DESIGN, AUTOMATION & TEST IN EUROPE
25 – 29 March, 2019 · Firenze Fiera · Florence · Italy
The European Event for Electronic System Design & Test

Coherently Attached Programmable Near-Memory Acceleration Platform and its application to Stencil Processing

Jan van Lunteren, Ronald Luijten, Dionysios Diamantopoulos, Florian Auernhammer, Christoph Hagleitner, Lorenzo Chelini, Stefano Corda, Gagandeep Singh

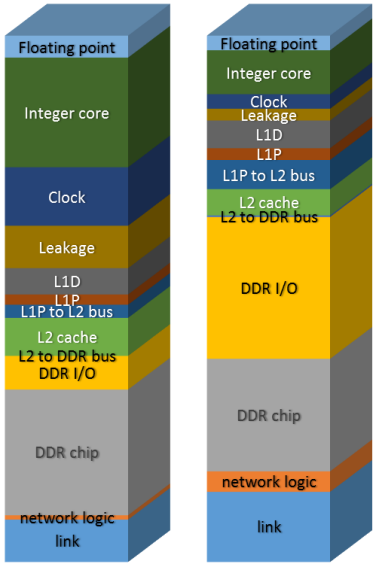
IBM Research – Zurich

Agenda

- Trends and challenges
- Near-memory acceleration platform
- Stencil computing in weather applications
- Compilation flow
- Implementation
- Conclusions

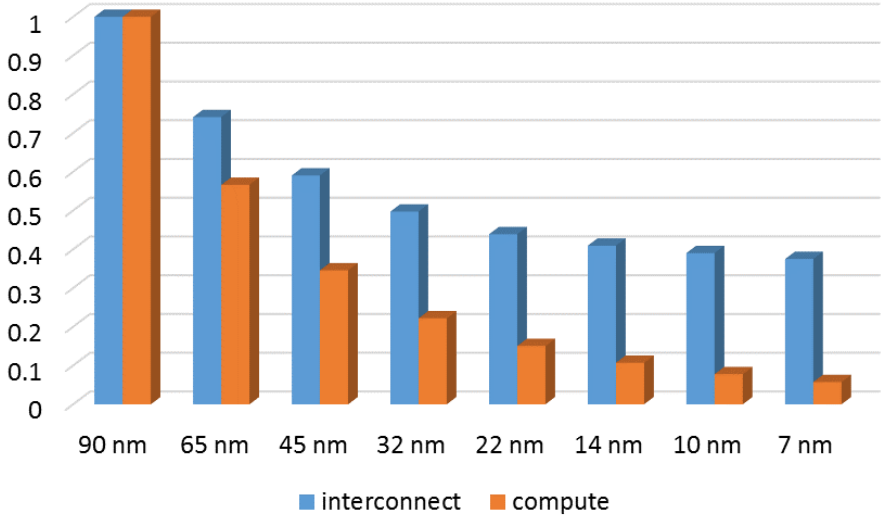
Trends and challenges

- Performance and power consumption are increasingly dominated by *data transfer* and *memory system operation*



HPC system-level power break-down

Source: R. Nair, "Active Memory Cube," 2nd Workshop on Near-Data Processing, 2014



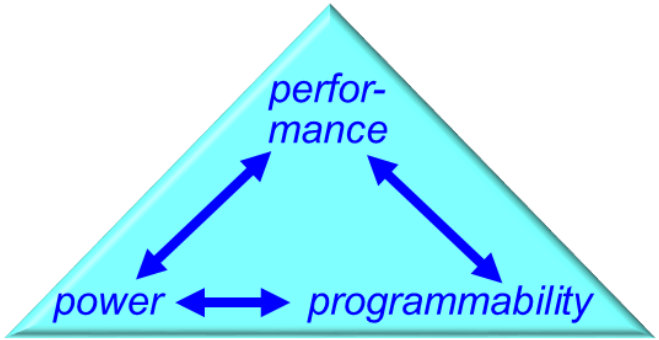
Chip-level energy trends

Source: S. Borkar, "Exascale Computing – a fact or a fiction?," IPDPS, 2013

Solutions

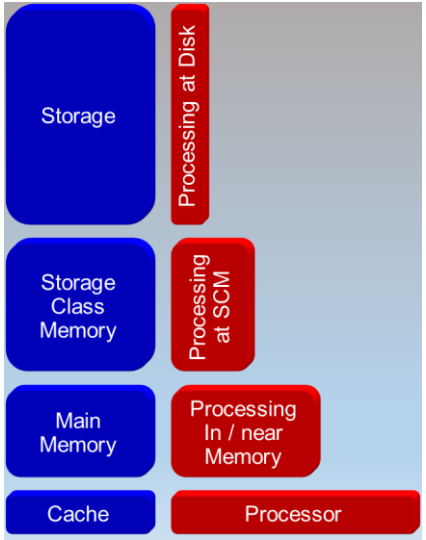
Specialization

- Workload-optimized systems
- General-purpose accelerators
  - GPUs, FPGAs, DSPs
- Fixed-function special-purpose accelerators (e.g., TPU)



Data-centric computing

- Bring computation closer to the data
- Reduce expensive data transfers by moving from a compute-centric to a data-centric model



Source: IBM

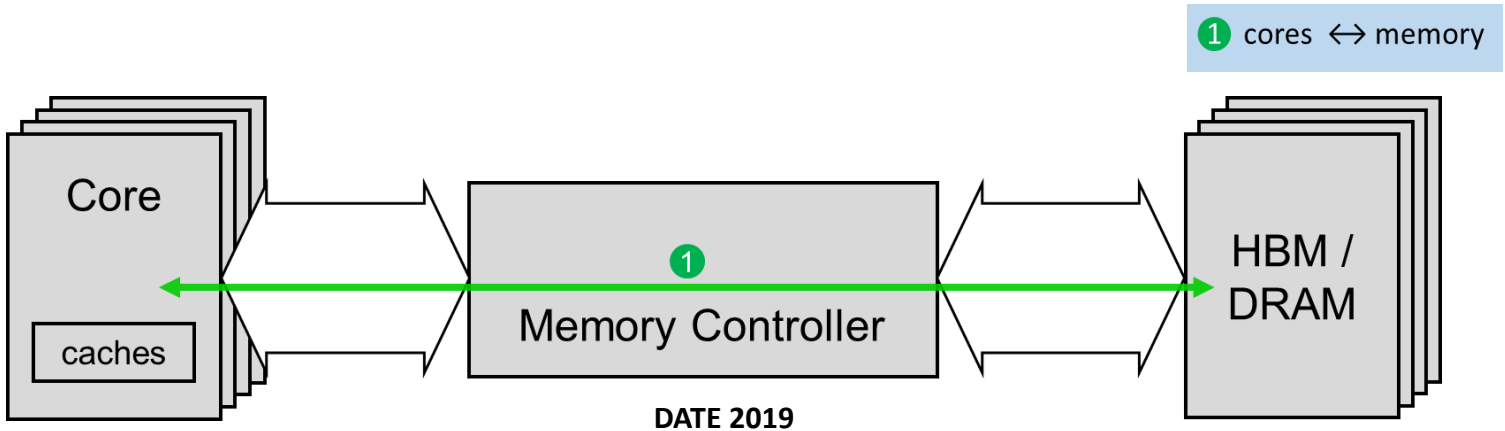
Near-memory acceleration platform

Conventional systems

- Memory system performance and power consumption depend on a complex interaction between workload and memory system
- No adaptive/programmable control over data flows (hardware-managed controllers)

➔ Negative impact on *memory-bound* applications (e.g., stencil computation)

- challenging to efficiently overlap computation and data access/transfer



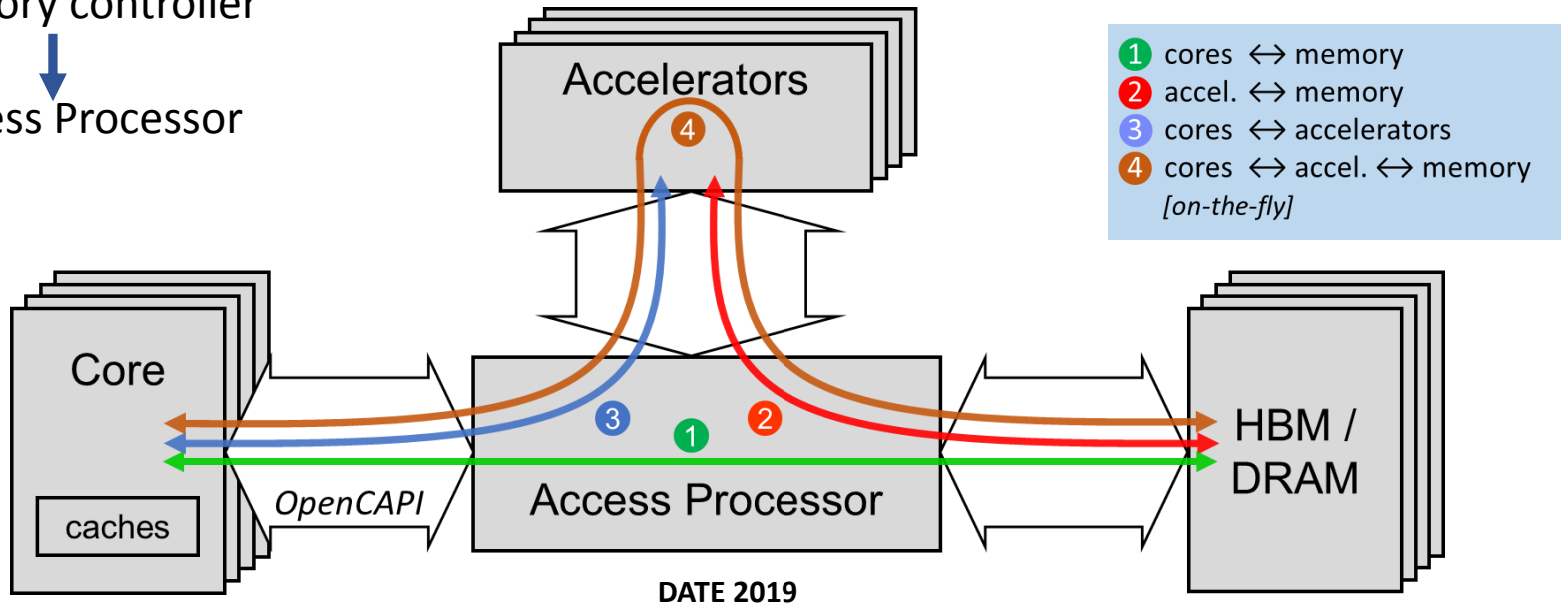
Near-memory acceleration platform

Our approach

- Programmable control over data flows between memory and near-memory accelerators
  - tightly control data access/transfer and computation to maximize overlap
  - optimize bandwidth utilization
- Memory controller

  ↓

  Access Processor

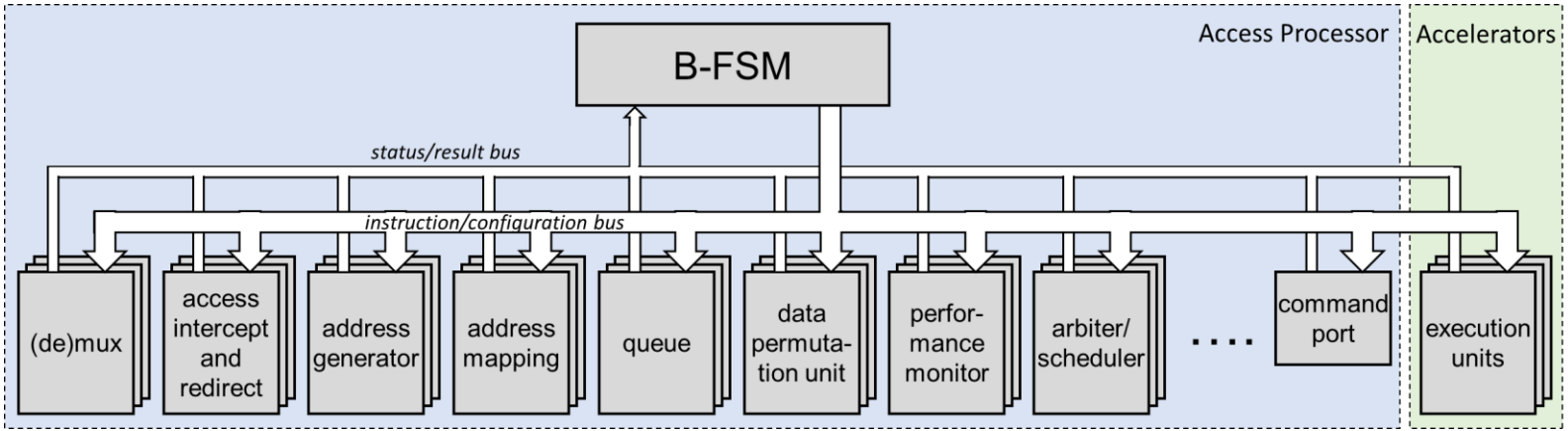


Near-memory acceleration platform

Enabling technology

- Access Processor micro-architecture based on B-FSM programmable state machine
- B-FSM monitors all units in parallel
  - evaluates hundreds of combinations of conditions in each clock cycle, and,
  - in response, dispatches instructions within 1-2 clock cycles (> 3 GHz, ASIC)

➔ manage data flows at speeds of tens of GB/s (DDR4) to hundreds of GB/s (HBM)

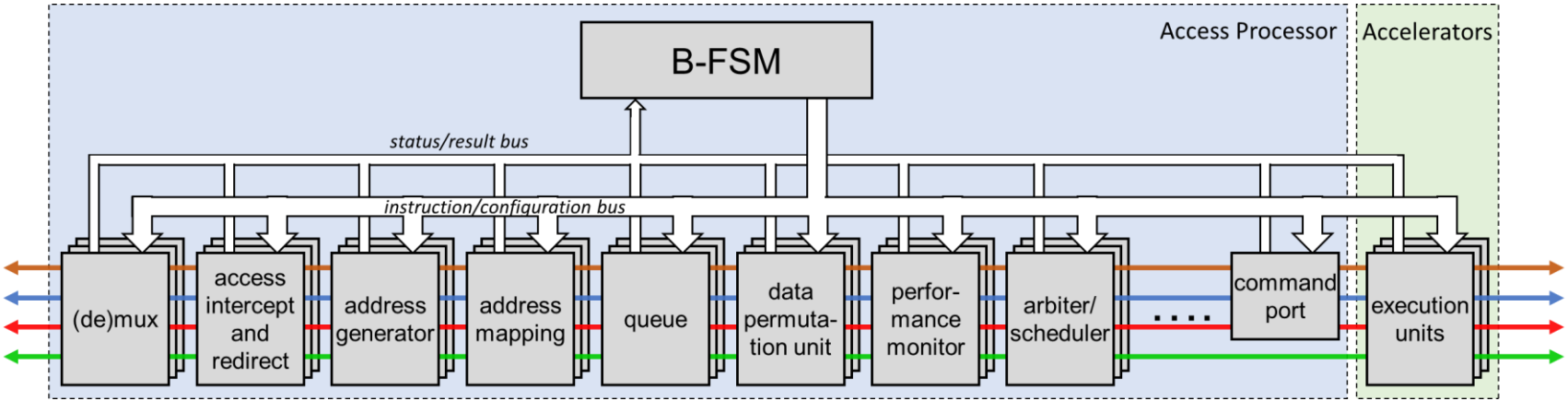


Near-memory acceleration platform

Enabling technology

- Access Processor micro-architecture based on B-FSM programmable state machine
- B-FSM monitors all units in parallel
  - evaluates hundreds of combinations of conditions in each clock cycle, and,
  - in response, dispatches instructions within 1-2 clock cycles (> 3 GHz, ASIC)

➔ manage data flows at speeds of tens of GB/s (DDR4) to hundreds of GB/s (HBM)

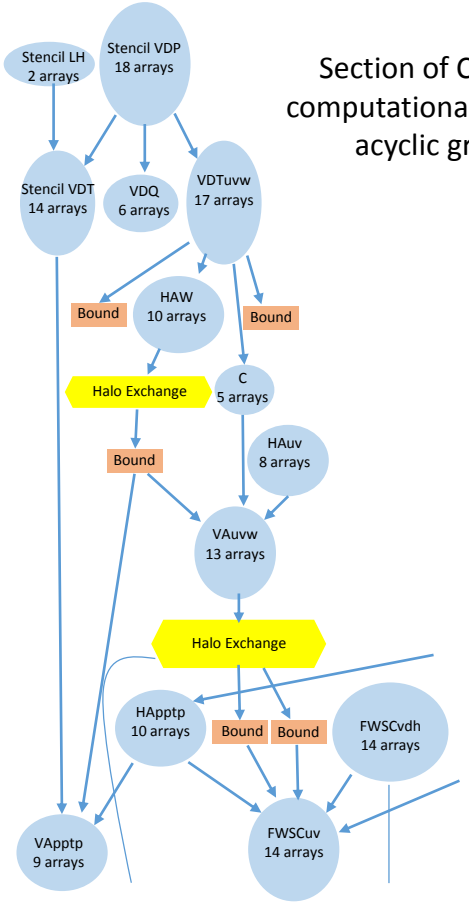


Stencil computing

Stencil computing in weather/climate applications

- Weather forecast programs (Cosmo, Icon, IFS,…) use stencil compute kernels to solve partial differential equations in the ‘dynamical core’ section
- *O*(100) different stencil compute motifs
- ~30 variable- and ~70 temporary arrays (3D grids)
- “Data in motion” typically exceeds caching capabilities in CPU and GPU
- Low compute intensity

➔ good candidate for near-memory acceleration



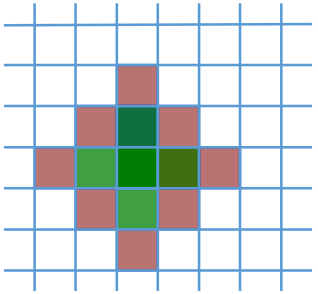
Section of COSMO computational directed acyclic graph

Stencil computing

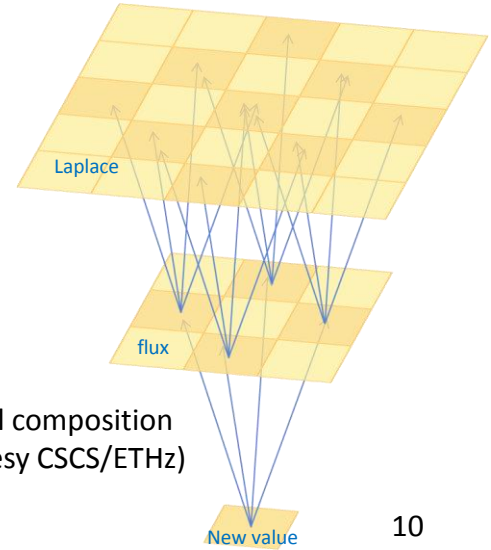
Stencil computing in weather/climate applications

- Stencil: operands fetched from ‘evaluation point’ and close neighbors (relative addressing)
- Stencils typically composition of elementary stencils
- Can use up to ~25 variable- / temporary arrays (3D)
- ‘vertical’ and ‘horizontal’ stencils typically alternate
- Array data layout chosen once for ‘dynamical core’
- Holistic optimization required at dynamical core level

➔ programmability of access processor

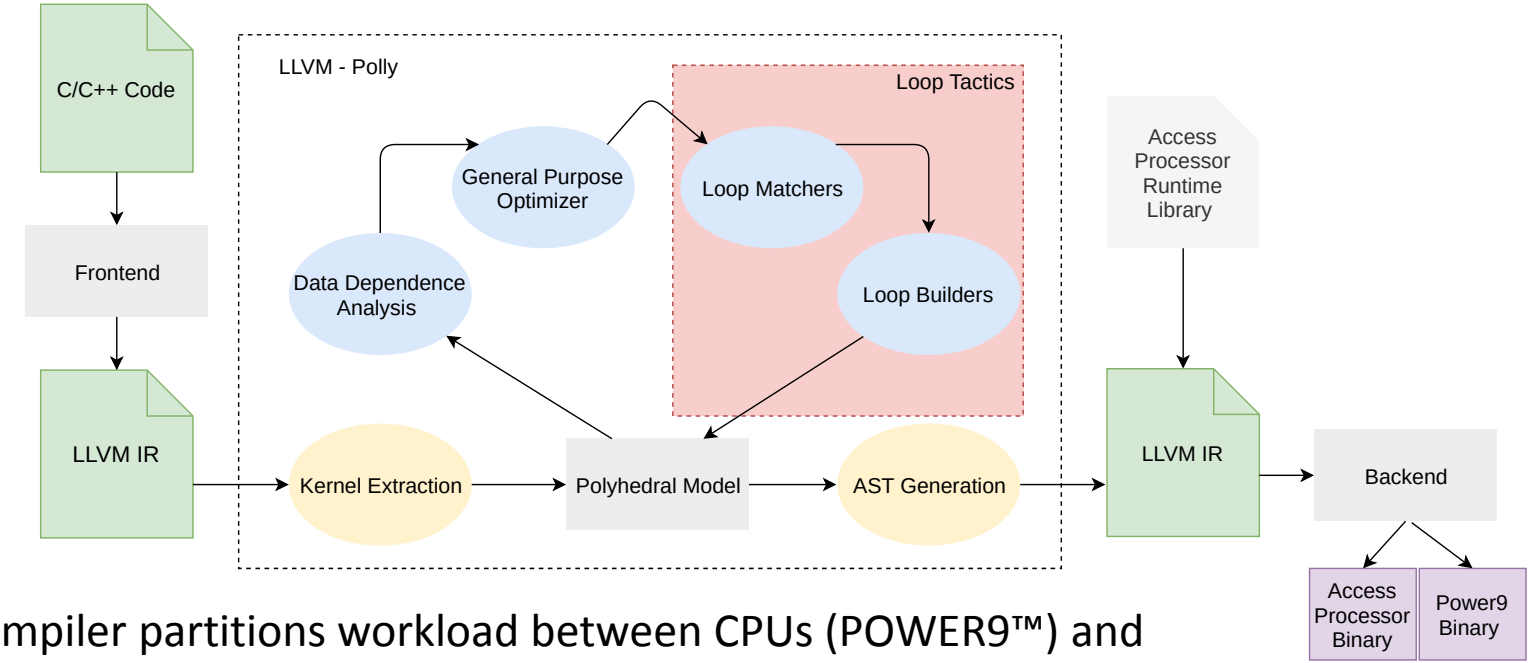


Neighbors
In 2D



Stencil composition
(Courtesy CSCS/ETHz)

Compilation flow



- Compiler partitions workload between CPUs (POWER9™) and Access Processor(s)
- Compiler is based on LLVM and exploits polyhedral optimization techniques (Polly) to analyze and optimize memory access patterns

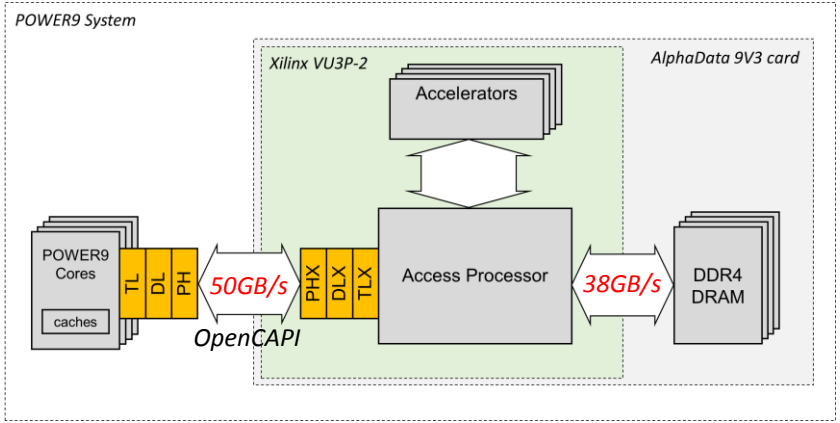
Implementation

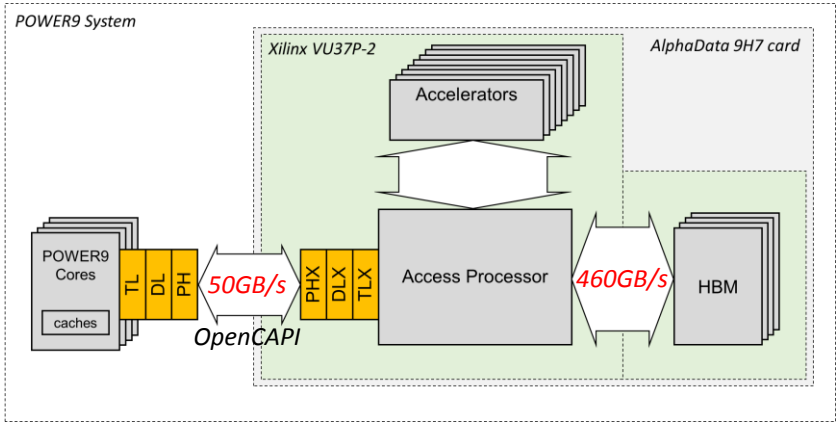
Components

- POWER9™ system
- AlphaData cards
  - 9V3: Xilinx® UltraScale+™ VU3P-2 FPGA
  - 9H7: Xilinx® UltraScale+™ VU37P-2 FPGA

OpenCAPI™

- Open Coherent Accelerator Processor Interface
- 50GB/s bidirectional bandwidth between POWER9™ and each AlphaData card
- Access-Processor-attached DDR4 DIMMs / HBM mapped into POWER9™ memory space and accessed like normal main memory



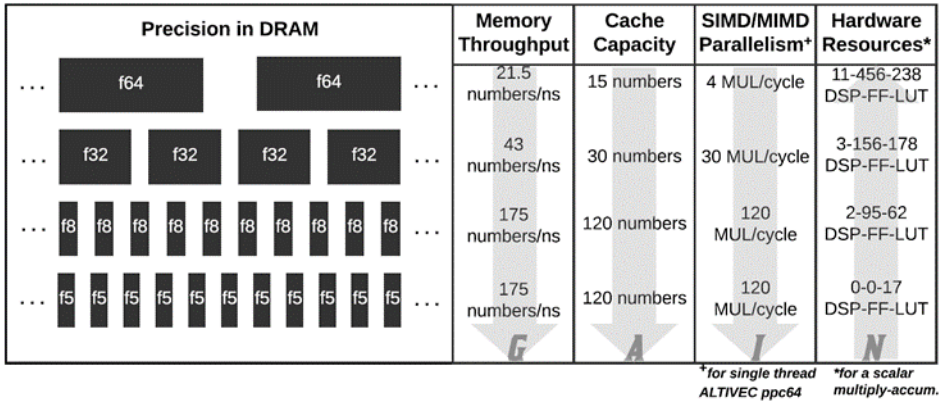


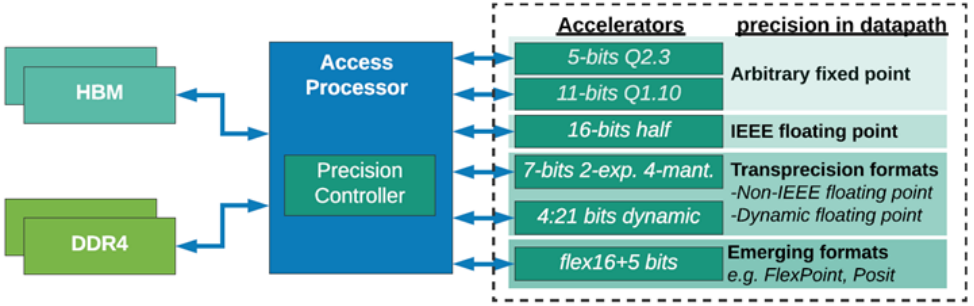
Transprecision

Transprecision in the Access Processor

- Hardware resources for a given silicon area may enable higher OPS at lower precision as these operations require less space and power
- Many operations are memory bandwidth bound: reducing precision enables improved cache utilization and reduction of bandwidth bottlenecks

➔ Precision controller is implemented as a modular extension to the B-FSM and enables precision control at design and run-time





Experimental results

Preliminary POWER8®/DMI-based prototype (DDR3, Altera® Stratix®-V)

- Demonstrated at OpenPOWER™ summit 2016/Nvidia® GTC
- 20-fold performance improvement for several functions (e.g., FFT, min/max) over optimized multi-threaded software implementations on POWER8®

POWER9™/OpenCAPI™-based prototypes

- DDR4 card operational recently (HBM card operational standalone)
- Initial experiments confirm that Access Processor can fully utilize the available memory bandwidth through overlap control for many applications

➔ processing performance (rate) then only dependent on memory bandwidth

- Measured on single DDR4 channel: ~9 GB/s read data, ~9 GB/s write results (total 18 GB/s)
  - 1024-pts FFT (64-bit complex sample): 1.15 GSamples/sec processing rate
  - 2D Jacobi stencil (64x64 matrix, double-precision): 280K matrix updates/second (R+W)

(note: actual bandwidth/processing rates fluctuate due to DRAM refresh cycles and host access)

Conclusions

Near-memory acceleration

- "Just scaling up" today's HPC systems is not sufficient to arrive at exascale systems
- Near-memory acceleration might be one of the key innovations to realize this objective for an important class of applications
- Besides optimizing performance and power by reducing data transfers, Access Processor-based near-memory acceleration offers new opportunities to
  - make memory system operation programmable such that it can be adapted to workload characteristics and used for balancing performance and power
  - realize new processing architectures that reduce (programmability) overhead by exploiting available regularity in processing and/or data access
- Ongoing work focuses on evaluating the performance impact of the near-memory acceleration platform (DDR4, HBM) at application level for a wide range of workloads

Acknowledgement

The contributions of Lorenzo Chelini, Stefano Corda, and Gagandeep Singh were performed in the framework of Horizon 2020 program for the project "Near-Memory Computing (NeMeCo)" which is funded by European Commission under Marie Sklodowska-Curie Innovative Training Networks European Industrial Doctorate (Project ID: 676240)