KALRAY

Consolidating High-Integrity, & High-Performance Functions on a Manycore Processor

Benoît Dupont de Dinechin
CTO

July 2020

www.kalrayinc.com

Kalray in a Nutshell

Kalray is a fabless company offering a new type of processor targeting the booming market of intelligent systems.

A Global Presence

- France (Grenoble, Sophia-Antipolis)
- USA (Los Altos, CA)
- Japan (Yokohama)
- China (Partner)
- South Korea (Partner)

10+ year's experience in Manycore

3rd generation of MPPA® processor

~€85m R&D investment

30 Patent families

Industrial investors

NXP · RENAULT NISSAN MITSUBISHI · SAFRAN · MBDA

EURONEXT

- **Public Company** (ALKAL)
  ~ €90M EQUITY RAISED
- **Strong support from European Governments**

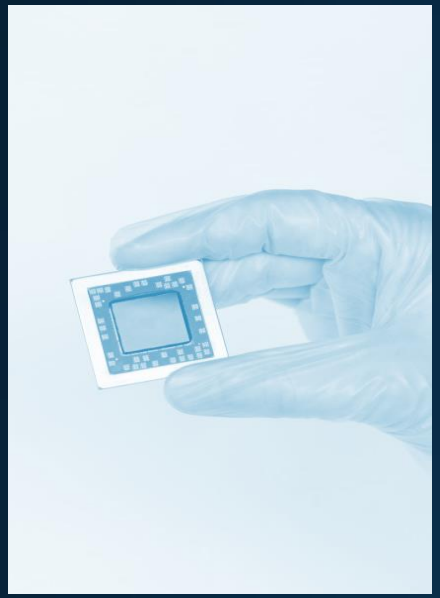
Financial Investors: CEA Investissement, BpiFrance, ACE, INOCAP Gestion, Pengpai

KALRAY

Outline

1. MPPA®3 Manycore Processor
2. Standard Programming Environments
3. Model-Based Development Environments

Kalray's MPPA®



MPPA® (Massively Parallel Processor Array) Platform

Hardware

Manycore CPU architecture

Compute clusters of 16 high-performance CPU cores with local memory

DSP-like timing predictability

'Fully timing compositional' cores for accurate static timing analysis

Service guarantees of local memory system and network-on-chip

FPGA-like I/O capabilities

Software

CPU programming

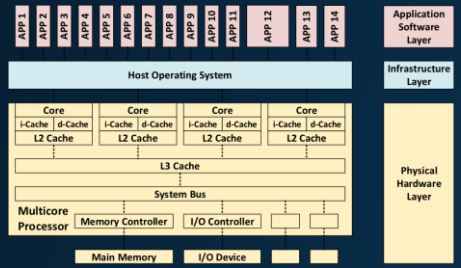
Standard C/C++/OpenMP/OpenCL, OpenVX

Library code generators (MetaLibm, KaNN)

Model-based (SCADE Suite®, Simulink®)

Multicore and Manycore Processors

Homogeneous Multicore Processor

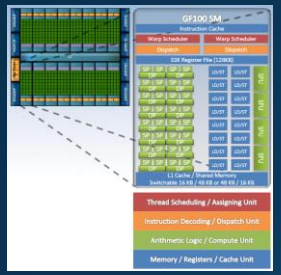


Multiple CPU cores sharing a cache-coherent memory hierarchy
- Scalability by replicating CPU cores
- Standard programming models

Energy efficiency issues
- Global cache coherence scaling

Time-predictability issues
- No scratch-pad or local memories

GPGPU Manycore Processor

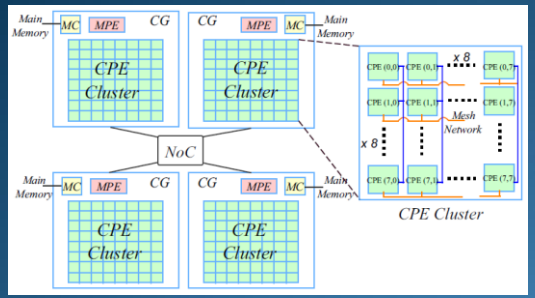


Multiple Streaming Multiprocessors
- Restricted programming models

Performance issues of 'thread divergence'
- Branch divergence slow down the execution
- Memory divergence: non-coalesced accesses

Time-predictability issues
- Dynamic allocation of thread blocks
- Dynamic scheduling of warps

CPU-Based Manycore Processor



Multiple "Compute Units" connected by a network-on-chip (NoC)
- Scalability by replicating Compute Units
- Standard multicore programming inside a Compute Unit

Compute Unit
- Group of cores + DMA
- Scratch-pad memory (SPM)
- Local cache coherency

KALRAY

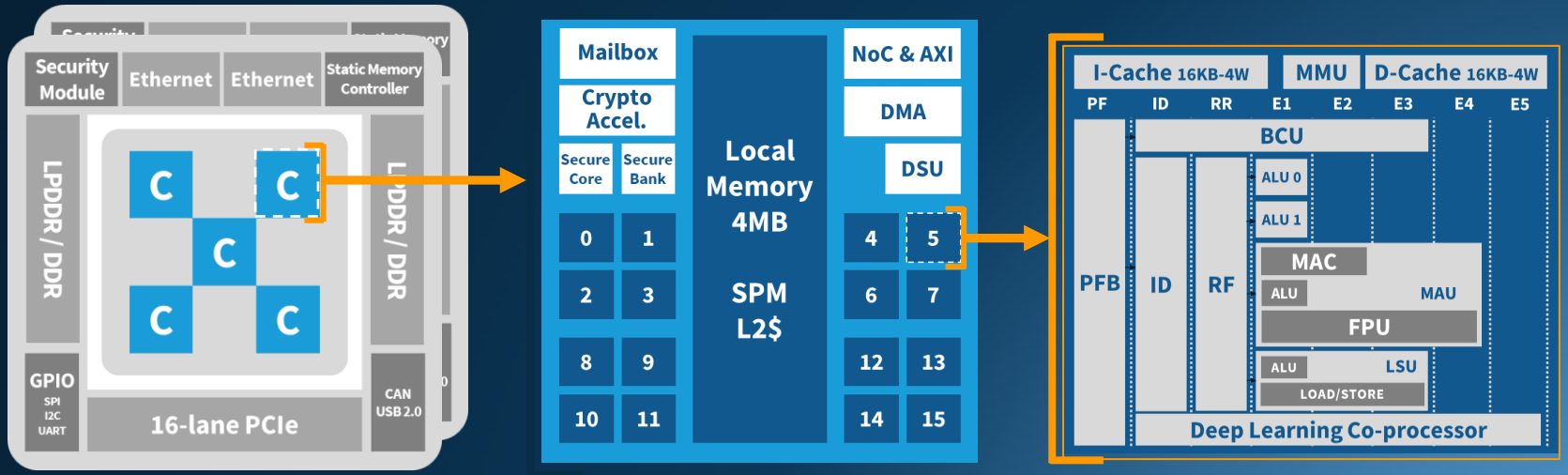
MPPA® Processor Family and Roadmap

SAMPLES AVAILABILITY ····· ▶

		2019	2020 (IP3) / 2021 (IC)
	BOSTAN2	**COOLIDGE1-80**	**COOLIDGE2 – 80** **COOLIDGE2 – 160**
PROCESS	28 nm	16 nm	16 nm
FIXED POINT OPERATIONS	1.3 TOPS	25 TOPS (8bit)	50 / 100 TOPS (8bit)*
FLOATING POINT OPERATIONS	512 GFLOPS	4 TFLOPS (16.32bit)	16 / 32 TFLOPS (16.32bit)*
DMIPs	250 KDMIPS	190 KDMIPS	190 / 380 KDMIPS
CONSUMPTION (Typ.)	8 – 25W	5 – 25W	5 – 30W / 5 – 60W
FEATURES	• 288 Kalray VLIW Cores • 128 Crypto Copro • 2xDDR3 • 8x 1/10G GbE • 2xPCIe 8 lane Gen3	• 80 Kalray 64-bit cores • 80 Coprocessor for vision and learning • 2 x LP/DDR4 • 8x 1/10/25GbE • 16-lane PCIe Gen4	• 80/160 Kalray 64-bit cores • 80/160 Coprocessor for vision and learning • 2 x LP/DDR4 • 8x 1/10/25GbE • 16-lane PCIe Gen4
QUALIF/CERTIF	Industrial (-20/+85C)	• AEC-Q100 / QM	• ASIL B / ISO 26262
TARGET MARKET	• DATA CENTER • AUTO (proto)	• DATA CENTER • AUTOMOTIVE	• DATA CENTER • AUTOMOTIVE
		AVAILABLE (IC and IP)	UNDER DEVELOPMENT (IC and IP)

KALRAY

Kalray MPPA® Manycore Processor



COOLIDGE PROCESSOR

5 compute clusters at 1200 MHz
2x 100Gbps Ethernet, 16x PCIe Gen4

COMPUTE CLUSTER

16+1 cores, 4 MB local memory
NoC and AXI global interconnects

6-ISSUE VLIW CORE

64x 64-bit register file
128MAC/c tensor coprocessor

C Cluster

KALRAY

Network-on-Chip for Global Interconnects

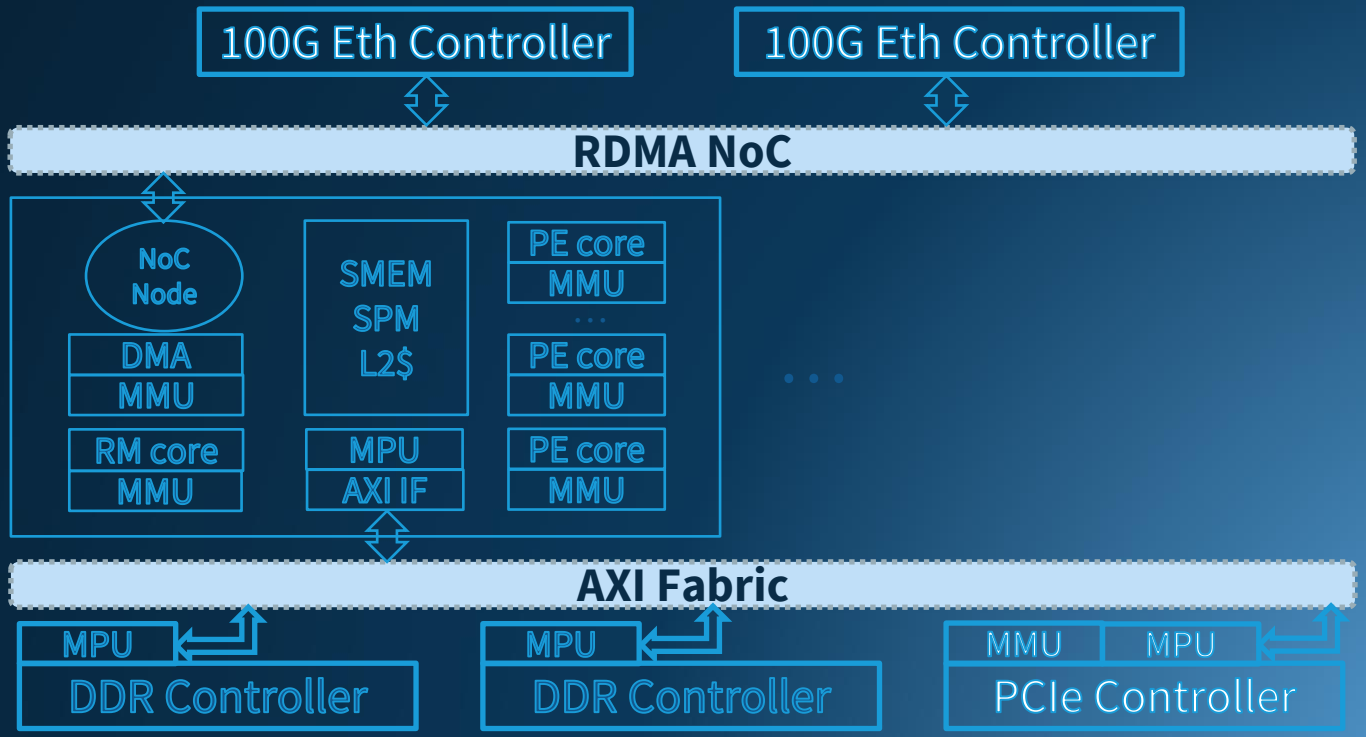
NoC as generalization of busses

- Connectionless
- Address-based transactions
- Flit-level flow control
- Implicit routing
- Inside a coherence domain
- Reliable communication
- Coherency protocol messages
- Coordinate with DDR memory controller front-end (Ex. Arteris FlexMem Memory Scheduler)

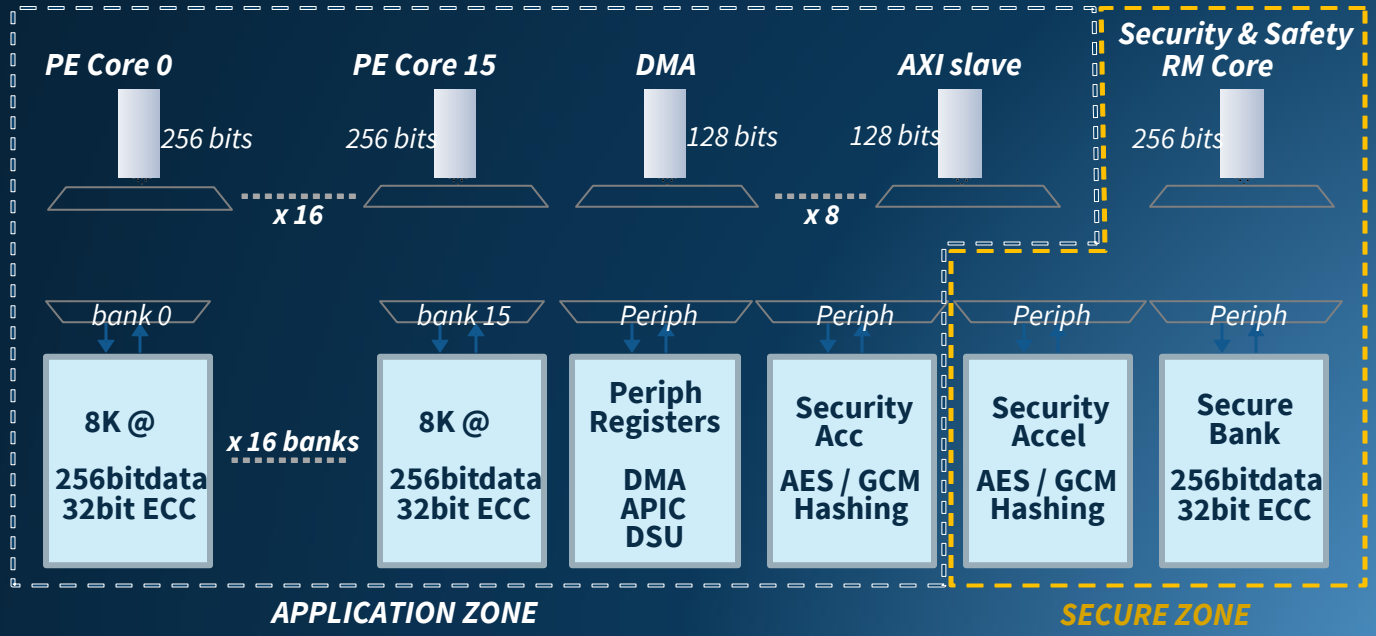
NoC as integrated macro-network

- Connection-oriented
- Stream-based transactions
- [End-to-end flow control]
- Explicit routing
- Across address spaces (RDMA)
- [Packet loss or packet reordering]
- Traffic shaping for QoS (application of DNC)
- Terminate macro-network (Ethernet, InfiniBand)
- Support of multicasting

MPPA®3 Global Interconnects



MPPA®3 Cluster Interconnect



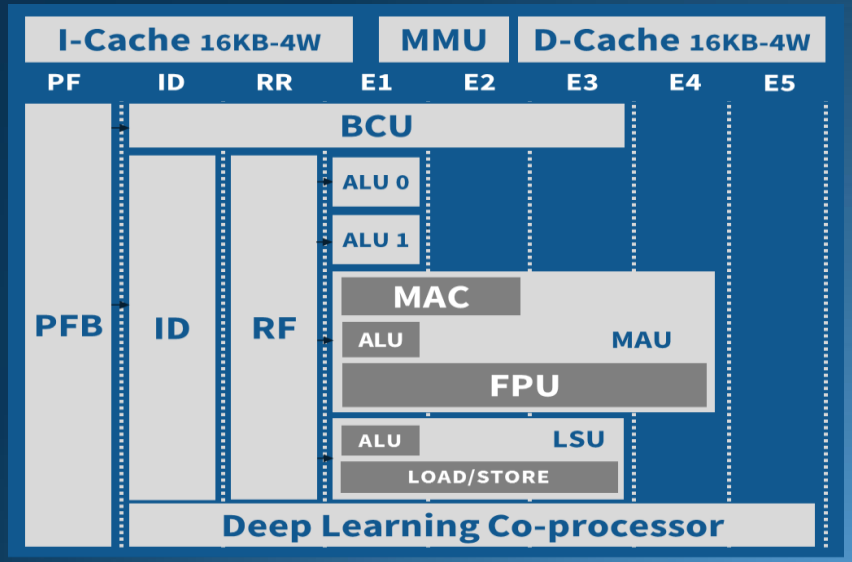
MPPA®3 64-Bit VLIW Core

Vector-scalar ISA

- 64x 64-bit general-purpose registers
- Operands can be single registers, register pairs (128-bit) or register quadruples (256-bit)
- Immediate operands up to 64-bit, including F.P.
- 128-bit SIMD instructions by dual-issuing 64-bit on the two ALUS or by using the FPU datapath

FPU capabilities

- 64-bit x 64-bit + 128-bit → 128-bit
- 128-bit op 128-bit → 128-bit
- FP16x4 SIMD 16 x 16 + 32 → 32
- FP32x2 FMA, FP32x4 FADD, FP32 FMUL Complex
- FP32 Matrix Multiply 2x2 Accumulate



VLIW CORE PIPELINE

MPPA®3 Tensor CoProcessor

Extend VLIW core ISA with extra issue lanes

- Separate 48x 256-bit wide vector register file
- Matrix-oriented arithmetic operations (CNN, CV …)

Full integration into core instruction pipeline

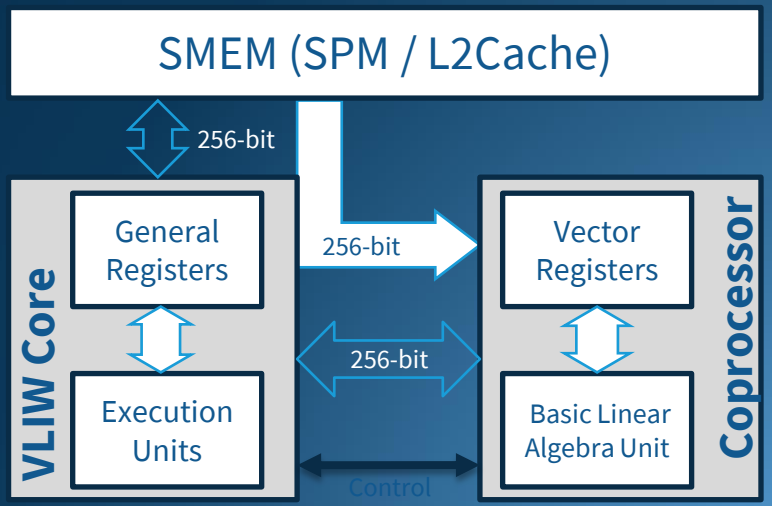
- Move instructions supporting matrix-transpose
- Proper dependency / cancel management

Leverage MPPA memory hierarchy

- SMEM directly accessible from coprocessor
- Memory load stream aligment operations

Arithmetic performances (MPPA3-v1)

- 128x INT8→INT32 MAC/cycle
- 64x INT16→INT64 MAC/cycle
- 16x FP16→FP32 FMA/cycle



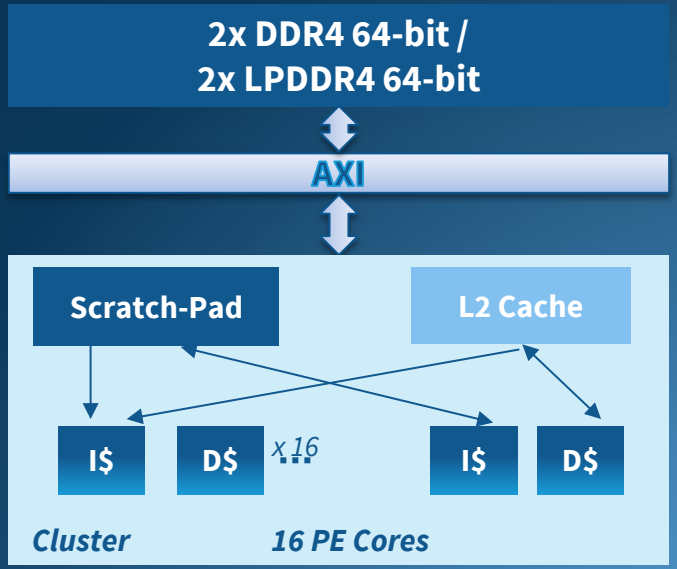
MPPA®3 Memory Hierarchy

VLIW Core L1 Caches

- 16KB / 4-way LRU instruction cache per core
- 16KB / 4-way LRU data cache per core
- 64B cache line size
- Write-through, write no-allocate (write around)
- Coherency configurable across all L1 data caches

Cluster L2 Cache & Scratch-Pad Memory

- Scratch-pad from 2MB to 4MB
  - 16 independent banks, full crossbar
  - Interleaved or banked address mapping
- L2 cache from 0MB to 2MB
  - 16-way Set Associative
  - 256B cache line size
  - Write-back, write allocate



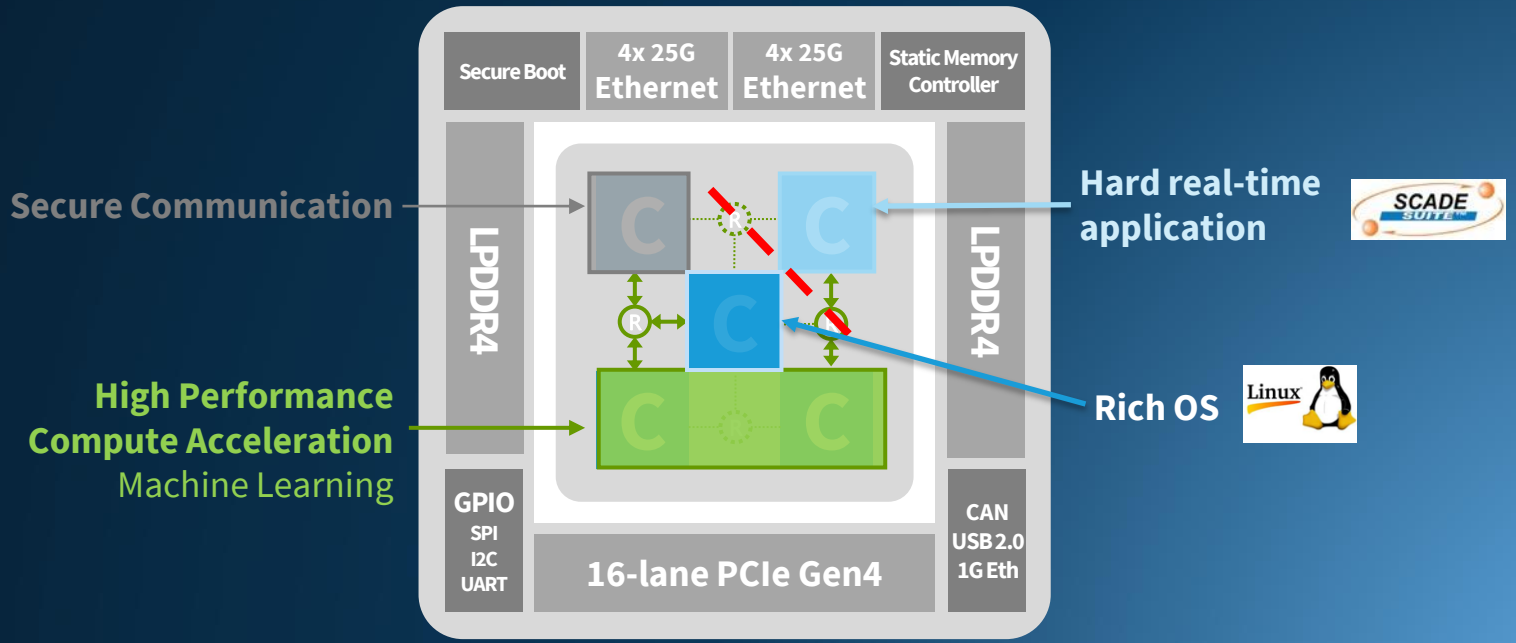
L1 cache coherency	L2 cache coherency
enable /disable	enable /disable

Outline

1. MPPA®3 Manycore Processor
2. Standard Programming Environments
3. Model-Based Development Environments

Mapping Functions to Compute Units

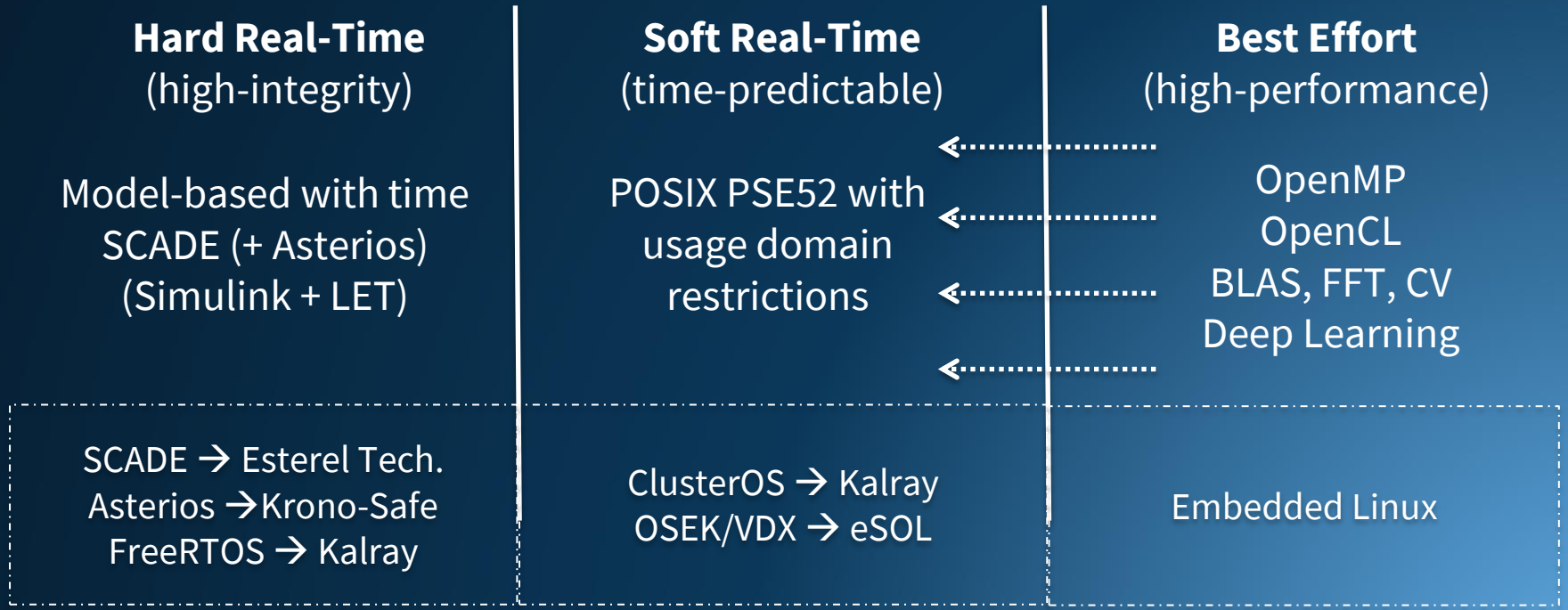
Run Multiple application and Multiple OS Concurrently



C Cluster

KALRAY

MPPA® Embedded Platform



High-Performance Programming Models

STANDARD PROGRAMMING ENVIRONMENTS

OPENCL 1.2 Programming

Standard accelerator programming model for offloading on MPPA®

- POSIX host CPU accelerated by MPPA device (OpenAMP interface)
- OpenCL 1.2 compatibility with POCL environment and LLVM for OpenCL-C
- OpenCL offloading modes:
  - Linearized Work Items on a PE (LWI)
  - Single Program Multiple Data (SPMD)
  - Native code called from kernels

C/C++ POSIX Threads Programming

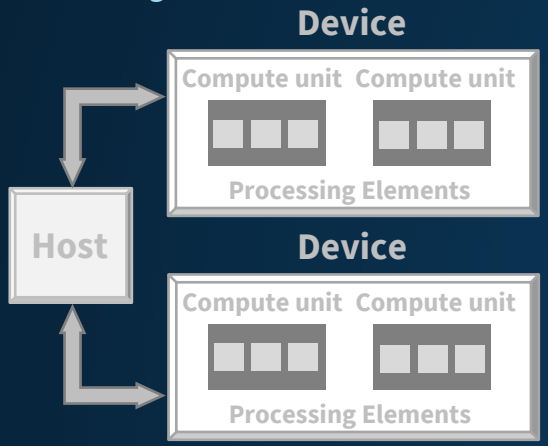
Standard multicore programming model with exposed MPPA® communications

- MPPA Linux and ClusterOS
- Standard C/C++ programming
  - GCC, GDB, Eclipse system trace
- POSIX threads interface
- GCC OpenMP support
- RDMA using the MPPA Asynchronous Communication library (mppa_async)

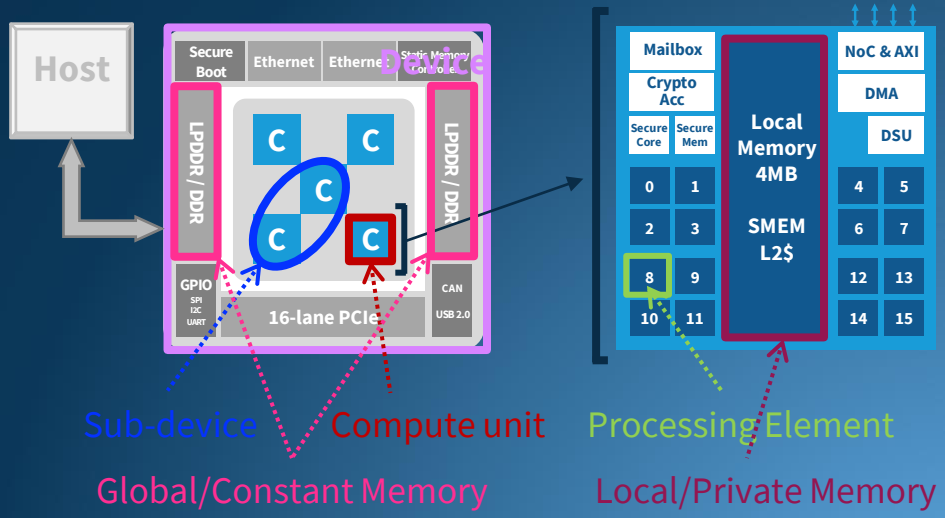
OpenCL Compute Platform Mapping for MPPA

OpenCL Compute Platform Model

Topology: Host CPU connected to one or several Device(s)
Host: CPU which runs the application under a rich OS (Linux)
Device: Compute Unit(s) sharing a Global Memory
Hierarchy: Multi-Device => Device => Sub-Device => Compute Unit(s) => Processing Elements

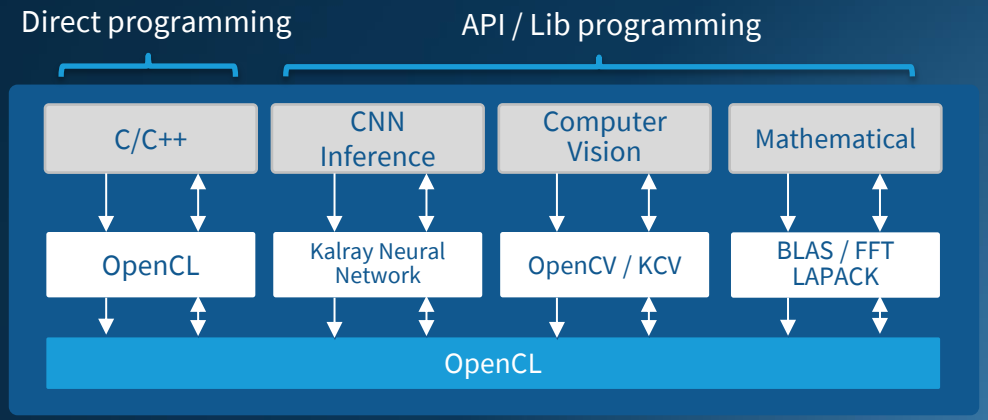


‘SPMD’ Mapping to MPPA® Architecture



Kalray Acceleration Framework (KAF™)

A versatile way to program manycore architecture based on OpenCL



OpenCL Native Function Extension

- Enable to call ASM, C/C++/OpenMP/POSIX (ClusterOS) code from OpenCL kernels
- Generalization of TI 'OpenMP Dispatch With OpenCL' for KeyStone-II platforms
- Used by Kalray KaNN deep learning compiler
- Used by BLAS and multi-cluster libraries

```
void
my_vector_add(int *a, int *b, int *c, int n)
{
  #pragma omp parallel for
  for (int i = 0; i < n; ++i)
  {
    c[i] = a[i] + b[i];
  }
}
```

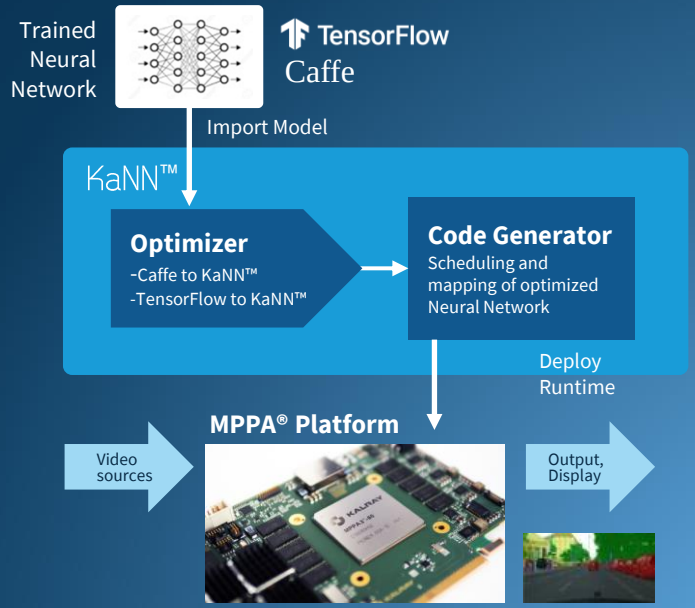
```
__attribute__((mppa_native))
void my_vector_add(__global int *a, __global int *b, __global int *c, int n);

__kernel void vector_add(__global int *a, __global int *b, __global int *c, int n) {
  my_vector_add(a, b, c, n);
}
```

KaNN™, Kalray Neural Network, Inference Compiler

From standard Machine Learning frameworks
to code generation, setup and multiple CNN execution

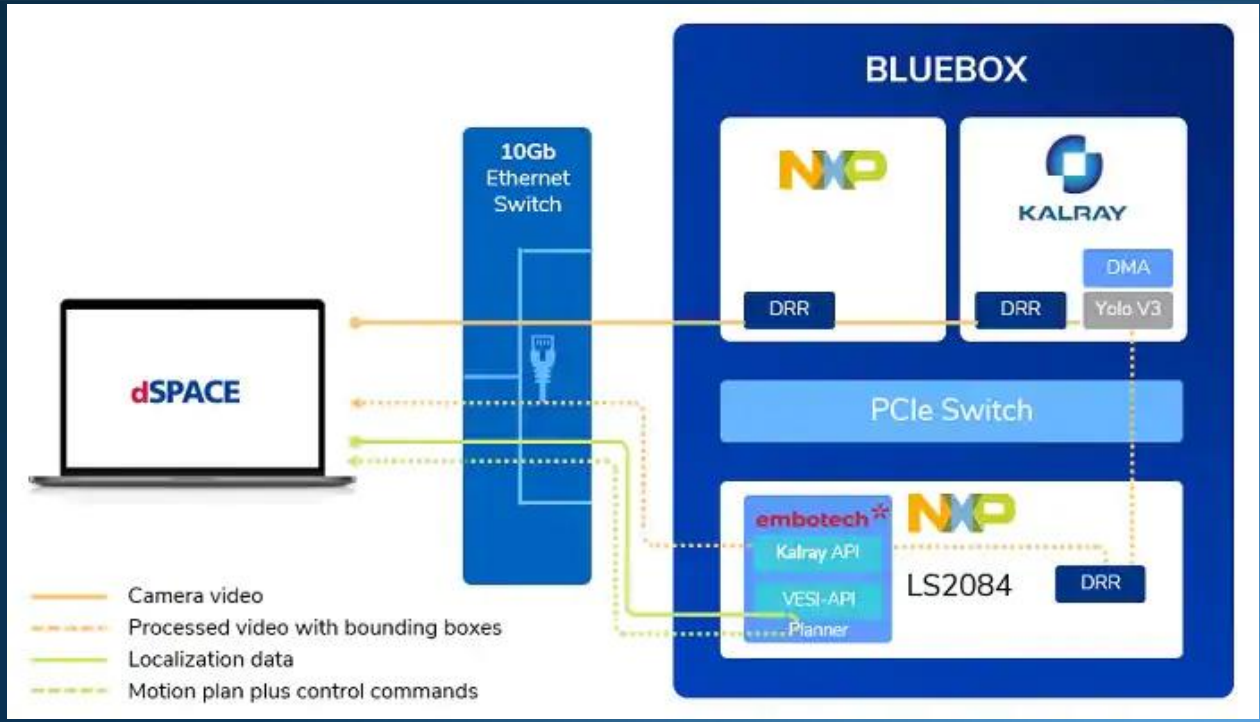
Deep Learning Inference Code Generator
- Optimization of neural networks for MPPA®
- Deployment of neural networks on MPPA®

Deep Learning Inference Runtime Support of:
- Major frameworks
- Major networks
- Custom networks



CES 2020 NXP Demonstration

- NXP BlueBox 2nd generation Autonomous Driving Development platform with production ready automotive silicon
- Kalray Coolidge 3rd Generation MPPA Perception Accelerator and AI Software (Yolo v3 416x416)
- Embotech Forces Pro and ProCruiser Real-time optimal control software and Highway planner solution
- dSPACE ASM Traffic Real time simulation environment with traffic, sensor simulation, full VD and BEV powertrain.



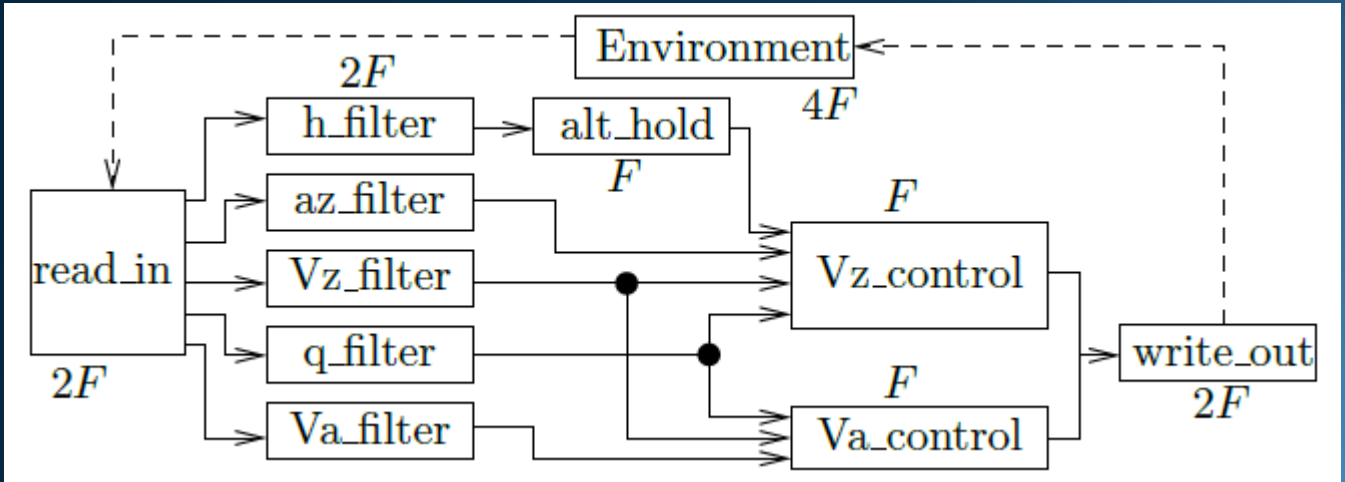
KALRAY

Outline

1. MPPA®3 Manycore Processor
2. Standard Programming Environments
3. Model-Based Development Environments

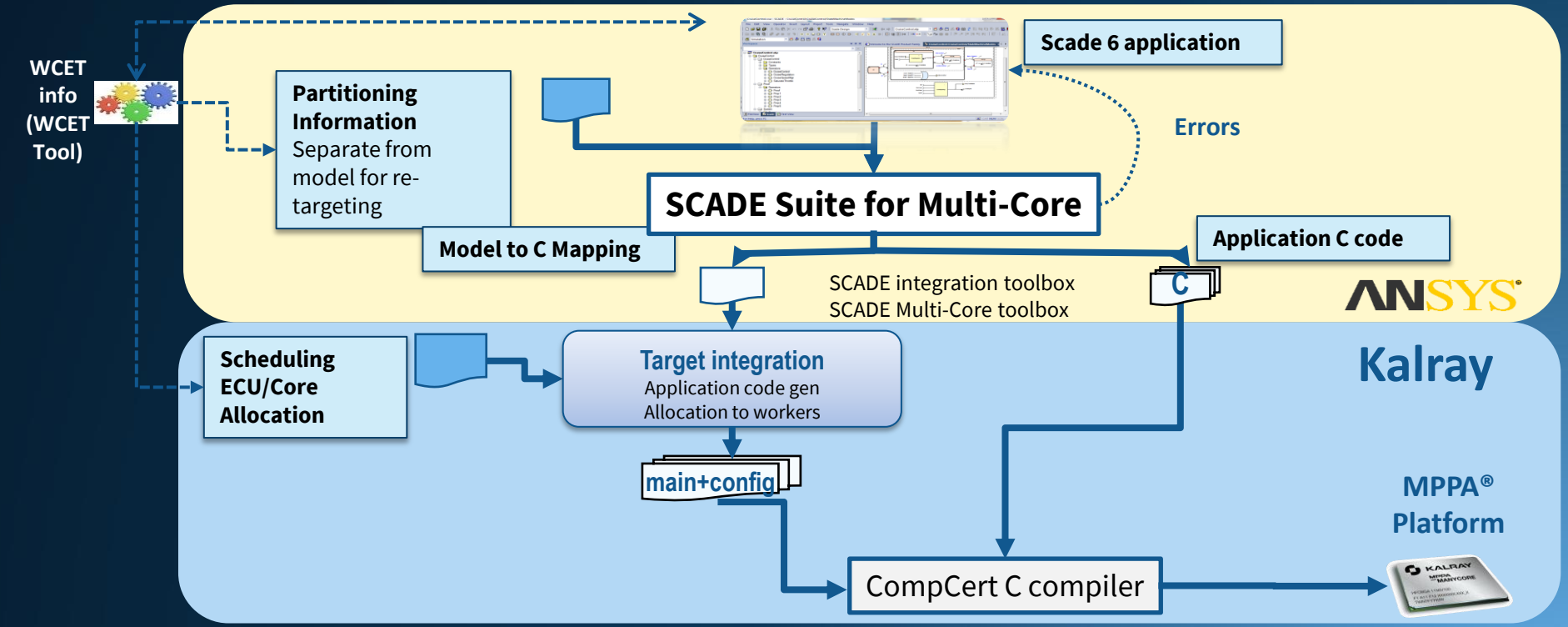
ROSACE Case Study for MBD on Multicore

- Simplified controller for the longitudinal motion of a medium-range civil aircraft in en-route phase: cruise and change of cruise level sub-phases



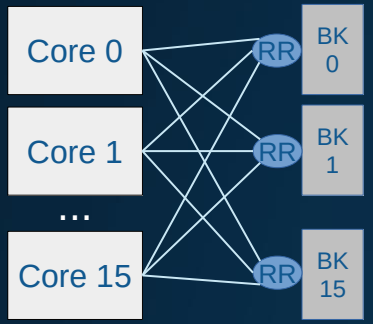
- Application has 3 harmonic periods: F, 2F, 4F

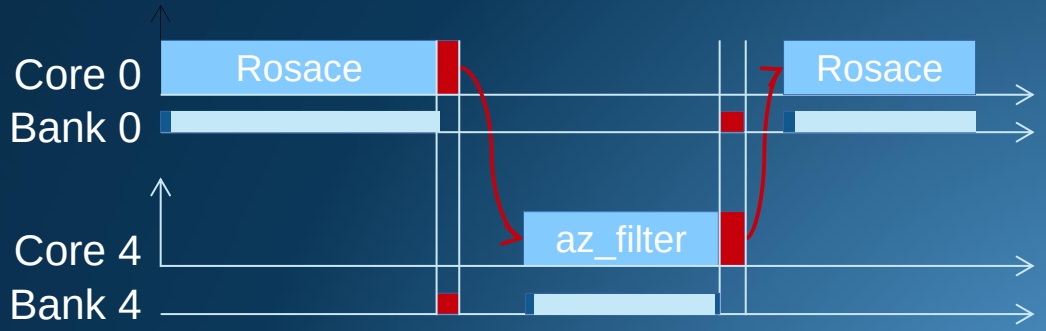
SCADE Suite Multi-Core Code Generation Flow



SCADE Suite MCG Code Generation

- Exploit the MPPA cluster configuration for ‘high-integrity’ execution
  - Enable the cluster local memory mapping of one bank per core

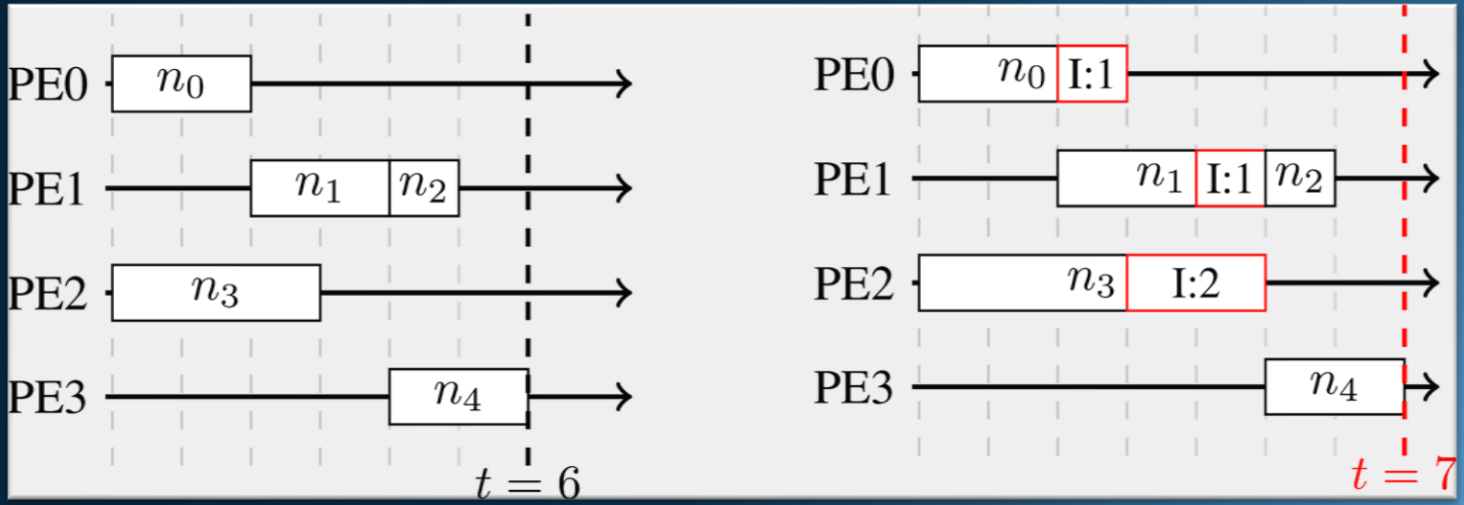




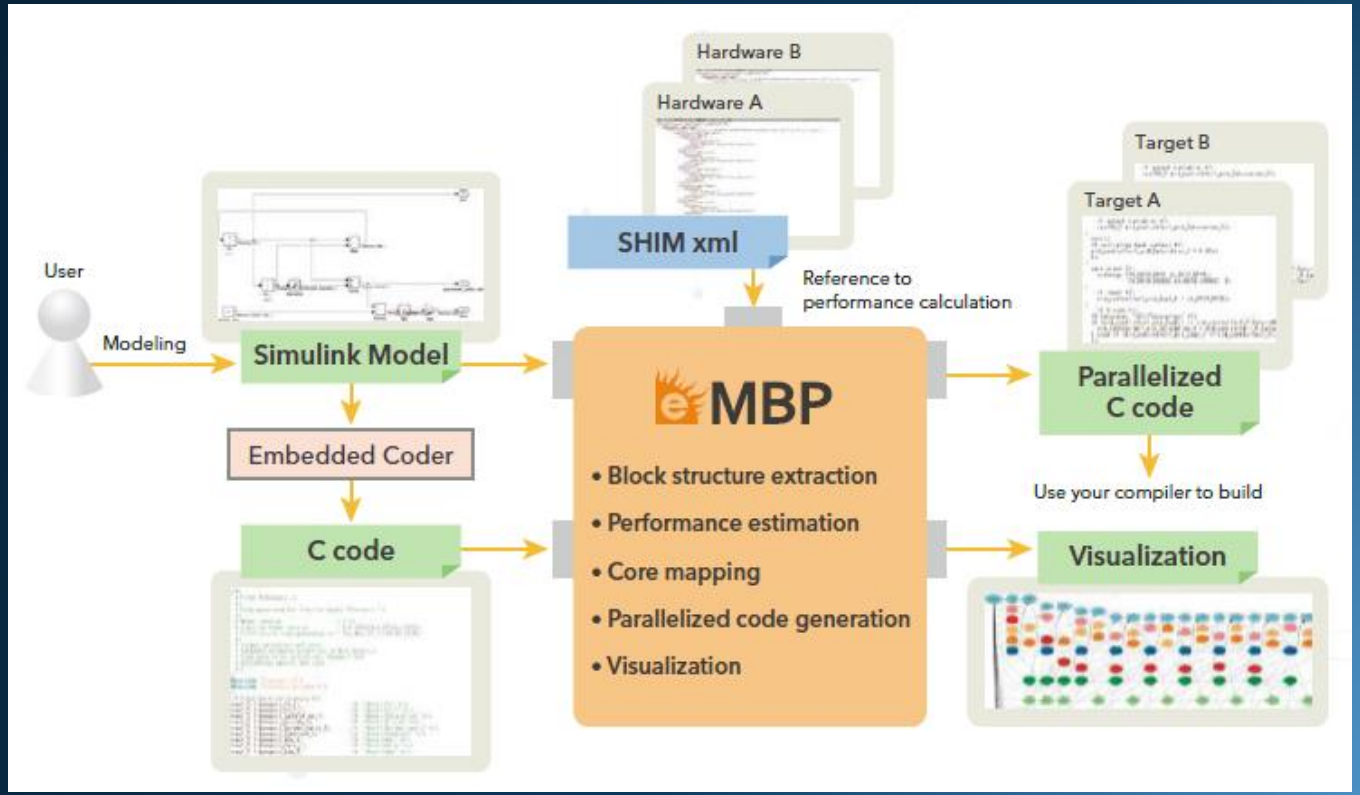
- Precisely compute the task WCETs (Worst-Case Execution Times)
  - Static analysis or measurement for the WCET of tasks in isolation
  - Refine the WCET with interferences [Rihani RTNS'16][Schuh DATE’20]

Time-Triggered Multicore Scheduling [Schuh DATE'20]

- Given a task mapping and release dates, schedule by forward time sweep
- Release a task when its dependencies are satisfied and after its release date
- Adjust interferences considering to the subset of curently executing tasks



eSOL eMBP Multi-Core Code Generation Flow



COOLIDGE™
3rd Generation of MPPA® Processor

The Compute Solution for Next Generation Vehicles

AI Acceleration and much more!

- Computing power
- Data processing in real time
- Accelerate Multiple Applications in parallel
- Power efficiency
- Programmable/Open systems
- Security & Safety (ASIL-B)

KALRAY

Thank You

KALRAY S.A.
Corporate Headquarters
180, avenue de l'Europe
38 330 Montbonnot, France
Phone: +33 (0)4 76 18 90 71
contact@kalrayinc.com

CERTIFIED ORGANIZATION ISO 9001
CERTIFICATION IN PROGRESS ISO 26262

KALRAY INC.
America Regional Headquarters
4962 El Camino Real
Los Altos, CA - USA
Phone: +1 (650) 469 3729
contact@kalrayinc.com

KALRAY JAPAN - KK
Represented by MACNICA Inc. Strategic Innovation Group
Macnica Building, No.1, 1-6-3 Shin-Yokohama
Kouhoku-ku, Yokohama 222-8561, Japan
Phone: +81-45-470-9870

KALRAY S.A.
Sophia-Antipolis
1047 allée Pierre Ziller
Business Pôle – Bâtiment B, Entrée A
06560 Sophia-Antipolis, France
Phone: + 33(0) 4 76 18 09 18

www.kalrayinc.com