

Dataflow Language Compilation for a Single Chip Massively Parallel Processor

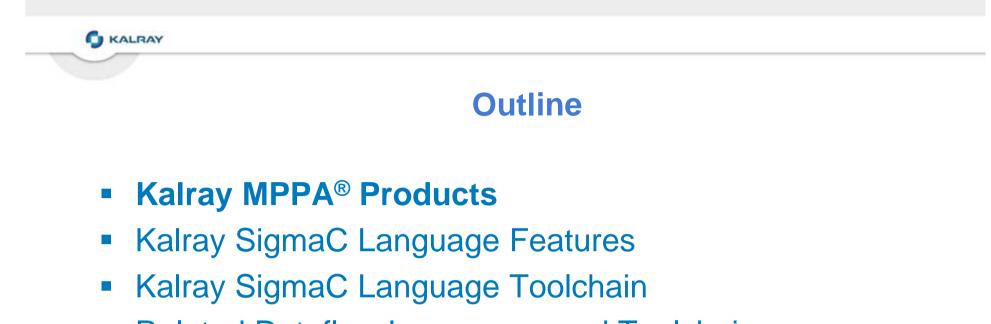
MuCoCoS 2013

September 7, 2013, Edinburgh, Scotland, UK

Benoît Dupont de Dinechin



www.kalray.eu



- Related Dataflow Languages and Toolchains
- Kalray MPPA[®] Application Examples
- Conclusions

Kalray MPPA® Products



High performance, low power singlechip massively parallel processors





C/C++ based Software Development Kit (SDK) for massively parallel programing





Development platform Reference Design Board



MuCoCoS 2013



Kalray MPPA®-256 Processor with CMOS 28nm TSMC

256 VLIW processing engine cores + 32 VLIW resource management cores



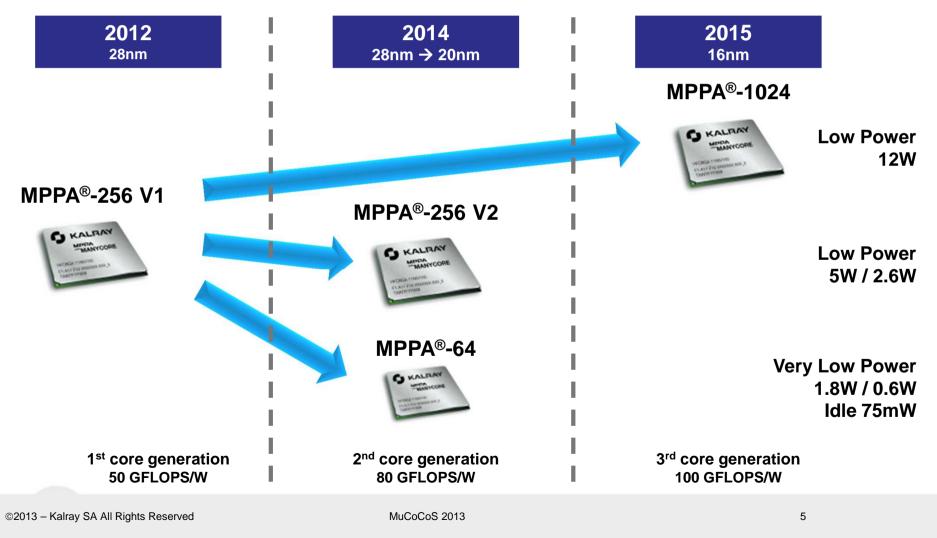
Available since November 2012

- High processing performance 700 GOPS – 230 GFLOPS SP
- Low power consumption
- High execution predictability
- High-level programming models
- PCI Gen3, Ethernet 10G, NoCX

MPPA MANYCORE Processor Roadmap

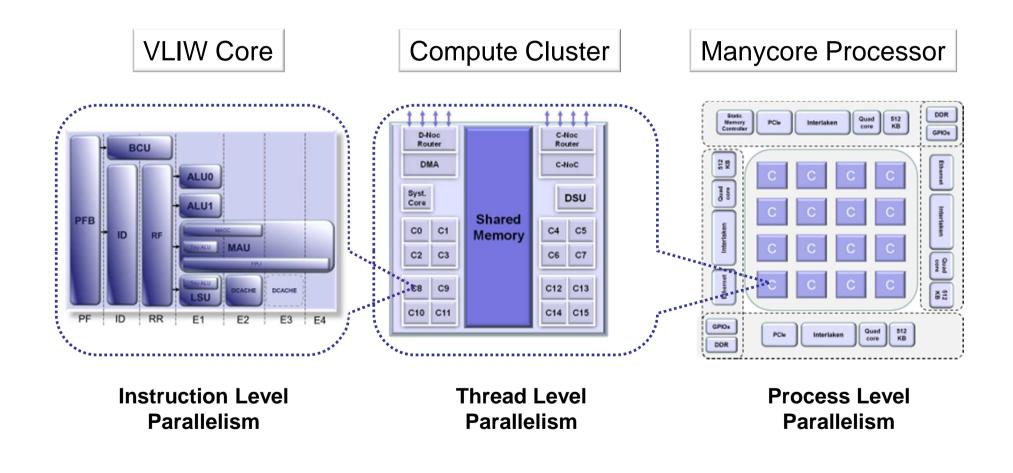
C KALRAY

Architecture scalability for high performances and low power



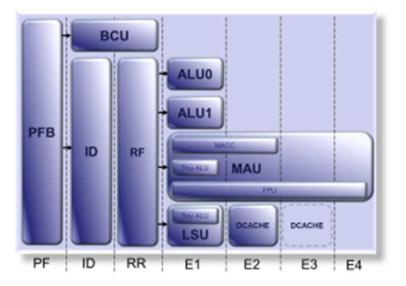


MPPA®-256 Processor Hierarchical Architecture





MPPA®-256 VLIW Core Architecture

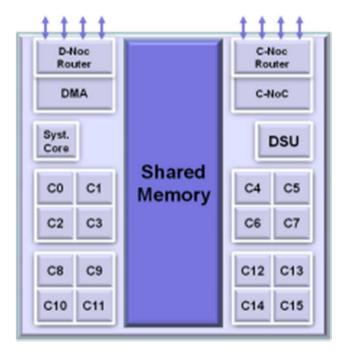


- 5-issue VLIW architecture
- Predictability & energy efficiency
- 32-bit/64-bit IEEE 754 FPU
- MMU for rich OS support

- Data processing code
 - Byte memory alignment
 - Standard & effective FPU
 - Configurable bitwise logic
 - Hardware looping
- System & control code
 - MMU → single memory port → no function unit clustering
- Execution predictability
 - Fully timing compositional core
 - LRU caches, low miss penalty
- Energy and area efficiency
 - 7-stage instruction pipeline, 400MHz
 - Idle modes and wake-up on interrupt



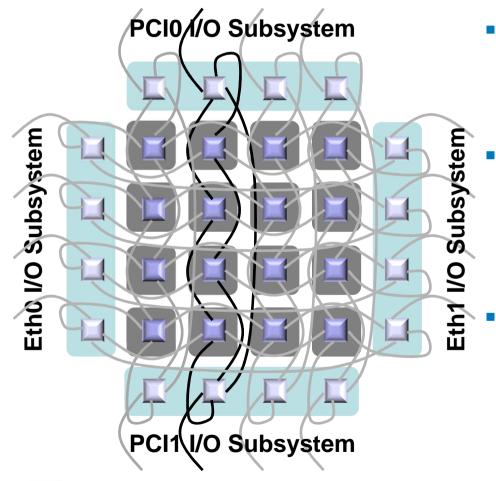
MPPA®-256 Compute Cluster



- 16 PE cores + 1 RM core
- NoC Tx and Rx interfaces
- Debug Support Unit (DSU)
- 2 MB of shared memory

- Multi-banked parallel memory
 - 38,4GB/s of bandwidth @400MHz
- Reliability
 - ECC in the shared memory
 - Parity check in the caches
 - Faulty cores can be switched off
- Predictability
 - Multi-banked shared memory with interleaved or blocked address map
- Low power
 - Memory banks with low power mode
 - Voltage scaling





- 20 memory address spaces
 - 16 compute clusters
 - 4 I/O subsystems with direct access to external DDR memory

MPPA

MANYCORE

- Dual Network-on-Chip (NoC)
 - Data NoC & Control NoC
 - Full duplex links, 4B/cycle
 - 2D torus topology + extension links
 - Unicast and multicast transfers
- Data NoC QoS
 - Flow control and routing at source
 - Guaranteed services by application of network calculus
 - Oblivious synchronization

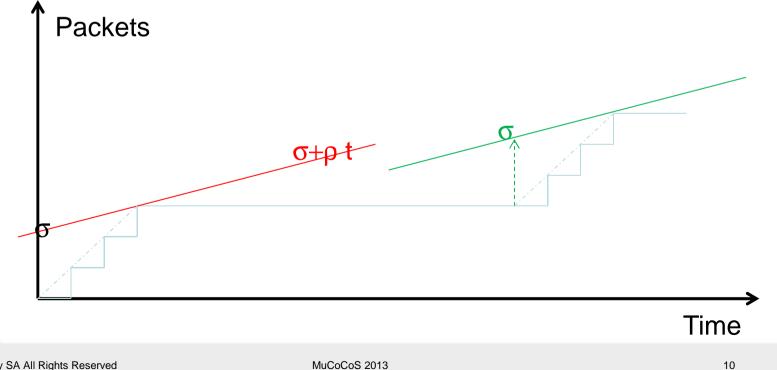
C KALRAY





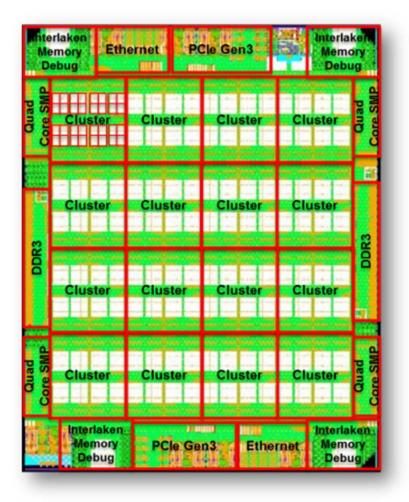
MPPA®-256 Data NoC Guaranteed Services

- Source traffic regulation using (σ, ρ)
 - A packet flow obeys (σ , ρ) if for any time interval τ the number of packets is not greater than $\sigma+\rho\tau$
 - The initial (σ , ρ) is set at the Tx Data NoC interface





MPPA®-256 Processor I/O Interfaces



- DDR3 Memory interfaces
- PCIe Gen3 interface
- 1G/10G/40G Ethernet interfaces
- SPI/I2C/UART interfaces
- Universal Static Memory Controller (NAND/NOR/SRAM)
- GPIOs with Direct NoC Access (DNA) mode
- NoC extension through Interlaken interface (NoC Express)

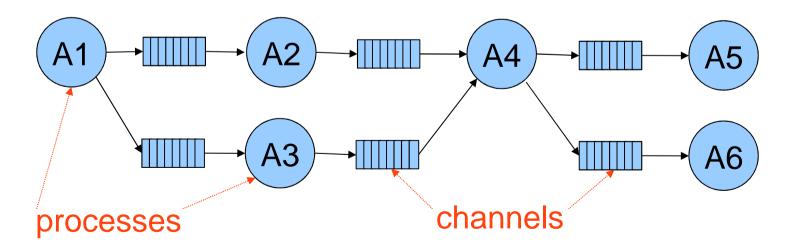
MPPA® Architecture Compared to other Manycores

- NVIDIA, ATI, ARM generalize the GPU architecture into GP-GPU
 - Streaming multiprocessors that share a cache and DDR memory
 - Each stream multiprocessor operates multi-threaded cores in SIMT
 - CUDA or OpenCL data parallel kernel programming models
- Cavium, Tilera TILE Gx, Intel MIC support shared coherent memory
 - Thread-based parallel programming (POSIX threads, OpenMP)
 - Non uniform memory access (NUMA) times, challenging cache design
- Kalray MPPA[®] extends the supercomputer clustered architecture
 - Clustered memory architecture scales to > 1M cores (BlueGene/Q)
 - Low energy per operation, high execution predictability
 - Stand-alone operation with I/O, low-latency processing



Dataflow Models of Computation

- Kahn Process Networks (KPN) [Kahn 1974]
 - Sequential "processes" connected through FIFO "channels"
 - Blocking "read", non blocking "write" on channels
 - Processes are also called "actors" or "agents"
 - Determinacy of results, independent of firing sequence



C KALRAY

Dataflow Models of Computation

- Dataflow Process Networks (DPN) [Lee & Parks 1995]
 - KPN with functional actor firing (no persistent agent state)
 - KPN with sequential firing rules (can be tested in a pre-defined order using only blocking reads)
- Synchronous Dataflow [Benveniste et al. 1994]
 - Clocks are associated with tokens carried by the channels
- Static Dataflow (SDF) [Lee & Messerschmitt 1987]
 - Agents producing and consuming a constant number of tokens
 - Single-rate SDF is known as Homogenous SDF (HSDF)
- Cyclo-Static Dataflow (CSDF) [Lauwereins 1994]
 - A cyclic state machine unconditionally advances at each firing
 - Known number of tokens produced and consumed for each state

C KALRAY



- Computation blocks and communication graph written in C
- Cyclostatic data production & consumption
- Firing thresholds of Karp & Miller
- Dynamic dataflow extensions
- Language called Sigma-C

Automatic mapping on MPPA[®] memory, computing, & communication resources

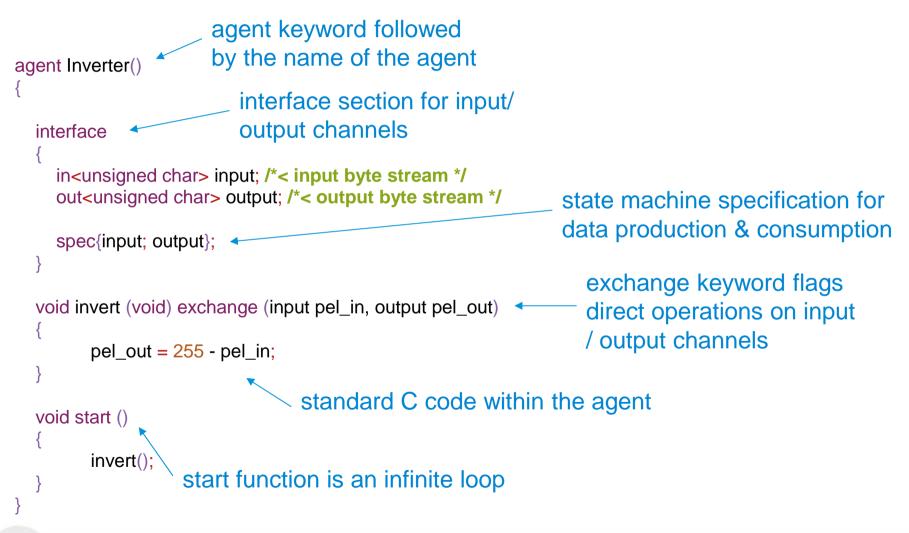
15A R5A R5A R5A



Outline

- Kalray MPPA[®] Products
- Kalray SigmaC Language Features
- Kalray SigmaC Language Toolchain
- Related Dataflow Languages and Toolchains
- Kalray MPPA[®] Application Examples
- Conclusions

Sigma-C Agent Example

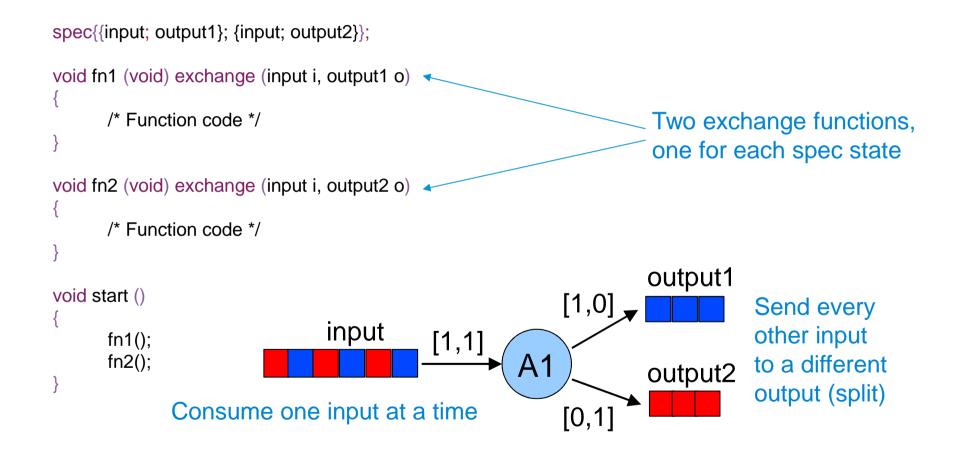


MuCoCoS 2013

Example of Cyclostatic Specs

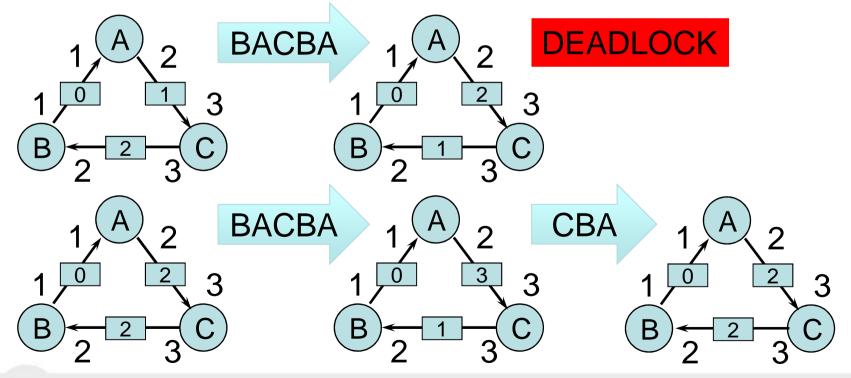
```
spec{(5){input}; {input; output};
void fn1 (void) exchange (input i)
                                                 5 input transitions before processing
                                                 another input and firing output
      /* Function code */
void fn2 (void) exchange (input i, output o)
      /* Function code */
void start ()
      int i,
                                                                         [0,1]
                                                        [5,1]
      for (i=0; i<5; i++) {
                                                                 A1
                                               input
                                                                                output
               fn1();
      fn2();
```

Example of Cyclostatic Specs



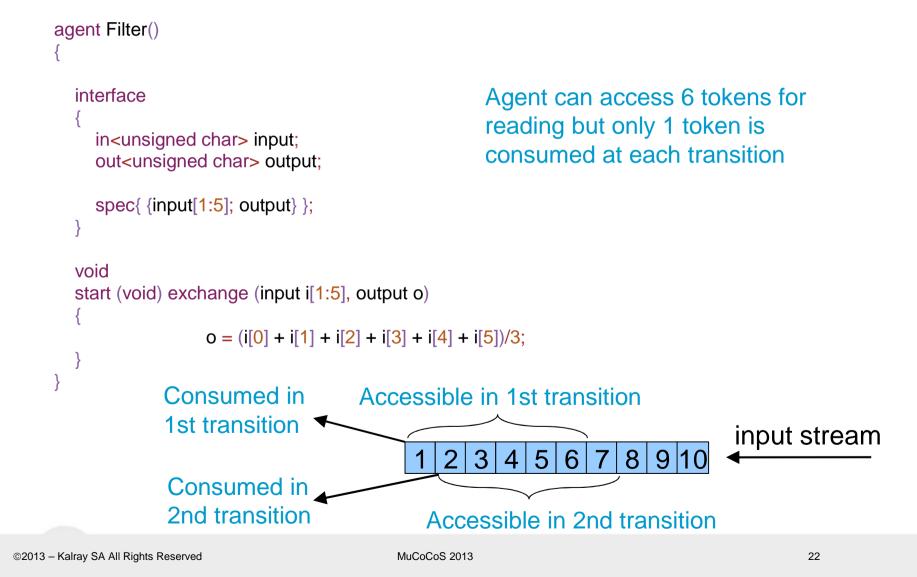


- At program startup, some channels may be non-empty
 - Required for the liveness of some dataflow graphs
 void preload(input_channel, int token_nbr, int data_size, void *input_data);



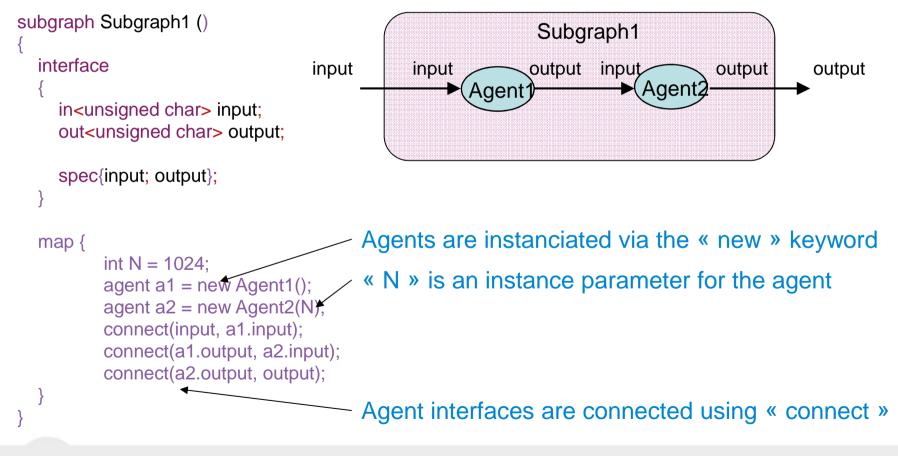
MuCoCoS 2013

Generalization of Karp & Miller Thresholds



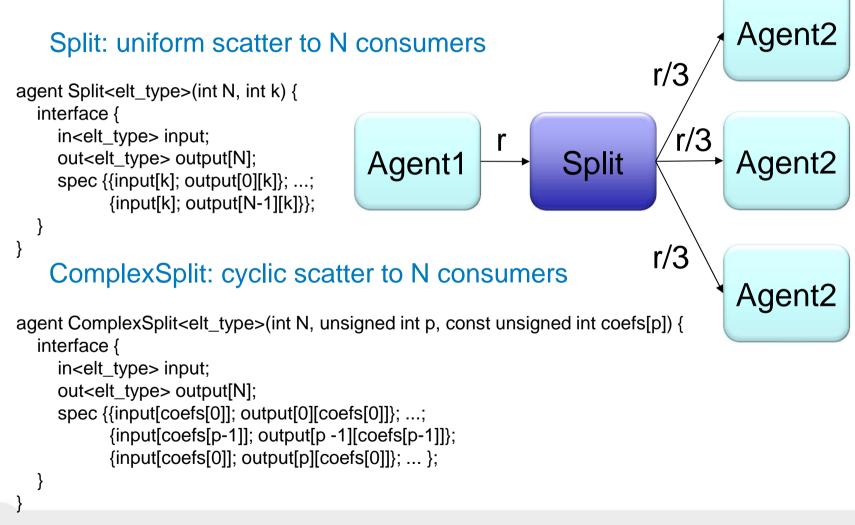
Instanciating and Connecting Agents

The « map » section of « subgraphs »



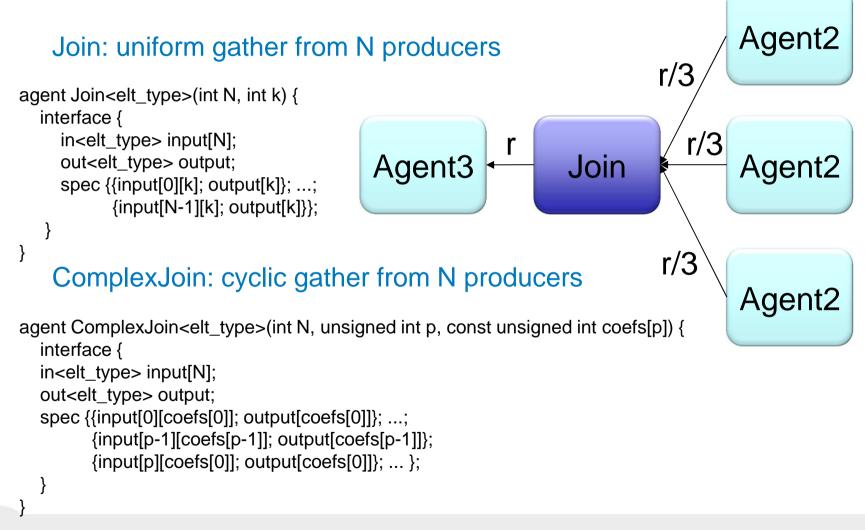
C KALRAY

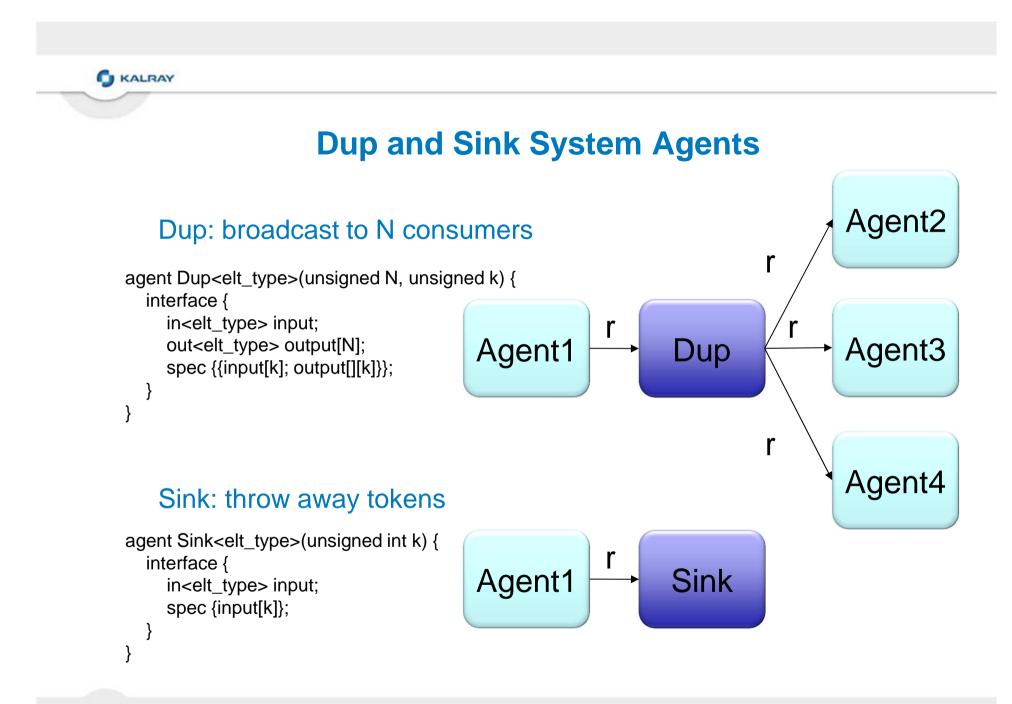
Split and ComplexSplit System Agents

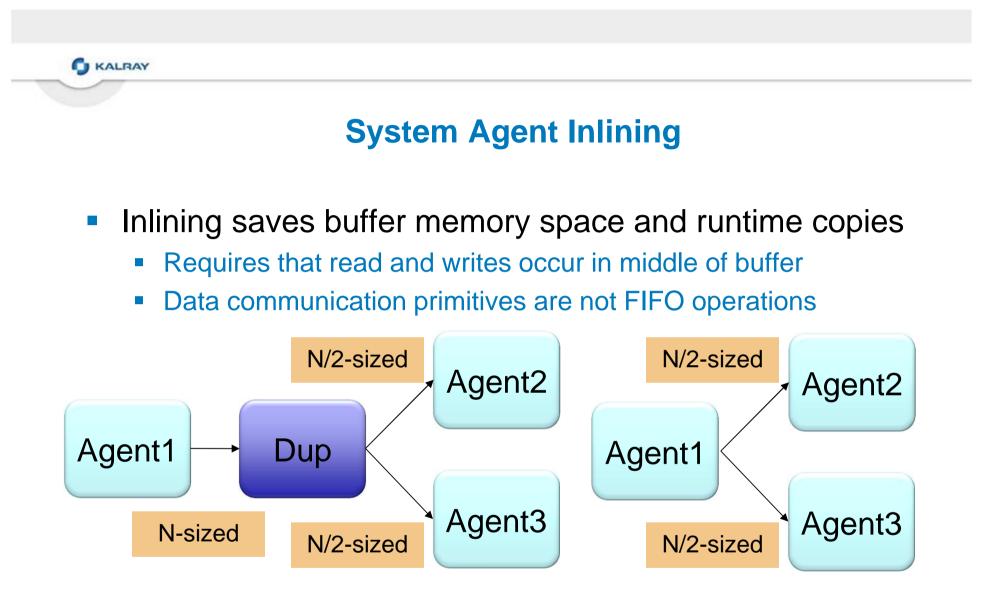


©2013 - Kalray SA All Rights Reserved

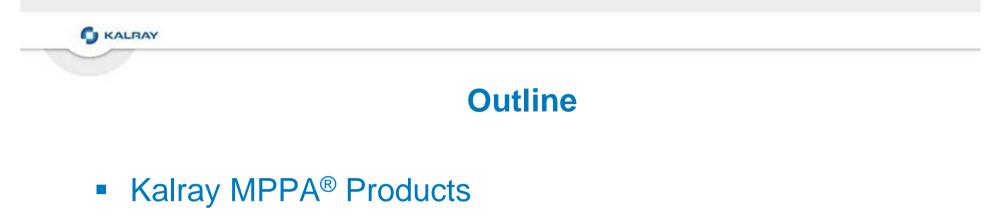
Join and ComplexJoin System Agents





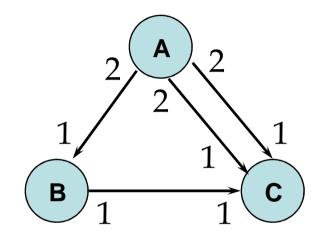


- Inlining constrainted by the 'pointer equivalence' principle
 - User code inside agents use regular pointers to access tokens



- Kalray SigmaC Language Features
- Kalray SigmaC Language Toolchain
- Related Dataflow Languages and Toolchains
- Kalray MPPA[®] Application Examples
- Conclusions

Static Dataflow Graph Boundedness



- Balance equations
 - 2 N(A) N(B) = 0
 - N(B) N(C) = 0
 - 2 N(A) N(C) = 0
 - 2 N(A) N(C) = 0

Graph incidence matrix

$$M = \begin{vmatrix} 2 & -1 & 0 \\ 0 & 1 & -1 \\ 2 & 0 & -1 \\ 2 & 0 & -1 \end{vmatrix}$$

- Must be non-full rank
 - Any multiple of the repetition vector N = |1 2 2|^T satisfies the balance equations
- Solution to balance equations ensures bounded execution

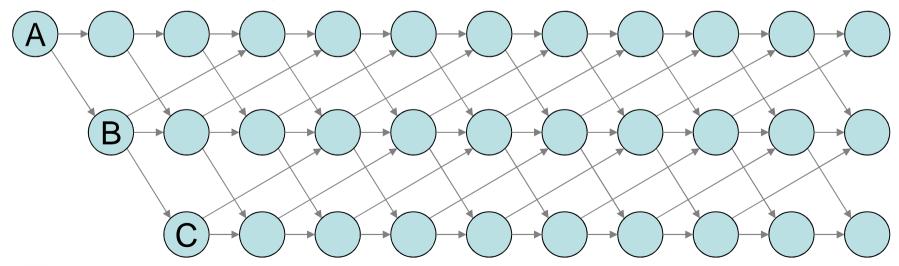
Sequencing Static Dataflow Graphs

- Symbolic execution of the dataflow graph
 - Execute one agent firing at a time
 - Find an 'hyperperiod', where each agent executes its number of times in the repetition vector and where the channel token count returns to the same values
 - Preloaded tokens in channels and firing thresholds may delay the first occurrence of the hyperperiod
- Symbolic execution of a balanced static dataflow graph always succeeds, unless the graph is not alive
 - Take advantage of choice over ready agent firing to heuristically optimize objectives such as maximum buffer use

C KALRAY

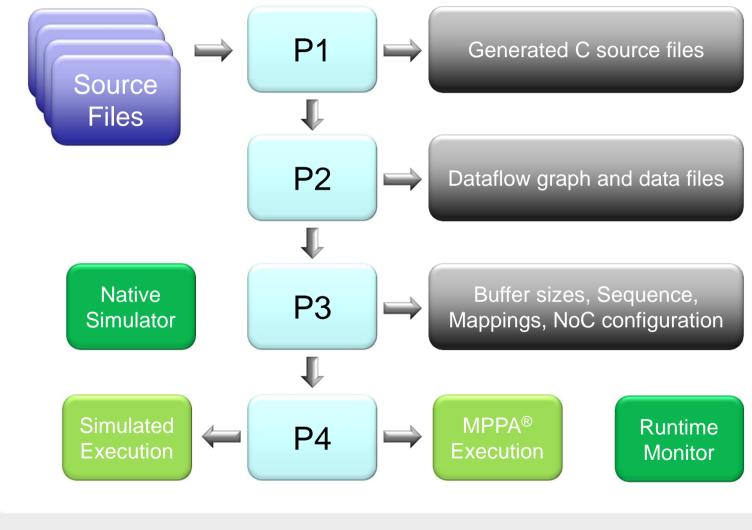
Dataflow Graph and Dependence Graph

- Static Dataflow graph execution can be interpreted
 A → 3 → B → 3 → C
- Efficient parallel execution is achieved by unfolding a dependence graph that ensures correct buffer accesses
 - True data dependence arcs and buffer size feedback arcs



C KALRAY

Dataflow Compilation and Execution Overview



©2013 - Kalray SA All Rights Reserved

MuCoCoS 2013

- Parse Sigma-C source files
 - Flex / Bison lexer-parser
 - Accept C99 + GNU C extensions
 - Resolve templating of agents
- C code generation
 - Generate code for instanciation of dataflow graph
 - Generate code for agents local data and functions
 - Leverage nested functions of GNU C
 - Insert buffer access macros in agent code
 - Allow late changes to buffer implementation



- Dataflow graph construction
 - Compile and execute map sections on toolchain host
- Dataflow graph coherency checks
 - Ensure there are no dangling ports
 - Check token structure compatibility between execution targets
- Produce intermediate representation
 - Flatten the dataflow graph
 - Compute channel initial tokens values
 - Resolve agent instance parameters to constants



- Balance equations
 - Find agent periods Ni (hyperperiod)
 - Replicate graph to consume initial tokens (k1-hyperperiod)
- System agent inlining analysis
 - Check that pointer equivalence is maintained
 - Compute minimum sizes of inlined buffers
- First symbolic execution
 - Compute minimum buffer sizes for liveness of dataflow graph
 - Build the generic data precedence graph
- (Advanced cyclostatic dataflow sizing and sequencing)

- Second symbolic execution
 - Compute k2-hyperperiod that activates the buffer feedback arcs
- Inlining of system agents
 - Resize the inlined buffers
- Pad buffers and insert shadow copy code
 - Maintain pointer equivalence with preloads and thresholds
- Mapping of tasks to platform resources
 - Use simulated annealing with placement constraints
 - Check effects on buffer sizes and inlining decisions
- Routing over NoC, PCIe and Ethernet
 - Compute routes and source flow restrictions



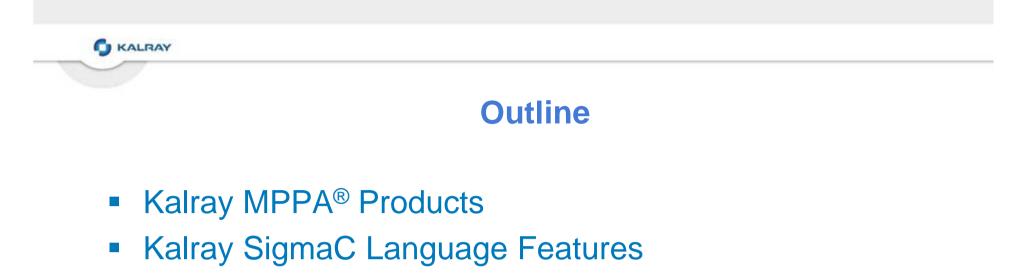
Phase 3

- Runtime generation
 - Compute buffer pointer increments
 - Generate dependency descriptors for runtime engine
 - Generate NoC configuration bit-stream
 - Compute FIFO sizes for inter-cluster dependency descriptors
- Dataflow graph rewriting
 - Map non-inlined system agents to DMA tasks
 - Coalesce inter-cluster transfers
 - Combine system agents
- Any dataflow graph rewriting restarts P3

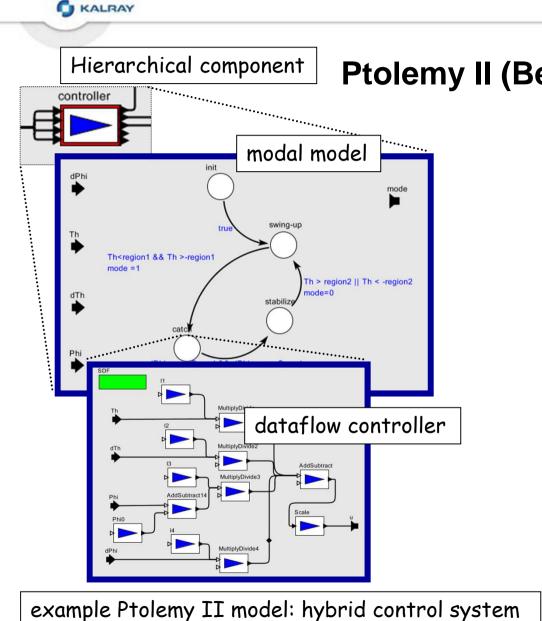


Sigma-C Toolchain Targets

- Native simulator
 - Self scheduled
 - Synchronised by channel read/write
 - Sequential
 - Only one agent active at a time for debug purposes
 - Sequenced
 - Synchronisation via a pre-computed partial order of P3
- Mixed simulator
 - Native simulation engine running on host
 - Agents compiled to VLIW core instruction set and run on ISS
- Multicores and manycore
 - X86_32, x86_64, MPPA platforms



- Kalray SigmaC Language Toolchain
- Related Dataflow Languages and Toolchains
- Kalray MPPA[®] Application Examples
- Conclusions



Ptolemy II (Berkeley)

Framework for experimentation with actor-oriented design, concurrent semantics, visual syntaxes, and hierarchical, heterogeneous design.

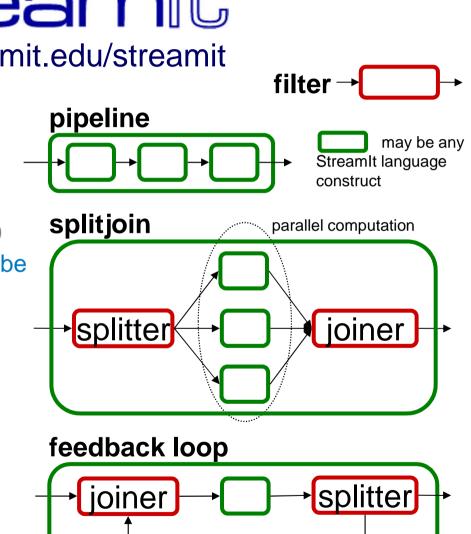


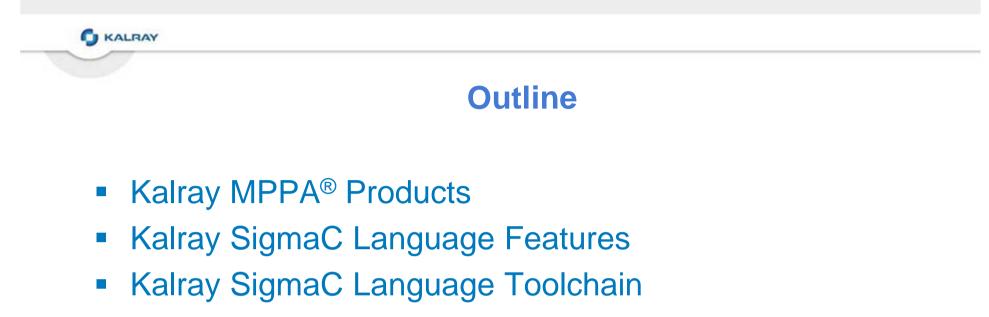
©2013 - Kalray SA All Rights Reserved

Streamt

http://cag.lcs.mit.edu/streamit

- Filters are autonomous unit of computation
 - No global resources
 - **FIFO** channels pop() /peek(index) /push(value)
 - Peek / pop / push rates must be constant
- Graph optimizations
 - Horizontal/vertical filter fusion/fission
 - Time/frequency domains
- **Teleport messaging**
- **Program morphing**
- RAW machine code generation





- Related Dataflow Languages and Toolchains
- Kalray MPPA[®] Application Examples
- Conclusions





MPPA®-256 PCIe Application Board AB01

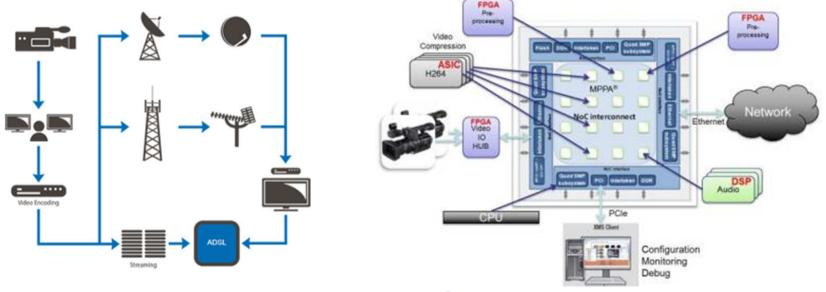
- Connect to the 4 I/O subsystems
 - 2 PCIe GEN3 x8 interfaces through a x16 PCIe switch
 - 2 DDR3 interfaces
 - 4 Ethernet interfaces (2x10G + 2x1G)
 - 4 Interlaken interfaces
 - NOR flash, GPIOs, leds, buttons, extensions & debug connectors





Video Broadcasting Demonstrator

- High definition H264 encoder on one MPPA[®]-256 processor
- System integration, lower power and cost
- Intel CPU + MPPA[®] implementation
- Flexibility & scalability



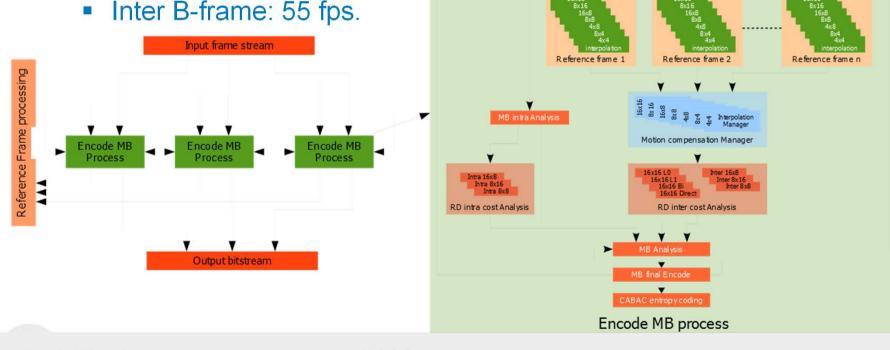
H264 encoder running on MPPA®-256 at less than 6W

©2013 - Kalray SA All Rights Reserved

MuCoCoS 2013

Dataflow H264 Encoder on the MPPA®-256 processor

- Better quality (SSIM and PSNR criteria) than C reference
 - Additional motion vectors and intra predictors tested (in parallel) without throughput impact.
- Intra I-frame: 110 fps.
- Inter P-frame: 40 fps.
- Inter B-frame: 55 fps.

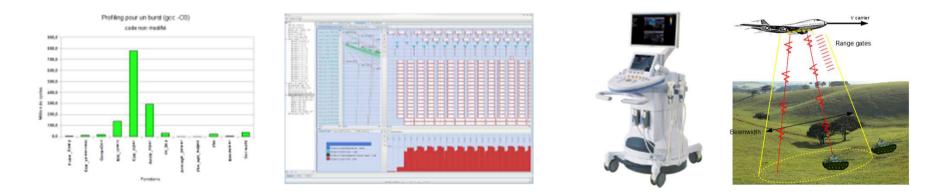


🕨 MB data Manager



Signal Processing Examples

- Radar applications: STAP, …
- Beam forming: Sonar, Echography
- Software Defined Radio (SDR)
- Dedicated libraries (FFT, FTFR, ...)

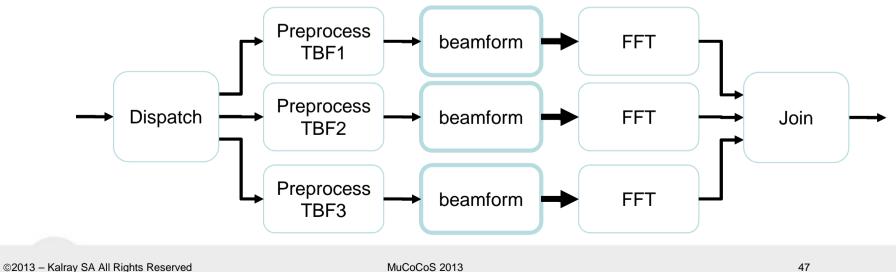


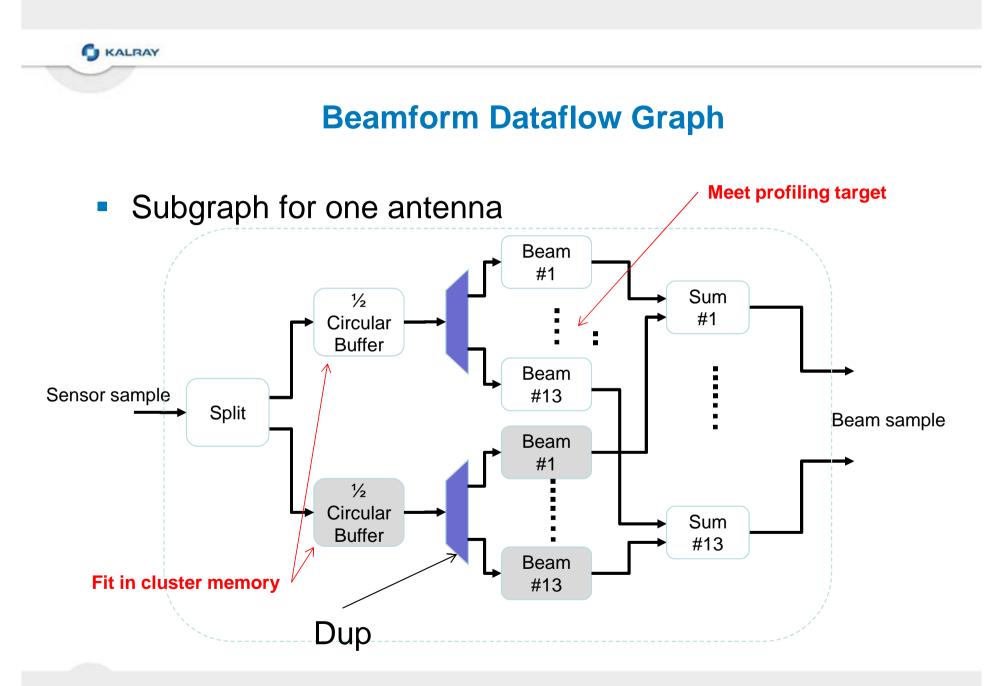
Well suited to massively parallel architectures Alternative of embedded DSP + FPGA platforms



Sonar Beam Forming

- Panoramic surveillance with a linear antenna
 - 3 very low frequency antennas sampled at 3840 Hz
 - [640, 1280 Hz]: 144 hydrophons
 - [320, 640 Hz] : 144 hydrophons
 - [160, 320 Hz] : 144 hydrophons
- Compute 180 beams per antenna



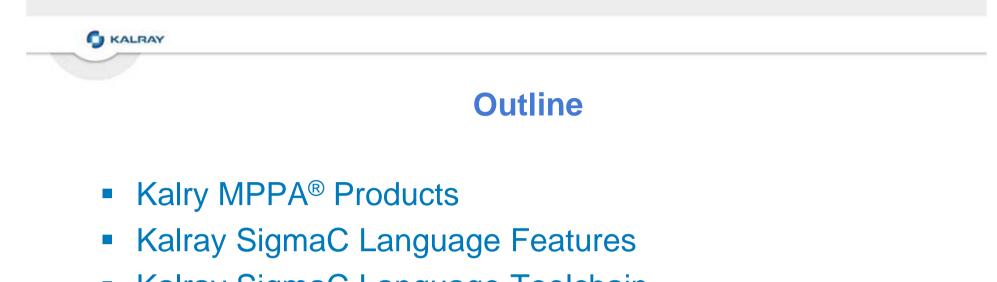


Application Performance Analysis Agent execution waveform

OP A 4	Context Selector Buffer Sizes Scheduling I	nteractive placer				
Control_1	Unlimited Resources View Limited Resources					
Identity_197 [MULTI INLINI Antenna 2			and the free states of the sta			
Dup_194	Cursors #1 #2 View 🖀 🔍 🔍	🕴 🕶 🚳 💎 Mode 👫 🛃	Extra (*)			
Identity_196	Gr Dup_72 [INUNED] REACHED	A1	4 14		2 0	10
Join_195 [MULTI INLINED]	Dup 73 (INUNED) REACHED		8	500	00	000
Processing_3 ⊢ ‴Dispatch_antenna_4	60°	=	ns sin	nis - Ins	ziri ziri	su
- Preprocess_tbf_7	Beamform_circular_tbf2_74 REAGHED		j		la se	
Preprocess_tbf_6	Beamform_circular_tbf2_75 REACHED					
Preprocess_tbf_8	Beamform select tbf 76 REACHED					
Beamform_tbf3_129						
Beamform_tbf2_69	Beamform_tbf2_sensor_77 REACHED	0 1 2 3 4 5 6 7	8 9 10 11 12 13 14 15 16 17 18 1	920212223242526 <mark>27</mark> 2829303132	2 33 34 35 36 37 38 39 40 41 42 4	3 44 45 46 4
← □Dup_71	Beamform_tbf2_sensor_78 REACHED	0 1 2 3 4 5 6 7	8 9 10 11 12 13 14 15 16 17 18 1	9 20 21 22 23 24 25 26 27 28 29 30 31 32	2 33 34 35 36 37 38 39 40 41 42 4	3 44 45 46 4
- Beamform_circular_t	Beamform adder tbf 79 REACHED					
🗢 🎩 Beamform_circular_t	Beamform_adder_tbf_80_REACHED					
← IDup_73 [INLINED]						
← □Dup_72 [INLINED] ← □Beamform_tbf2_sen	Beamform_tbf2_sensor_81 REACHED	0 1 2 3 4 5 6 7	8 9 10 11 12 13 14 15 16 17 18 1	920212223242526272829303132	2 33 34 35 36 37 38 39 40 41 42 4	3 44 45 46 4
← □Beamform_tbf2_sen	Beamform_tbf2_sensor_82 REACHED	0 1 2 3 4 5 6 7	8 9 10 11 12 13 14 15 16 17 18 1	920212223242526272829303132	2 33 34 35 36 37 38 39 40 41 42 4	3 44 45 46 4
• Beamform_tbf2_sen	Beamform adder tbf 83 REACHED		1-			
← IBeamform_tbf2_sen						
► ■Beamform_tbf2_sen						
• Beamform_tbf2_sen • Beamform_tbf2_sen						
~ Beamform_tbf2_sen		🗘 🗘 Parallelism				
∽ ■Beamform_tbf2_sen		€ 130	k			
∽ ■Beamform_tbf2_sen	Selected Agent:	120 1 1	30	50	70	000
← ■Beamform_tbf2_sen	Beamform_tbf2_sensor_89	Q 120 00 mg	om m	m m	om m	Om
► ■Beamform_tbf2_sen ► ■Beamform_tbf2_sen	Constraints:		· · · · · · · · · · · · · · · · · · ·	м		
- Beamform_tbf2_sen		100				
- Beamform_tbf2_sen	Frequency constraint : Constraint non set Propagated frequency constraint : 3.40909090	90		and the second state of the second	And a state of the second second second	
- Beamform_tbf2_sen	Propagated frequency constraint : 3.40909090	80 Barbarbarbarbarbarbarbarbarbarbarbarbarba				
← IBeamform_tbf2_sen	Frequency:					
← IBeamform_tbf2_sen	No Constraints : 3.99Hz	7.9	terrar and the second			
- Beamform_tbf2_sen	Buffers size limited : 3.96Hz	60	(Internet and I and the second se			
∽ ■Beamform_tbf2_sen → ■Beamform_tbf2_sen	Limited Ressources : 3.96Hz	50				
- Beamform_tbf2_sen	Clusterized Ressources : 3.96Hz	40				
- Beamform_tbf2_sen						
← IBeamform_tbf2_sen		3.0				
∽ IBeamform_tbf2_sen	< · · · · · · · · · · · · · · · · · · ·	20				
← IBeamform_tbf2_sen						
• Beamform_adder_tb	Info #1@55.56ms	#2 @ 313.119ms	Delta 257.559ms	Para	lism 78.8	
irence info PARAM. SPEC.	PORTS					
alysed Occurence:				Agent a	effective	
10				лует с		Char
				paralle		Clear

Application Mapping Analysis

Lamma Dame Remon. 1000 Lamma Remon. 1000 Additional biology Data 150 Data 150 <t< th=""><th>DP ACCONTROL 1</th><th>Context Selector Buffer Sizes Scheduling Interactive placer Mapped Buffer Sizes Place&Route</th></t<>	DP ACCONTROL 1	Context Selector Buffer Sizes Scheduling Interactive placer Mapped Buffer Sizes Place&Route
Autem.2 Du.34 Du.35 Displation Dis		Name RemaTotalBuffers stack code data 1 📰 😁 🏨 🚓 👁 💬 🗛
Dup_1si Hunu; 18 Punces 		Workstation with MPF 8.1 GB 22.8 19.5 3.26 1.46 KB 120 b
Henury, 1946 Henury, 1946 Henury, 1946 Henury, 1946 Henury, 1947 Henury, 1947 He		← 🗂 Workstation №0 4.09 2.21 420 1.8 MB 0 bytes 0 bytes
June, 15: MULTI NUMED P Compute Culture 1: 112 113 11 114 213.0 0 brtes 12 br Dispath, antime 1: Dispath, antime		Compute Cluster N 38.3 1.4 MB 1.34 56 KB 0 bytes 0 bytes
Dispation, steins, 4 Basimorm, 400, and another and a stein and a st	Join_195 (MULTI INLINED)	
Type press tot 7 Prepress tot 7 Prep	Processing_3	
Proprocess. bit. 6 Proprocess. bit. 6 Proprocess. bit. 6 Proprocess. bit. 6 Balantom. bit. 129 Balantom. bit. 299 B	- IDispatch_antenna_4	
Bit autom, 107, 26 Baamform, 107, 26 Baamform, 107, 26 Bit Baamform, 107, 26 Baamform, 107, 26 Baamform, 107, 26 Bit Baamform, 107, 26 Baamform, 107, 26 Baamform, 107, 26 Bit Baamform, 107, 26 Baamform, 107, 26 Baamform, 107, 26 Bit Baamform, 107, 26 Baamform, 107, 26 Baamform, 107, 26 Bit Baamform, 107, 26 Baamform, 107, 26 Baamform, 107, 26 Bit Baamform, 107, 26 Baamform, 107, 26 Baamform, 107, 26 Bit Baamform, 107, 26 Baamform, 107, 26 Baamform, 107, 26 Bit Baamform, 107, 26 Baamform, 107, 26 Baamform, 107, 26 Bit Baamform, 107, 26 Baamform, 107, 26 Baamform, 107, 26 Bit Baamform, 107, 26 Baamform, 202 Control 100, 100, 100, 100, 100, 100, 100, 100	Preprocess_tbf_7	
Elstanform. 107: 3.12 Baamform. 2012 43042612480 bytes 0 bytes Baamform. 107: 4 Baamform. 107: 4 4304260 bytes 0 bytes Baamform. 107: 4 Baamform. 107: 4 4304260 bytes 0 bytes Baamform. 107: 4 Baamform. 107: 4 4304260 bytes 0 bytes Baamform. 107: 4 Baamform. 107: 4 4304260 bytes 0 bytes Baamform. 107: 4 Baamform. 107: 4 4304260 bytes 0 bytes Baamform. 107: 4 Baamform. 107: 4 4304260 bytes 0 bytes Baamform. 107: 4 Baamform. 107: 4 4304260 bytes 0 bytes Baamform. 107: 4 Baamform. 107: 4 4304260 bytes 0 bytes Baamform. 107: 4 Baamform. 107: 4 4304280 bytes 0 bytes Baamform. 107: 4 Baamform. 107: 4 4304280 bytes 0 bytes Baamform. 107: 4 Baamform. 107: 4 4304280 bytes 0 bytes Baamform. 107: 4 Baamform. 107: 4 4304280 bytes 0 bytes Baamform. 107: 4 Baamform. 107: 4 4304280 bytes 0 bytes Baamform. 107: 4 Baamform. 107: 4 Baamform. 107: 4 1000 bytes 0 bytes Baamform. 107: 4 Baamform. 107: 4 <t< th=""><th>► ■ Preprocess_tbf_6</th><th></th></t<>	► ■ Preprocess_tbf_6	
200-071 Baardrom, tbr2, 499, 428, 1228, 0.0 bytes 0 bytes Baardrom, tbr2, 499, 428, 1228, 0.0 bytes 0 bytes Baardrom, tbr2, 499, 428, 1228, 0.0 bytes 0 bytes Baardrom, tbr2, 499, 428, 1228, 0.0 bytes 0 bytes Baardrom, tbr2, 499, 428, 1228, 0.0 bytes 0 bytes Baardrom, tbr2, 499, 428, 1228, 0.0 bytes 0 bytes Baardrom, tbr2, 499, 428, 1228, 0.0 bytes 0 bytes Baardrom, tbr2, 499, 428, 1228, 0.0 bytes 0 bytes Baardrom, tbr2, 499, 428, 428, 0.1288, 0.0 bytes 0 bytes Baardrom, tbr2, 499, 428, 428, 0.1288, 0.0 bytes 0 bytes Baardrom, tbr2, 499, 428, 428, 0.1288, 0.0 bytes 0 bytes Baardrom, tbr2, 499, 428, 428, 0.1288, 0.0 bytes 0 bytes Baardrom, tbr2, 499, 428, 428, 0.1288, 0.0 bytes 0 bytes Baardrom, tbr2, 499, 428, 428, 0.0 bytes 0 bytes Baardrom, tbr2, 499, 428, 428, 0.0 bytes 0 bytes Baardrom, tbr2, 489, 418, 0.0 bytes 0 bytes <		
www.sec. Pows.7 Baardform, Ltd2 439		
• Due, 71 • Beamform, ft/2 -43, 418, 12.66		
 Beamform_drid 12, set Beamform_drid 2, set		
• Beamform_dtrict -4942612480 bytes 0 bytes • Dup, 72 [NNN8D] Beamform_tbf2 -4942612480 bytes 0 bytes • Dup, 72 [NNN8D] Beamform_tbf2 -4942612480 bytes 0 bytes • Beamform_tbf2 -4942612480 bytes 0 bytes Beamform_tbf2 -4942612480 bytes 0 bytes • Beamform_tbf2 -4942612480 bytes 0 bytes Beamform_tbf2 -494260 bytes 0 bytes • Beamform_tbf2 -4942612480 bytes 0 bytes Beamform_tbf2 -494260 bytes 0 bytes • Beamform_tbf2 -4942612480 bytes 0 bytes Beamform_add -21821880 bytes 0 bytes • Beamform_tbf2 -4942612480 bytes 0 bytes Beamform_add -21821880 bytes 0 bytes • Beamform_tbf2 -49486.0 bytes 0 bytes Beamform_add -2182182188486 bytes 0 bytes • Beamform_tbf2 -2182182188486 bytes 0 bytes Beamform_add -218218218118148480 bytes 0 bytes • Beamform_tbf2 -2182182181481		
 Dup, 72 (NUNED) Dup, 72 (NUNED) Beamform, tbf2 = 49:42612.680 bytes 0 bytes Beamform, tbf2 = 49:448.0 bytes 0 bytes 0 bytes Beamform, tbf2 = 49:448.0 bytes 0 bytes 0 bytes Beamform, tbf2 = 49:448.0 bytes 0 bytes 0 bytes Beamform, tbf2 = 49:448.0 bytes 0 bytes 0 bytes Beamform, tbf2 = 49:448.0 bytes 0 bytes 0 bytes Beamform, tbf2 = 49:448.0 bytes 0 bytes 0 bytes Beamform, tbf2 = 49:448.0 bytes 0 bytes 0 bytes Beamform, tbf2 = 49:448.0 bytes 0 bytes 0 bytes Beamform, tbf2 = 49:448.0 bytes 0 bytes 0 bytes Beamform, tbf2 = 49:448.0 bytes 0 bytes 0 bytes Beamform, tbf2 = 49:448.0 bytes 0 bytes 0 bytes Beamform, tbf2 = 49:448.0 bytes 0 bytes 0 bytes Beamform, tbf2 = 49:448.0 bytes 0 bytes 0 bytes Beamform, tbf2 = 49:448.0 bytes 0 bytes 0 bytes Beamform, tbf2 = 49:448.0 bytes 0 bytes 0 bytes Beamform, tbf2 = 49:448.0 bytes 0 bytes 0 bytes Beamform, tbf2 = 49:448.0 bytes 0 bytes 0 bytes Beamform, tbf2 = 49:448.0 bytes 0 bytes 0 bytes Beamform, tbf2 = 49:448.0 bytes 0 bytes 0 bytes Beamform, tbf2 = 49:448.0 bytes 0 bytes 0 bytes Compute Cluster 1 43:11.11.12125.0 bytes 12 by Compute Cluster 1 43:11.11.14148.0 bytes 0 bytes 0 bytes Compute Cluster 1 43:11.14.14.1448.0 bytes 0 bytes 0 bytes Compute C		
 Beamform, 10/2 see Beamform, 20/2 see Compute Cluster N 321. 113. 113. 12. 256 b. 0. bytes 10 bytes Compute Cluster N 322. 113. 113. 12. 256 b. 0. bytes 10 bytes Compute Cluster N 450. bytes 10 bytes Compute Cluster N 450. bytes 10 bytes Compute Cluster N 450		
 Beamform_bt/2_set Beamform_bt/		
Beamform. bt7. see Beamform. bt7.		
Examform_tbf2_sen Beamform_tbf2_sen Compute Cluster N 221 113 113 114 12.85 (No bres 0 bres 12 by: Compute Cluster N 221 113 113 114 12.85 (No bres 0 bres 12 by: Compute Cluster N 221 113 113 114 12.85 (No bres 0 bres 12 by: Compute Cluster N 221 113 113 114 12.85 (No bres 0 bres 12 by: Compute Cluster N 221 113 113 114 12.85 (No bres 0 bres 12 by: Compute Cluster N 232 113 113 114 12.85 (No bres 0 bres 12 by: Compute Cluster N 232 113 114 12.85 (No bres 0 bres 12 by: Compute Cluster N 232 113 114 12.85 (No bres 0 bres 12 by: Compute Cluster N 232 113 114 12.85 (No bres 0 bres 12 by: Compute Cluster N 232 113 114 12.85 (No bres 0 bres 12 by: Compute Cluster N 232 113 114 12.85 (No bres 0 bres 12 by: Compute Cluster N 232 113 114 12.85 (No bres 0 bres 12 by: Compute Cluster N 232 113 114 12.85 (No bres 0 bres 12 by: Compute Cluster N 232 113 114 12.85 (No bres 0 bres 12 by: Compute C		Beamform_tbf2 431 418 12.68 0 bytes 0 bytes
becamform. tb72.sen becamform. tb72.sen becamf		Beamform_tbf2 _ 431 418 12.68 0 bytes 0 bytes
 Beamform. 107.2 sen Compute Clutater N 221 11.3n. 11.2 256 b 0 bytes 12 br Compute Clutater N 221 11.3n. 11.2 256 b 0 bytes 12 br Compute Clutater N 221 11.3n. 11.2 256 b 0 bytes 12 br Compute Clutater N 221 11.3n. 11.2 2448.0 bytes 0 bytes Compute Clutater N 221 11.3n. 11.2 2448.0 bytes 0 bytes Compute Clutater N 231 11.3n. 11.2 2448.0 bytes 0 bytes Compute Clutater N 231 11.3n. 11.2 2448.0 bytes 0 bytes Compute Clutater N 231 11.3n. 11.2 2448.0 bytes 0 bytes Compute Clutater N 231 11.3n. 11.2 2448.0 bytes 0 bytes Compute Clutater N 231 11.4n. 141 448.0 bytes 0 bytes Compute Clutater N 231 11.3n. 11.2 2448.0 bytes 0 bytes Compute Clutater N 231 11.3n. 11.2 2448.0 bytes 0 bytes Compute Clutater N 231 11.3n. 11.2 2448.0 bytes 0 bytes Compute Clutater N 231 11.3n. 11.2 2448.0 bytes 0 bytes Compute Clutater N 231 11.3n. 11.4 2490 0 bytes 12 br Compute Clutater N 231 11.3n. 11.4		
Beamform_1012_set Compute Cluster N 31 11.3. 4 K8 0 bytes 0 bytes B bytes Beamform_1012_set Compute Cluster N 321 11.3. 4 K8 0 bytes 0 bytes B bytes Compute Cluster N 321 11.3. Ket M S Dytes 0 bytes Beamform_1012_set Beamform_1012_set Compute Cluster N 31 11.3. 4 K8 0 bytes 0 bytes Compute Cluster N 321 11.3. 1 K8 0 bytes 0 bytes Compute Cluster N 321 11.3. 1 K8 0 bytes 0 bytes Compute Cluster N 33 11.3. 1 K8 0 bytes 0 bytes Compute Cluster N 33 11.3. 1 K8 0 bytes 0 bytes Compute Cluster N 33 11.4. 1 K8 0 bytes 0 bytes Compute Cluster N 33 11.4. 1 K8 0 bytes 0 bytes Compute Cluster N 33 11.4. 1 K8 0 bytes 0 bytes Compute Cluster N 33 11.4. 1 K8 0 bytes 0 bytes Compute Cluster N 33 11.4. 1 K8 0 bytes 0 bytes Compute Cluster N 33 11.4. 1 K8 0 bytes 0 bytes Compute Cluster N 31 13 12 100 K8 0 bytes 0 bytes Compute Cluster N 31 13 12 100 K8 0 bytes 0 bytes Compute Cluster N 31 13 12.		
Beamform_ubf2_set Beamform_ubf2	• Beamform_tbf2_sen	
Beamform, Lb72, sei Beamform_add 32 K8 28 K8 4 K8 0 bytes 0 bytes Beamform, Lb72, sei Beamform_add 32 K8 28 K8 4 K8 0 bytes 0 bytes Beamform, Lb72, sei Beamform_add 16 K8 12 K8 4 K8 0 bytes 0 bytes Beamform, Lb72, sei Beamform_add 16 K8 12 K8 4 K8 0 bytes 0 bytes Beamform, Lb72, sei Beamform_add 16 K8 12 K8 4 K8 0 bytes 0 bytes Beamform, Lb72, sei Compute Cluster N 5312. 1.1.1.1.1.1.1.2. 4 K8 0 bytes 0 bytes Beamform, Lb72, sei Compute Cluster N 532. 1.1.1.1.2. 1.4 K8 0 bytes 0 bytes Beamform, Lb72, sei Compute Cluster N 532. 1.2.2. 1.4 K8 0 bytes 0 bytes Beamform, Lb72, sei Compute Cluster N 233. 1.1.2. 1.4 K8 0 bytes 0 bytes Beamform, Lb72, sei Compute Cluster N 233. 1.1.2. 1.4 K8 0 bytes 0 bytes Beamform, Lb72, sei Compute Cluster N 233. 1.1.9. 2.4 K8 0 bytes 0 bytes Beamform, Lb72, sei Compute Cluster N 233. 1.2.1.1.9. 2.4 K8 0 bytes 0 bytes Beamform, Lb72, sei Compute Cluster N 234. 1.2.8. 1.4 K8 0 bytes 0 bytes Beamform, Lb72, sei Compute Cluster N 343.	∽ IBeamform_tbf2_sen	
Beamform_bt/2_sen Compute Cluster h 228 1.28 1.48 1.24 256 0.0 bytes 12 by Compute Cluster h 228 1.22 1.24 B 16.25 0 bytes 0 bytes Beamform_bt/2_sen Compute Cluster h 228 1.22 1.24 B 16.25 0 bytes 0 bytes Beamform_bt/2_sen Compute Cluster h 228 1.22 1.24 B 16.25 0 bytes 0 bytes Compute Cluster h 237 1.11 1.24 248 0 bytes 0 bytes Compute Cluster h 237 1.11 1.24 248 0 bytes 0 bytes Compute Cluster h 237 1.21 1.24 248 0 bytes 0 bytes Compute Cluster h 237 1.21 1.24 248 0 bytes 0 bytes Compute Cluster h 237 1.21 1.24 248 0 bytes 0 bytes Compute Cluster h 237 1.21 1.24 248 0 bytes 0 bytes Compute Cluster h 237 1.21 1.24 248 0 bytes 0 bytes Compute Cluster h 34 1.41 448 0 bytes 0 bytes Compute Cluster h 34 1.41 1.44 480 0 bytes 0 bytes Compute Cluster h 34 1.41 1.44 448 0 bytes 0 bytes Compute Cluster h 31 1.14 249 0 bytes 0 bytes Compute Cluster h 34 1.41 448 0 bytes 0 bytes Compute Cluster h 34 1.41 440 0 bytes 0 bytes Compute Cluster h 34 1.41 440 0 bytes 0 bytes Compute Cluster h 34 1.41 440 0 bytes 0 bytes Compute Cluster h 34 1.41 440 0 bytes 0 bytes Com	⊶ IBeamform_tbf2_sen	
 Beamform. tbr2_sen Beamform. tbr2_sen Beamform. tbr2_sen Beamform. tbr2_sen Beamform. tbr2_sen Beamform. tbr2_sen Compute Cluster N 321 11.3 11.3 14.80 bytes 0 bytes Beamform. tbr2_sen Compute Cluster N 322 1.3 11.3 11.2 256 b 0 bytes 12 by Compute Cluster N 232 1.2 1.2 10. bytes 12 by Compute Cluster N 232 1.1.3 11.3 991 108.r 0 bytes 0 bytes Beamform. tbr2_sen Compute Cluster N 232 1.1.3 11.3 11.2 256 b 0 bytes 0 bytes Compute Cluster N 232 1.1.3 11.3 991 108.r 0 bytes 12 by Compute Cluster N 232 1.1.3 11.3 991 108.r 0 bytes 12 by Compute Cluster N 232 1.1.3 11.3 991 108.r 0 bytes 12 by Compute Cluster N 232 1.1.3 991 108.r 0 bytes 12 by Compute Cluster N 232 1.1.3 11.3 991 108.r 0 bytes 12 by Compute Cluster N 232 1.2 1.9 24KB 0 bytes 0 bytes Compute Cluster N 232 1.3 991 108.r 0 bytes 12 by Compute Cluster N 232 1.2 1.9 24KB 0 bytes 0 bytes Compute Cluster N 232 1.2 1.9 24KB 0 bytes 0 bytes Compute Cluster N 233 1.1.4 14.1 44 4480 bytes 0 bytes Compute Cluster N 343 14 14.90 0 bytes 12 by Compute Cluster N 343 14 14.90 0 bytes 12 by Compute Cluster N 343 14 14 14 4480 bytes 0 bytes Compute Cluster N 343 11.3 11.3 12.2 108. 10.0. bytes 12 by Compute Cluster N 343 11.3 11.2 248.0 bytes 0 bytes Compute Cluster N 343 11.4 14	∽ ≡Beamform_tbf2_sen	
Beamform_tbf2_sen Compute Cluster N 31.2.1.118124190 bytes 0 bytes Compute Cluster N 31.2.1.118124190 bytes 0 bytes Compute Cluster N 31.2.1.118124100 bytes 12 by Beamform_tbf2_sen Beamform_tbf2_s		
Beamform_Ubf2.sen Beamform_Ubf2		
Beamform_tbf2_sen Beamform_tbf2		
 Beamform, bt/2, see Compute Cluster N 521 113 112 4 K8 0 bytes 0 bytes Compute Cluster N 51.2 1.41 1.2 MB 217.0 0 bytes 12 by Compute Cluster N 228 1.23 1.2 MB 10.25 0 bytes 12 by Compute Cluster N 232 1.2 MB 10.25 0 bytes 12 by Compute Cluster N 232 1.2 MB 10.25 0 bytes 12 by Compute Cluster N 232 1.2 MB 10.25 0 bytes 12 by Compute Cluster N 232 1.2 MB 10.25 0 bytes 12 by Compute Cluster N 232 1.2 MB 10.25 0 bytes 12 by Compute Cluster N 232 1.2 MB 10.25 0 bytes 12 by Compute Cluster N 232 1.2 MB 10.25 0 bytes 12 by Compute Cluster N 237 1.2 MB 10.25 0 bytes 12 by Compute Cluster N 237 1.2 MB 10.25 0 bytes 12 by Compute Cluster N 237 1.2 MB 10.25 0 bytes 12 by Compute Cluster N 237 1.2 MB 10.25 0 bytes 12 by Compute Cluster N 237 1.2 MB 10.25 0 bytes 12 by Compute Cluster N 237 1.2 MB 10.25 0 bytes 12 by Compute Cluster N 237 1.2 MB 10.25 0 bytes 12 by Compute Cluster N 34.3 1.4 MB 0 bytes 0 bytes Compute Cluster N 34.3 1.4 MB 0 bytes 0 bytes Compute Cluster N 34.3 1.4 MB 0 bytes 0 bytes Compute Cluster N 34.3 1.4 MB 0 bytes 0 bytes Compute Cluster N 450 101 838		
Beamform_tbf2_set Compute Cluster N 5.12 1.41 1.2 M2 217.0 0 bytes 12 by Beamform_tbf2_set Compute Cluster N 328 1.13 1.12 256 b 0 bytes 12 by Beamform_tbf2_set Compute Cluster N 425 1.02 873 168.7 0 bytes 0 bytes Beamform_tbf2_set Compute Cluster N 427 1.13 91 168.7 0 bytes 0 bytes Beamform_tbf2_set Compute Cluster N 427 1.13 91 168.7 0 bytes 0 bytes Beamform_tbf2_set Compute Cluster N 427 1.19 118 12 1.19 24KB 0 bytes 0 bytes Beamform_tbf2_set Compute Cluster N 427 1.19 118 12 500 b 12 by Beamform_tbf2_set Compute Cluster N 43 1.14 141 4 KB 0 bytes 0 bytes Beamform_tbf2_set Compute Cluster N 451 1.19 188 12.25 500 b 12 by Beamform_tbf2_set Compute Cluster N 451 1.19 188 12.25 500 b 12 by Compute Cluster N 451 1.19 188 114 4480 bytes 0 bytes Defentive 2000000000000000000000000000000000000		
Beamform, tbf2_see Compute Cluster N 328 113 112 256 b 0 bytes 12 by Beamform, tbf2_see Compute Cluster N 228 1.2 1.2 873 168.7 0 bytes 0 bytes Beamform, tbf2_see Compute Cluster N 228 1.2 1.2 1.86.7 0 bytes 12 by Beamform, tbf2_see Compute Cluster N 228 1.2 1.9 MB 16.25 0 bytes 12 by Beamform, tbf2_see Compute Cluster N 207 1.13 12 1.9 MB 16.25 0 bytes 12 by Beamform, tbf2_see Compute Cluster N 207 1.2 1.9 MB 16.25 0 bytes 12 by Beamform, tbf2_see Compute Cluster N 237 1.21 1.19 24KB 0 bytes 0 bytes Beamform, tbf2_see Compute Cluster N 34.3 1.41 1.41 4 KB 0 bytes 0 bytes Beamform, tbf2_see Compute Cluster N 34.3 1.41 1.44 44 KB 0 bytes 0 bytes Beamform, tbf2_see Compute Cluster N 34.3 1.41 1.44 44 KB 0 bytes 0 bytes Compute Cluster N 31.2		
Beamform_tbf2_sen Beamform_tbf2_sen Compute Cluster N 228 1.22 1.2 M 168.7 0 bytes 0 bytes Compute Cluster N 207 11.3 991 168.7 0 bytes 0 bytes Compute Cluster N 207 11.3 991 168.7 0 bytes 0 bytes Compute Cluster N 207 11.3 991 168.7 0 bytes 0 bytes Compute Cluster N 207 11.3 991 168.7 0 bytes 0 bytes Compute Cluster N 207 11.3 991 168.7 0 bytes 0 bytes Compute Cluster N 207 11.3 991 168.7 0 bytes 0 bytes Compute Cluster N 207 11.3 991 168.7 0 bytes 0 bytes Compute Cluster N 207 11.3 991 168 200 0 bytes 12 by Compute Cluster N 207 11.3 991 168 120 100 bytes 0 bytes Compute Cluster N 506 876 794 KB 82 KB 0 bytes 0 bytes Compute Cluster N 506 876 794 KB 82 KB 0 bytes 0 bytes Compute Cluster N 450 114 440 0 bytes 12 by Compute Cluster N 450 114 2490 0 bytes 12 by Compute Cluster N 450 110 838 176.7 0 bytes 0 bytes Compute Cluster N 450 101 838 176.7 0 bytes 0 bytes Compute Cluster N 450 101 838 176.7 0 bytes 0 bytes Compute Cluster N 450 100 KB 0 bytes 0 bytes Compute Cluster N 81 1.36 1.27 100 KB 0 bytes 0 bytes Compute Cluster N 81 1.36 1.27 100 KB 0 bytes 0 bytes Compute Cluster N 81 1.36 1.27 100 KB 0 bytes 0 bytes Compute Cluster N 81 1.36 1.27 100 KB 0 bytes 0 bytes Compute Cluster N 81 1.36 1.27 100 KB 0 bytes 0 bytes Compute Cluster N 81 1.36 1.27 100 KB 0 bytes 0 bytes Compute Cluster N 81 1.36 1.27 100 KB 0 bytes 0 bytes Compute Cluster N 81 1.36 1.27 100 KB 0 bytes 0 bytes Compute Cluster N 81 1.36 1.27 100 KB 0 bytes 0 bytes Compute Cluster N 81 1.36 1.27 100 KB 0 bytes 0 bytes Compute Cluster N 81 1.36 1.27 100 KB 0 bytes 0 bytes Compute Cluster N 81 1.36 1.27 100 KB 0 bytes 0 bytes Compute Cluster N 81 1.36 1.27 100 KB 0 bytes 0 bytes Compu		A CE Compute Cluster N 228 112 112 256 h O hyter 12 hu
Beamform_tbf2_sen Compute Cluster N 228 1.22 1.2 MB 16.25 0 bytes 12 by Beamform_tbf2_sen Compute Cluster N 37 1.13 991 168 0 bytes 0 bytes Beamform_tbf2_sen Compute Cluster N 237 1.21 1.19 24 KB 0 bytes 0 bytes Beamform_tbf2_sen Compute Cluster N 237 1.21 1.19 24 KB 0 bytes 0 bytes Beamform_tbf2_sen Compute Cluster N 237 1.21 1.19 24 KB 0 bytes 0 bytes Beamform_tbf2_sen Compute Cluster N 34.3 1.41 1.41 4 KB 0 bytes 0 bytes Compute Cluster N 34.3 1.41 1.41 4 KB 0 bytes 0 bytes Compute Cluster N 34.3 1.44 1.41 4 KB 0 bytes 0 bytes Beamform_tbf2_sen Compute Cluster N 34.3 1.41 1.41 4 KB 0 bytes 0 bytes Documents Cluster N 50 704 KB 32 KB 0 bytes 0 bytes Beamform_tbf2_sen Compute Cluster N 51 1.38 1.14 249.0 0 bytes 12 by Documents Cluster N 51 1.38 1.14 249.0 0 bytes 0 bytes Beamform_tbf2_sen Compute Cluster N 51 1.38 1.14 249.0 0 bytes 0 bytes Documents Cluster N 51 1.38 1.27 1.00 bytes 0 bytes Compute Cluster N 51 1.1 1.27 1.00 bytes 0 bytes Compute Cluster N 540 101 838 1.76 0 bytes 0 bytes Documents Cluster N 540 0 bytes 0 bytes Compute Cluster N 550 101 838 1.27 1.00 bytes 0 bytes Compute Cluster N 540 0 bytes 0 bytes Documents Cluster N 540 0 bytes 0 bytes		
Beamform_tbf2_sen Beamform_tbf2		
Beamform_tbf2_sen Compute Cluster N 237 1.21 1.19 24 KB 0 bytes 0 bytes Beamform_tbf2_sen Compute Cluster N 24 1.19 1.18 1.225 500 b 12 b Beamform_tbf2_sen Compute Cluster N 3 1.14 4. KB 0 bytes 0 bytes Compute Cluster N 3 1.14 4 4. KB 0 bytes 0 bytes Beamform_tbf2_sen Compute Cluster N 3 1.14 4 4. KB 0 bytes 0 bytes Compute Cluster N 31 1.138 1.14 249.0 0 bytes 12 by Beamform_tbf2_sen Compute Cluster N 31 1.138 1.14 249.0 0 bytes 0 bytes Compute Cluster N 31 1.38 1.14 249.0 0 bytes 0 bytes Compute Cluster N 450 101 838 1.76 0 bytes 0 bytes Compute Cluster N 450 101 838 1.77 0 bytes 0 bytes Compute Cluster N 450 101 838 1.77 0 bytes 0 bytes Compute Cluster N 450 101 0 bytes 0 bytes Compute Cluster N 450 101 0 bytes 0 bytes Compute Cluster N 450 101 0 bytes 0 bytes Compute Cluster N 450 101 0 bytes 0 bytes Compute Cluster N 450 101 0 bytes 0 bytes Compute Cluster N 450 101 0 bytes 0 bytes Compute Cluster N 450 0 byt		
Beamform_tbf2_sen Compute Cluster N 261 1.19 1.18 12.25500 b 12 by Beamform_tbf2_sen Compute Cluster N 34.3 1.41 1.41 4 KB 0 bytes 0 bytes Beamform_tbf2_sen Compute Cluster N 56 876 794 KB 32 KB 0 bytes 0 bytes Beamform_tbf2_sen Compute Cluster N 54 1.38 1.41 1.41 4 KB 0 bytes 0 bytes Beamform_tbf2_sen Compute Cluster N 54 1.38 1.41 249.0 0 bytes 10 bytes Beamform_tbf2_sen Compute Cluster N 54 1.38 1.14 249.0 0 bytes 10 bytes Beamform_tbf2_sen Compute Cluster N 54 1.38 1.14 249.0 0 bytes 10 bytes Compute Cluster N 54 1.12 1.38 1.14 249.0 0 bytes 10 bytes Identity 203 Beamform_tbf2_sen Compute Cluster N 54 1.27 100 bytes 0 bytes Identity 203 Compute Cluster N 54 1.27 100 bytes 0 bytes Compute Cluster N 54 0 bytes 0 bytes Identity 203 Beamform_tbf2_sen Compute Cluster N 450 101 838 1.27 100 bytes 0 bytes Identity 203 Compute Cluster N 450 0 bytes 0 bytes 0 bytes Compute Cluster N 450 0 bytes 0 bytes 0 bytes Identity 203 Beamform_tbf2_sen Compute Cluster N 450 0 bytes 0 bytes 0 bytes Identity 203 Identity 203 Compute Cluster N 450 0 bytes 0 bytes 0 bytes Compute Cluster N 450 0 bytes 0 bytes 0 bytes <th></th> <th>a CE Campute Chuster 5, 227 1, 21, 1, 10 24 KB O buter 0 buter 0</th>		a CE Campute Chuster 5, 227 1, 21, 1, 10 24 KB O buter 0 buter 0
Beamform_tbr2_sen Compute Cluster N 34.3 1.41 1.41 4 K8 0 bytes 0 bytes Compute Cluster N 34.3 1.41 1.41 4 K8 0 bytes 0 bytes Compute Cluster N 31.2 1.38 1.14 24 K8 0 bytes 0 bytes Compute Cluster N 31.2 1.38 1.14 24 bytes 0 bytes 1 bytes Beamform_tbr2_sen Compute Cluster N 31.2 1.38 1.14 24 bytes 0 bytes 0 bytes 0 bytes Compute Cluster N 450 101 838 176.7 0 bytes 0 bytes 0 bytes Compute Cluster N 450 101 838 176.7 0 bytes 0 bytes 0 bytes 0 bytes Compute Cluster N 450 101 838 176.7 100 K8 0 bytes 0 bytes 0 bytes Compute Cluster N 450 101 838 176 100 K8 0 bytes 0 bytes 0 bytes Compute Cluster N 450 101 838 176 100 K8 0 bytes 0 bytes 0 bytes Compute Cluster N 450 100 K8 0 bytes 0 bytes 0 bytes Compute Cluster N 450 100 K8 0 bytes 0 bytes 0 bytes Compute Cluster N 450 100 K8 0 bytes 0 bytes 0 bytes Compute Cluster N 450 100 K8 0 bytes 0 bytes 0 bytes Compute Cluster N 450 100 K8 0 bytes 0 bytes 0 bytes Compute Cluster N 450 100 K8 0 bytes 0 bytes 0 bytes Compute Cluster N 450 100 K8 0 bytes 0 bytes 0 bytes Compute Cluster N 450 100 K8 0 bytes 0 bytes 0 bytes Compute Cluster N 450 100 K8 0 bytes 0 bytes 0 bytes Compute Cluster N 450 100 K8 0 bytes 0 bytes 0 bytes Compute Cluster N 450 100 K8 0 bytes 0 bytes 0 bytes Compute Cluster N 450 100 K8 0 bytes 0 bytes Compute Cluster N 450 100 K8 0 bytes 0 bytes 0 bytes Compute Cluster N 450 100 K8 0 bytes 0 bytes 0 bytes Compute Cluster N 450 100 K8 0 bytes 0 bytes Compute Cluster N 450 100 K8 0 bytes 0 bytes Compute Clu		← Compute Cluster N 261 1.19 1.18 12.25500 b 12 by
Beamform_tbf2_sen		C Compute Cluster N 34.3 1.41 1.41 4 KB 0 bytes 0 bytes
Compute Cluster N 31.2 1.38 1.14 249.0 0 bytes 12 by Beamform_tbf2_sen Compute Cluster N 450 101 838 176.7 0 bytes 0 bytes Compute Cluster N 450 101 838 176.7 0 bytes 0 bytes Compute Cluster N 450 101 838 176.7 0 bytes 0 bytes Compute Cluster N 450 101 838 176.7 0 bytes 0 bytes Compute Cluster N 450 101 838 176.7 0 bytes 0 bytes Compute Cluster N 450 101 838 176.7 0 bytes 0 bytes Compute Cluster N 450 101 838 176 0 bytes 0 bytes Compute Cluster N 450 101 838 176 127 100 bytes 0 bytes Compute Cluster N 450 101 838 176 127 100 bytes 0 bytes Compute Cluster N 450 101 136 127 100 bytes 0 bytes Compute Cluster N 450 101 127 100 bytes 0 bytes 0 bytes Compute Cluster N 450 127 100 bytes 0 bytes 0 bytes Compute Cluster N 450 127 100 bytes 0 bytes 0 bytes Compute Cluster N 450 1450.		C Compute Cluster N 596 876 794 KB 82 KB 0 bytes 0 bytes
Beamform_tbf2_sen Compute Cluster N 450 101 838 1767 0 bytes 0 bytes Compute Cluster N 451 136 101 838 1767 0 bytes 0 bytes Compute Cluster N 451 136 127 100 bytes 0 bytes Compute Cluster N 451 136 127 100 bytes 0 bytes Compute Cluster N 451 136 127 100 bytes 0 bytes Compute Cluster N 451 136 127 100 bytes 0 bytes Compute Cluster N 451 136 127 100 bytes 0 bytes Compute Cluster N 451 136 127 100 bytes 0 bytes Compute Cluster N 451 136 127 100 bytes 0 bytes Compute Cluster N 451 136 127 100 bytes 0 bytes Compute Cluster N 451 136 127 100 bytes 0 bytes Compute Cluster N 451 136 127 100 bytes 0 bytes Compute Cluster N 451 136 127 100 bytes		
Beamform_adder_tb Compute Cluster N 81.1 1.36 1.27 100 KB 0 bytes 0 bytes 0 bytes Townsform_adder_tb Compute Cluster N 1.4E Ohv. 0 bytes 0 bytes 0 bytes 0 bytes 0 KAML SPEC. PORTS		Compute Cluster N 450 101 838 176.7 0 bytes 0 bytes
XAML SPEC. PORTS		
	- Basmfarm addar th	Campute Chuter & 1.4E Ohu Ohuter Ohuter Ohuter Ohuter Ohuter V
AM OF agent Beamform_tbf2_sensor_89	RAM. SPEC. PORTS	
AM OF agent Beamform_tbf2_sensor_89		
	AM OF agent Beamforn	1_tbf2_sensor_89



- Kalray SigmaC Language Toolchain
- Related Dataflow Languages and Toolchains
- Kalray MPPA[®] Application Examples
- Conclusions



Lessons Learned

- Kalray Dataflow well suited to key MPPA[®] applications
 - Cyclostatic dataflow especially effective on signal processing and video encoding (AVC/H264, HEVC/H265)
 - Other applications that deploy data-dependent computation graphs (such as LTE base station) are more difficult to express
- Static Dataflow allows to automate parallel execution on clustered manycore processors such as the MPPA[®]-256
 - Code and data distribution, communication over NoC
 - No need for specific architectural support in NoC and DMA
- Kalray Dataflow toolchain also enables parallel execution of single applications on hybrid target systems
 - Demonstrated Intel CPU + 2 MPPA[®] AB01 boards



Future Developments

- Extended Cyclostatic Dataflow Techniques
 - Based on work by A. Munier et O. Marchetti (U. Paris VI / LIP6) on Marked Weighted Timed Event Graphs (MTWEG)
 - <u>K-Periodic schedules for evaluating the maximum throughput of a</u> <u>Synchronous Dataflow graph</u>
 B. Bodin, A. Munier-Kordon, B. Dupont de Dinechin Embedded Computer Systems (SAMOS), 2012
 - <u>Liveness evaluation of a cyclo-static DataFlow graph</u>
 M. Benazouz, A. Munier-Kordon, T. Hujsa, B. Bodin
 Proceedings of the 50th Annual Design Automation Conference
 - Periodic Schedules for Cyclo-Static Dataflow Accepted at ESTIMedia 2013
- Time-Triggered source and sink nodes, RT extensions