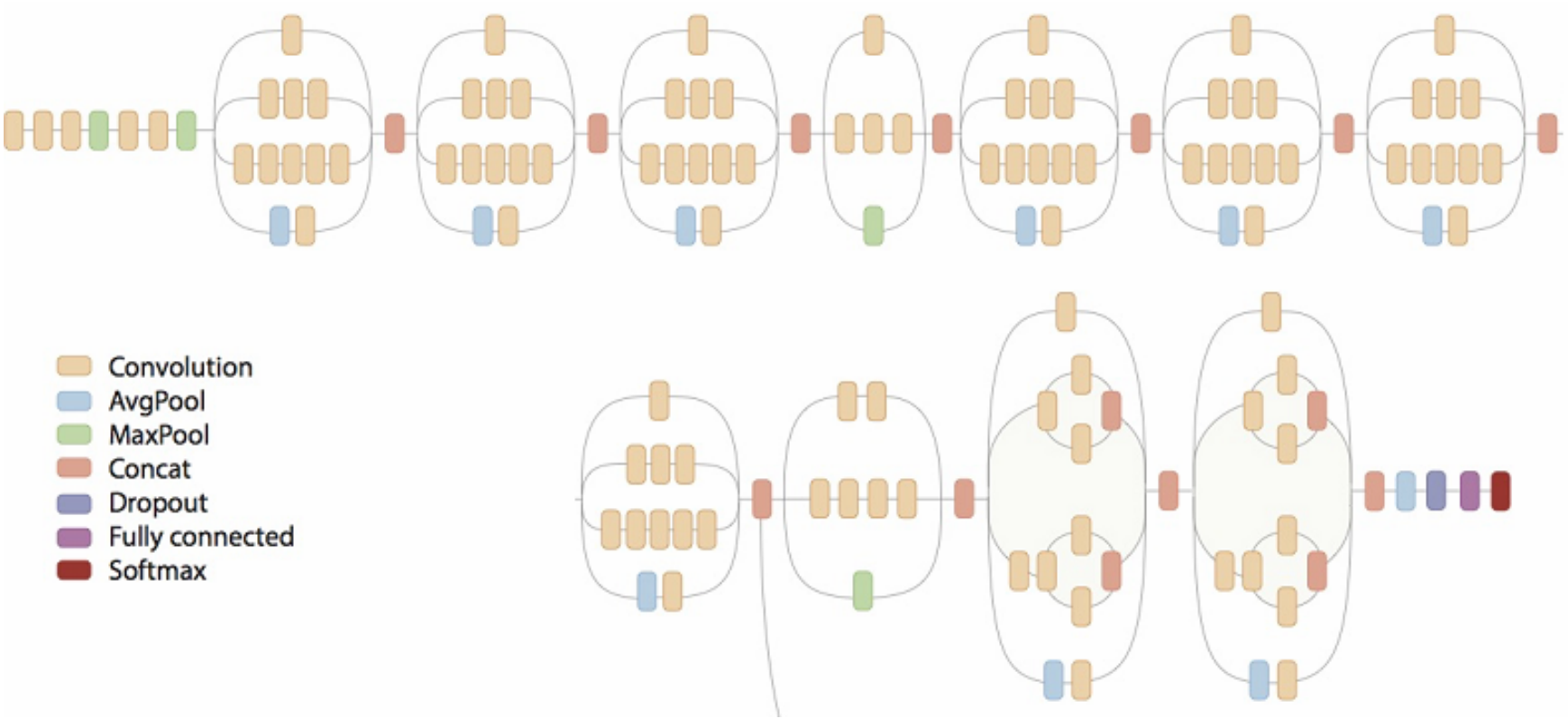
Multi-FPGA Infrastructure for ANNs

- **PCIe implements packet switched network**
- **Allows arbitrary streaming (only bound be the 7GB/s link capacity)**
- **Allows removing bottlenecks**



Multi-FPGA Infrastructure for ANNs

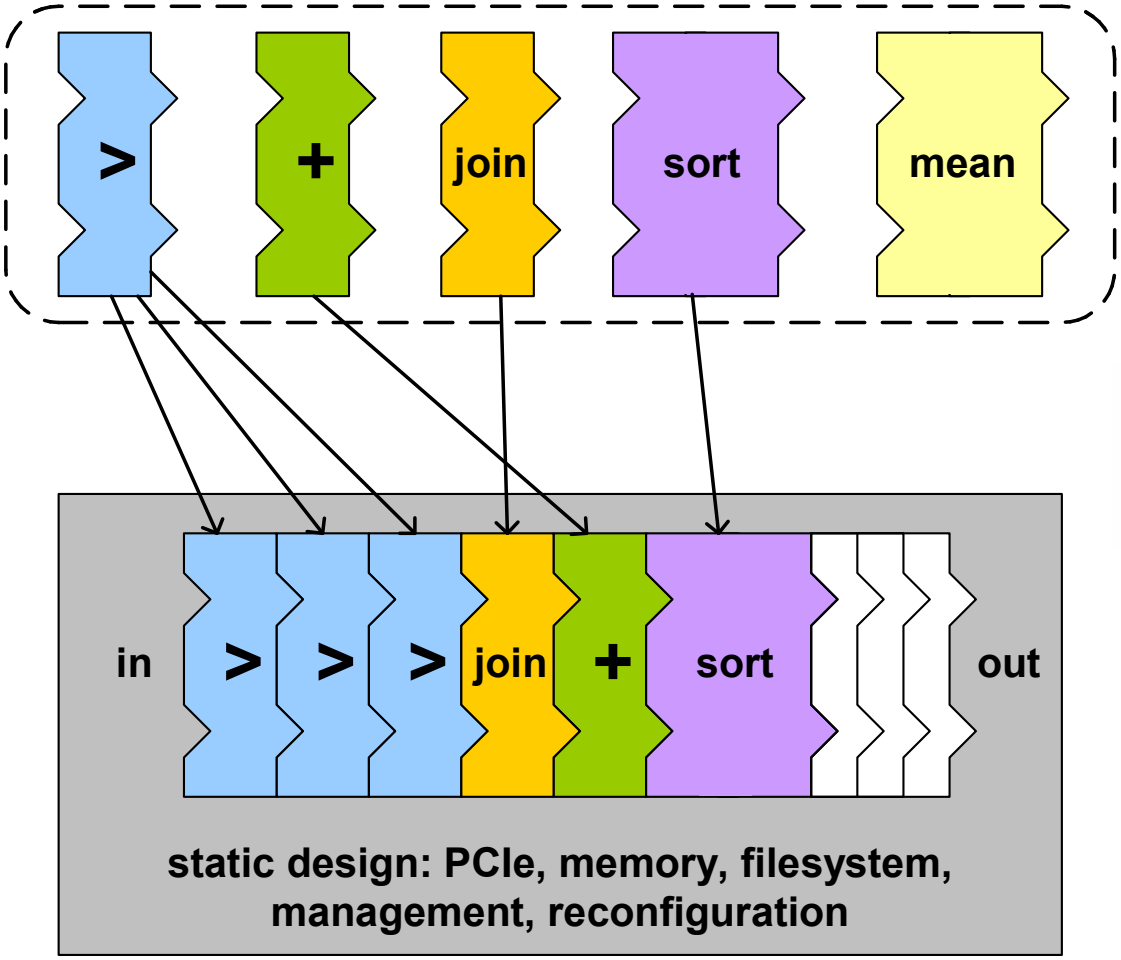
- **Multi-FPGA is a general design pattern to get rid of weight I/O**

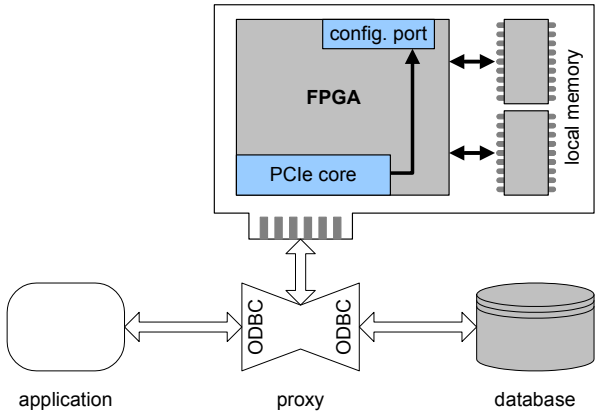


- **Example: TensorFlow Inception-v3**
- **Nothing published, collaboration welcome!**

FPGA Database Accelerator

- Build library with SQL operators
- Compose graph at run-time



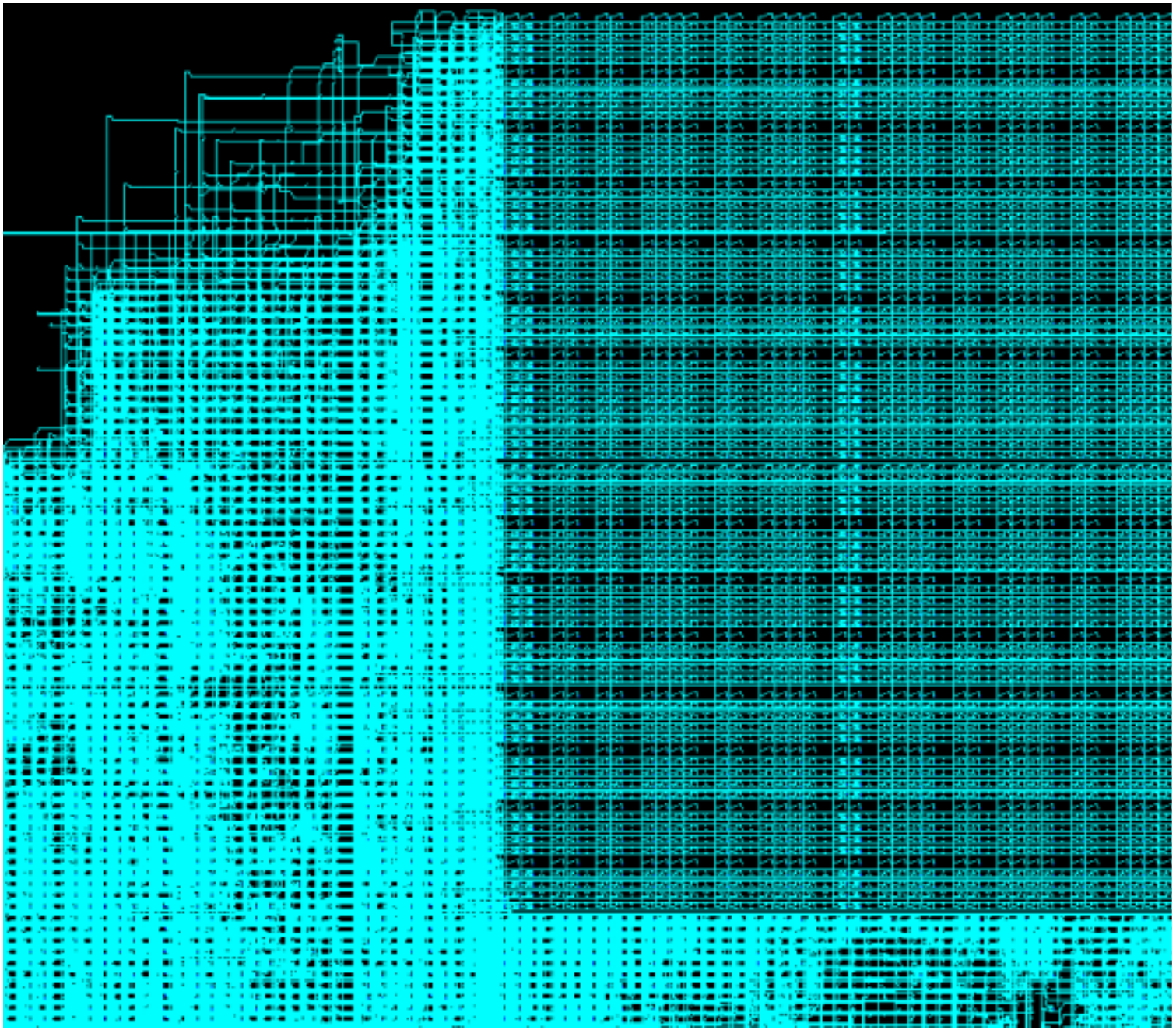


MAXELER Technologies
MAXIMUM PERFORMANCE COMPUTING

implementation:

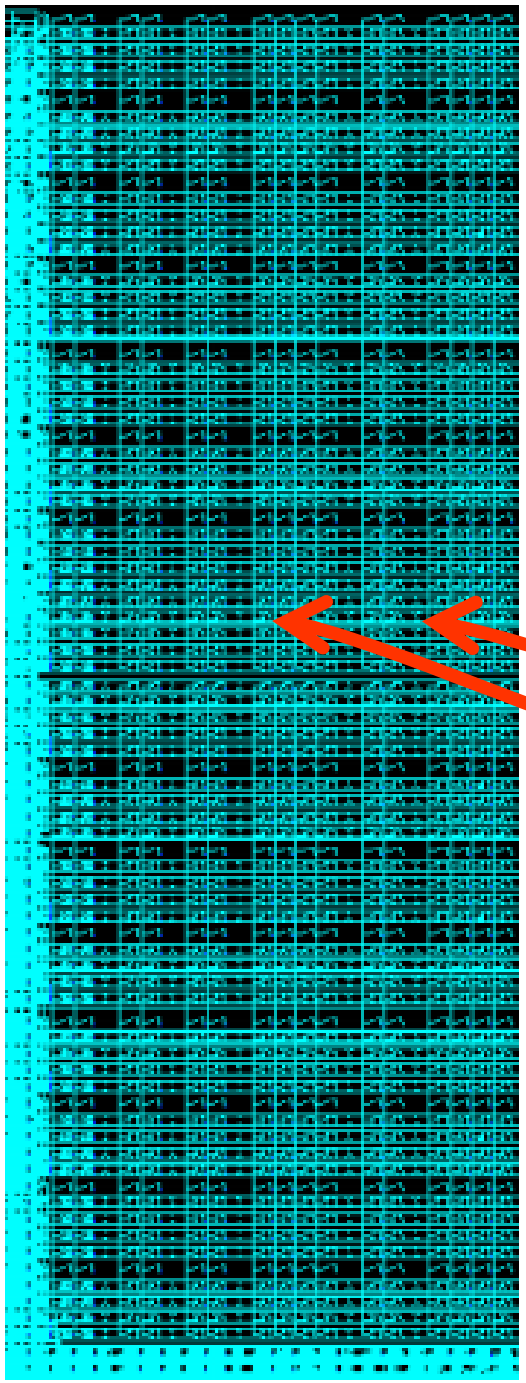
- 512 bit datapath
- 300 MHz (Virtex-6)
- Tables are stored in 24 GB on board RAM (48 GB possible)

FPGA Database Accelerator

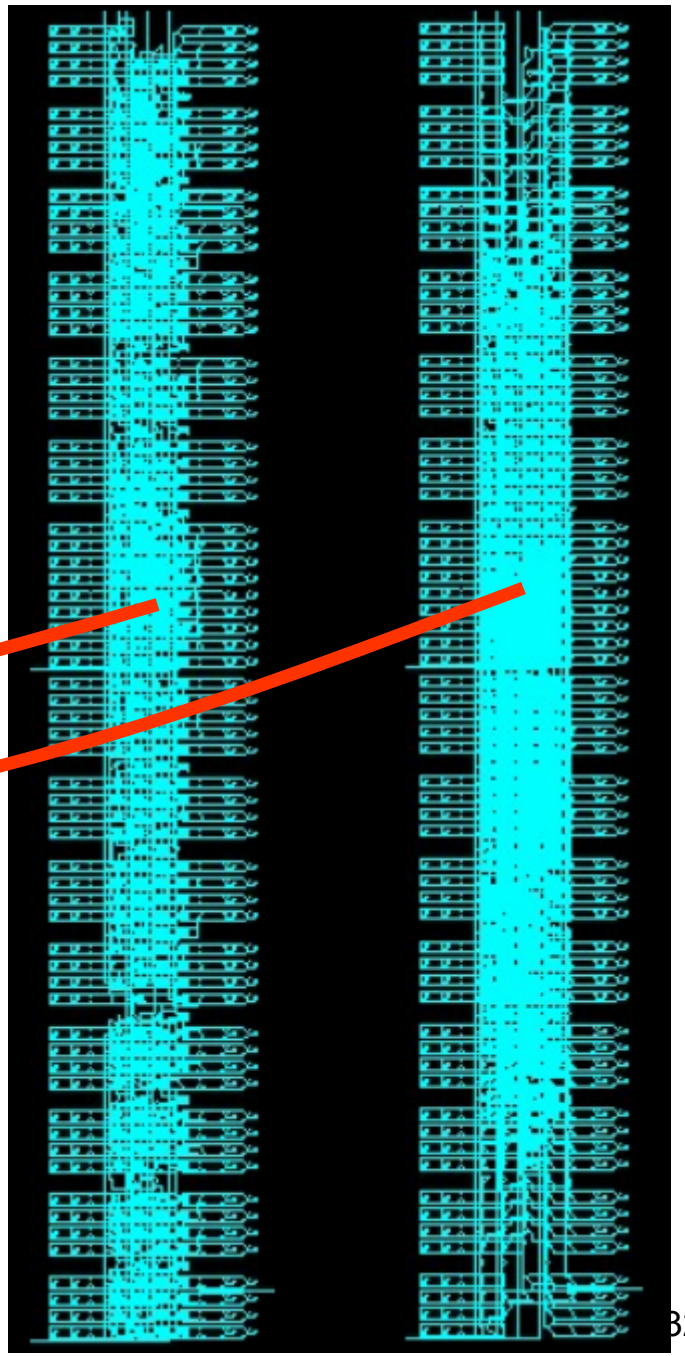


- Reconfigurable region
- Regularly routed
- 16 x 32-bit (512-bit total)
- @ 300 MHz
- ~40% of a MAX3 workstation is reconfigurable

FPGA Database Accelerator



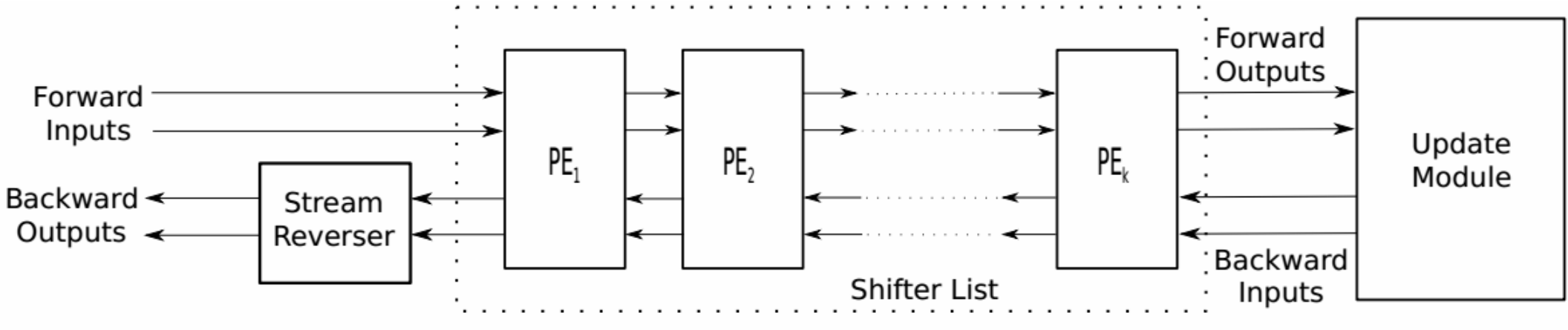
- Build library with SQL operators
- Compose steam processing machine at runtime



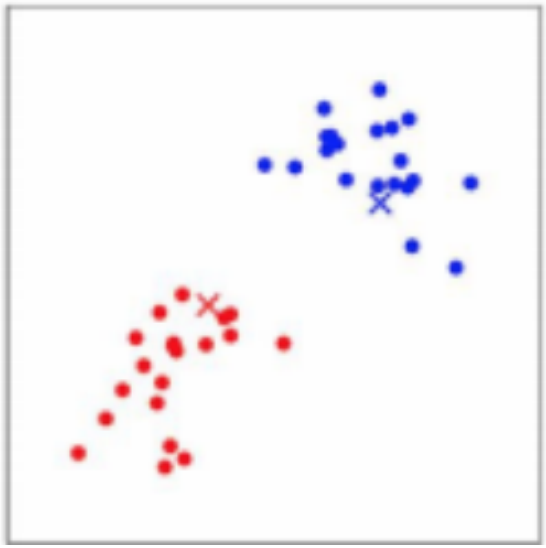
- Integrated in:

PostgreSQL

Reconfigurable Unsupervised Learning

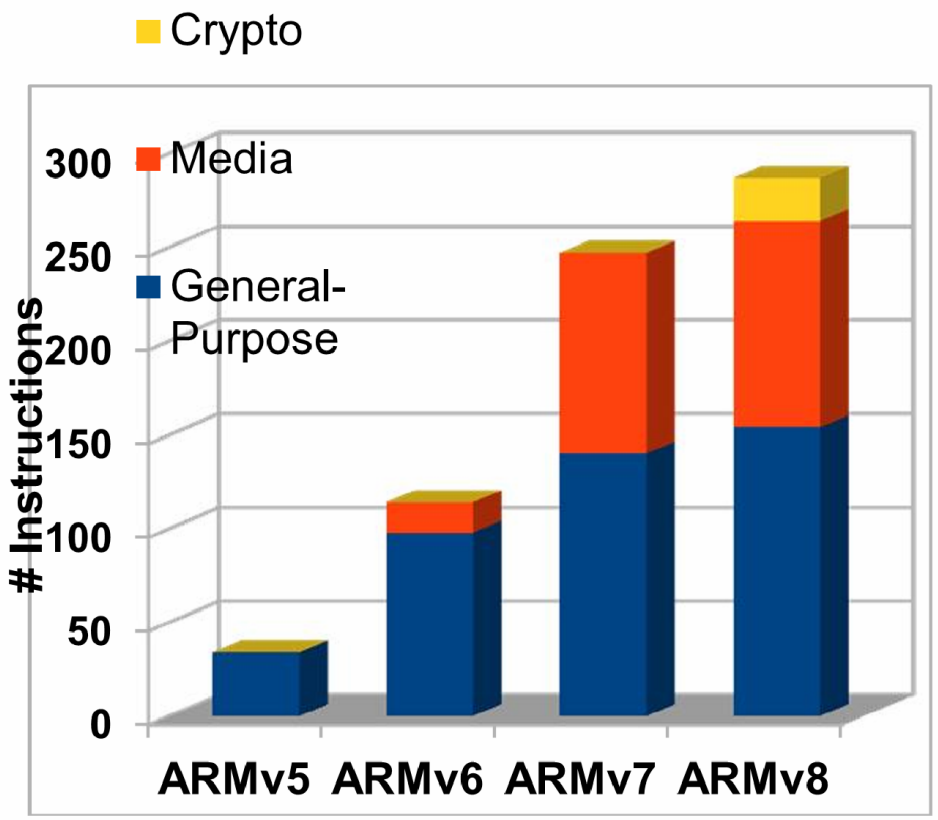


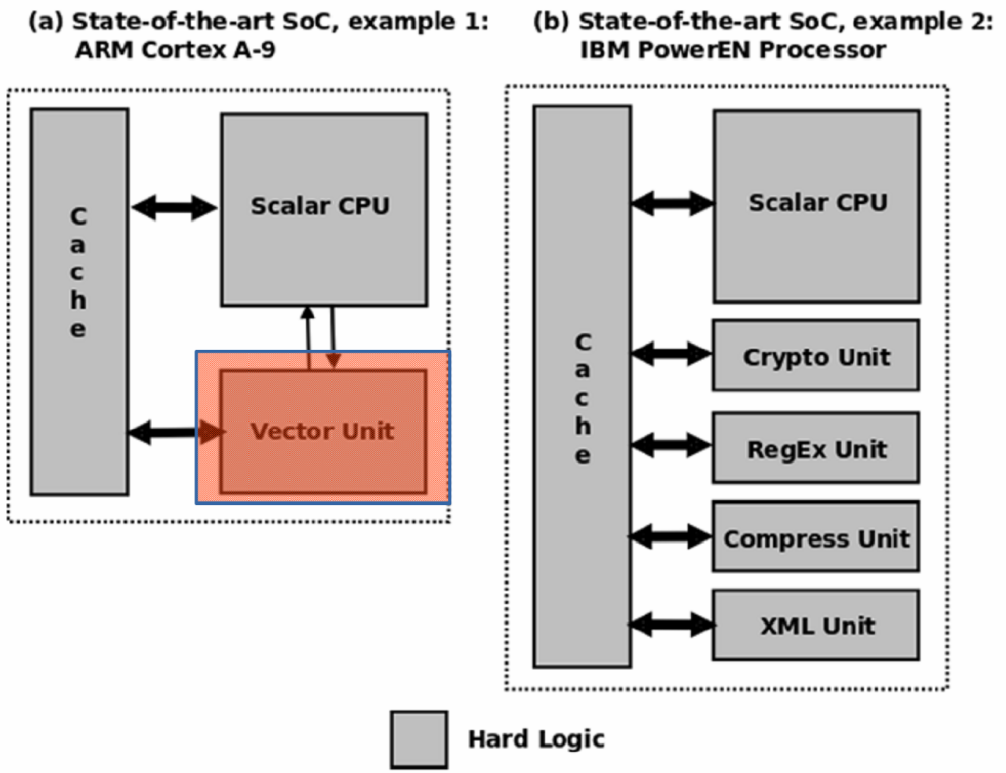
- **Generic architecture for KNN**
- **Built from a few basic building blocks (the update is needed only once and the number of PEs depends on the number of clusters**
- **Multiple runs if resource bound**
- **Can be applied to other problems (Euclidian distances)**



Reconfigurable Instruction Set Extensions

- **Present GP CPU micro architectures leave not much headroom for optimization**
- **CPU clock is limited by power**
- **→ trend to feature-rich instruction sets and acceleration**

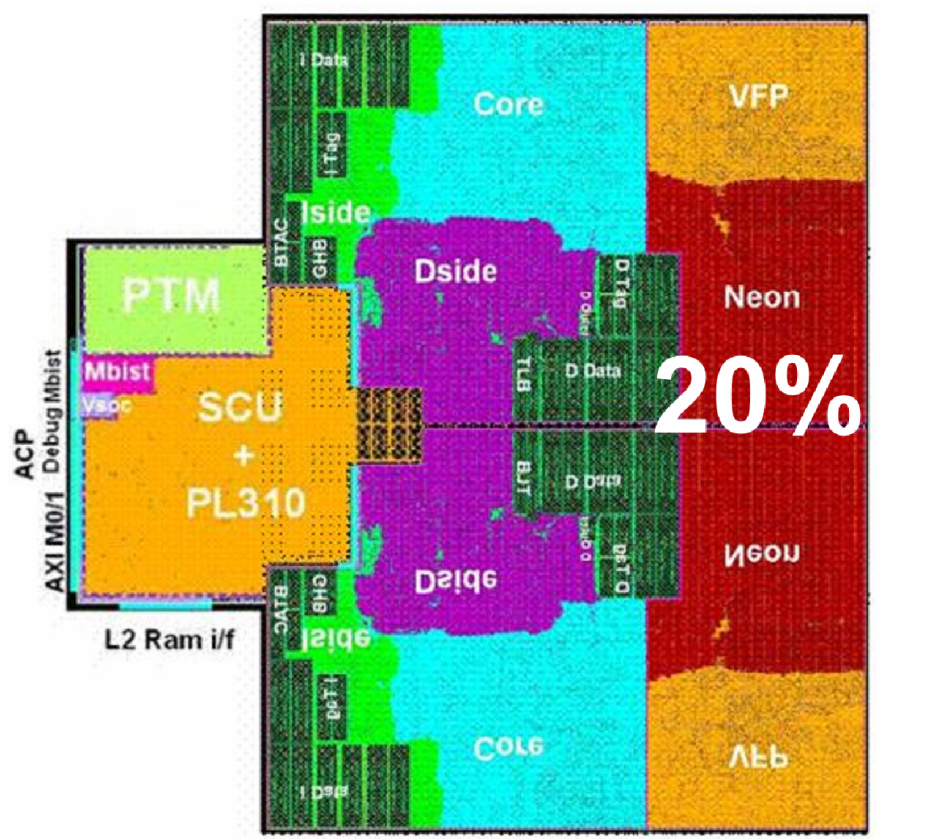




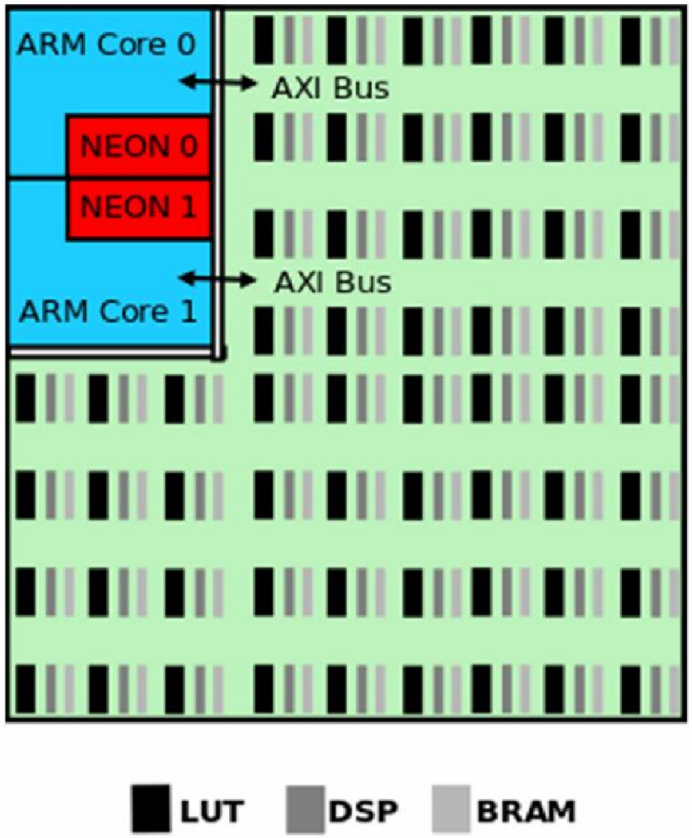
Reconfigurable Instruction Set Extensions

- **Let's replace the NEON vector unit with an FPGA fabric of ~identical size (i.e. 2080 LUTs, 16 DSPs, 8 BRAMs)**
- **Interesting for low precision SIMD arithmetic (128 bits allow 42 3-bit multiplications costing 1764 LUTs)**

Dual ARM A9 SoC Floorplan



Zynq chip with ARM SoC



Key Messages again

- **We have a once-in-a-lifetime chance to leave our niche**
- **We have to deliver yesterday rather than today**
- **This is a community effort (academia and industry)**

Contributors

- **Grigore Nicolae Bogdan (CDT)** nicolae.grigore@manchester.ac.uk
  Query optimization, resource management,
  FPGA virtualization
- **Malte Vesper (DSTL)** malte.vesper@manchester.ac.uk
  SSD stream processing infrastructure, applications
- **Raul Garcia (Conacyt)** raul.garcia@manchester.ac.uk
  Reconfigurable instruction set extensions
- **Christian Beckhoff** (hobbyist)
  GoAhead support (tool for building reconfigurable systems)
- **Edson Horta, Khoa Pham (H2020: ECOSCALE)** khoa.pham@manchester.ac.uk
  HLS support for PR and runtime management
- **Anuj Vaishnav (UniMan)** anuj.vaishnav@manchester.ac.uk
  Resource elastic FPGA virtualization, FPGA cloud infrastructure
- **Kristiyan Manev (UniMan)** kristiyan.manev@postgrad.manchester.ac.uk
  Resource elastic stream processing
- **Babis Kritikakis (UniMan)** babis_k4@hotmail.com
  Dynamic Dataflow on Maxeler