POLITECNICO DI MILANO

The Shining embedded system design methodology based on self dynamic reconfigurable architectures

C. A. Curino, L. Fossati, F. Redaelli, M. D. Santambrogio, D. Sciuto
{curino,fossati,rana,santambr,sciuto}@elet.polimi.it, francesco.redaelli@dresd.org

Speaker: Luca Fossati

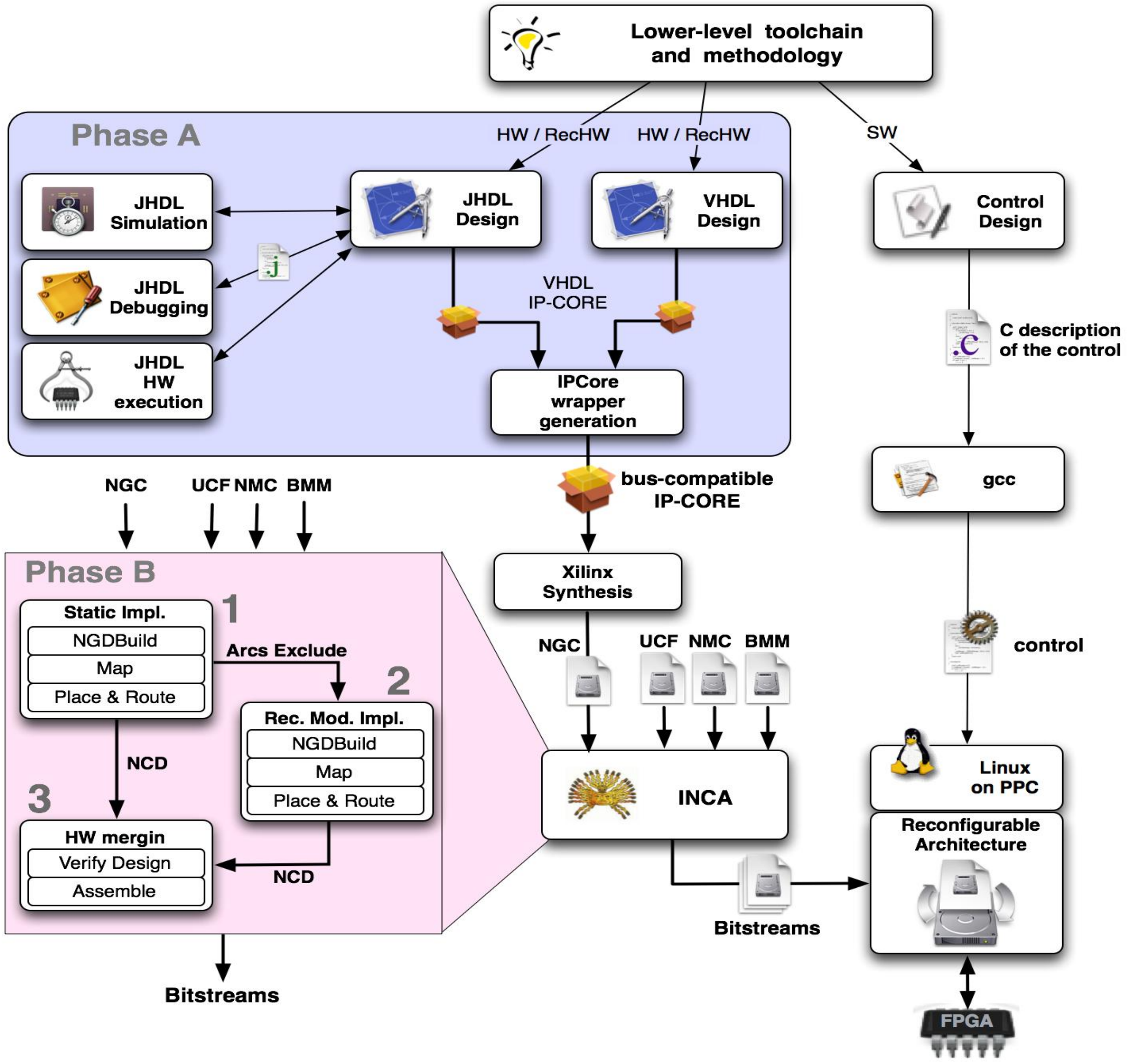
WWW.DRESD.ORG

Outline

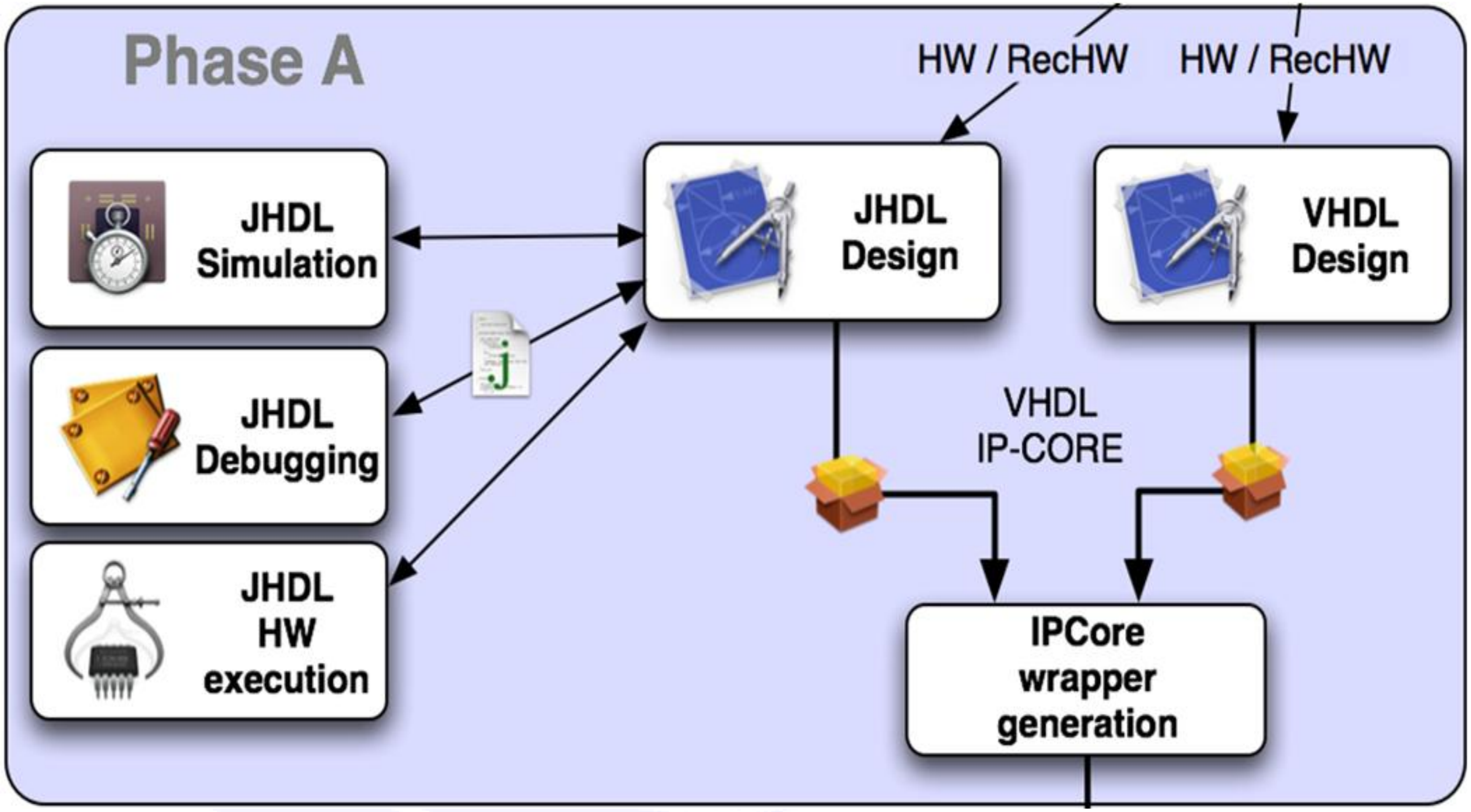
Motivations

- No generalized methodology allowing both the automatic derivation of a complete system solution able to fit into the final device, and mixed hardware-software solutions, exploiting partial reconfiguration capabilities
- Needs of a methodology that organizes the input specification of a complex System-on-Chip design into three different components: hardware, reconfigurable hardware and software, each handled by dedicated sub-flows

The Shining Methodology



Phase A

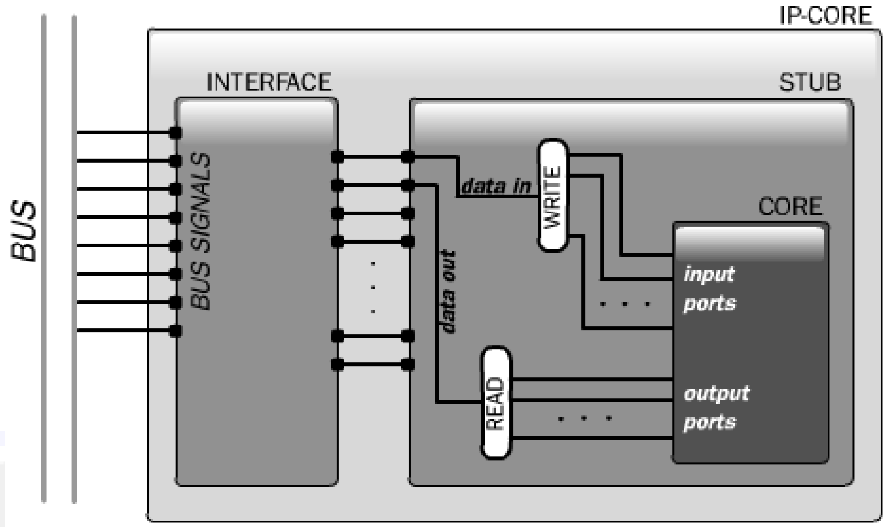

Phase A
HW / RecHW
HW / RecHW
JHDL Simulation
JHDL Design
VHDL Design
JHDL Debugging
JHDL HW execution
VHDL IP-CORE
IPCore wrapper generation

Phase A - Results

- JHDL

+

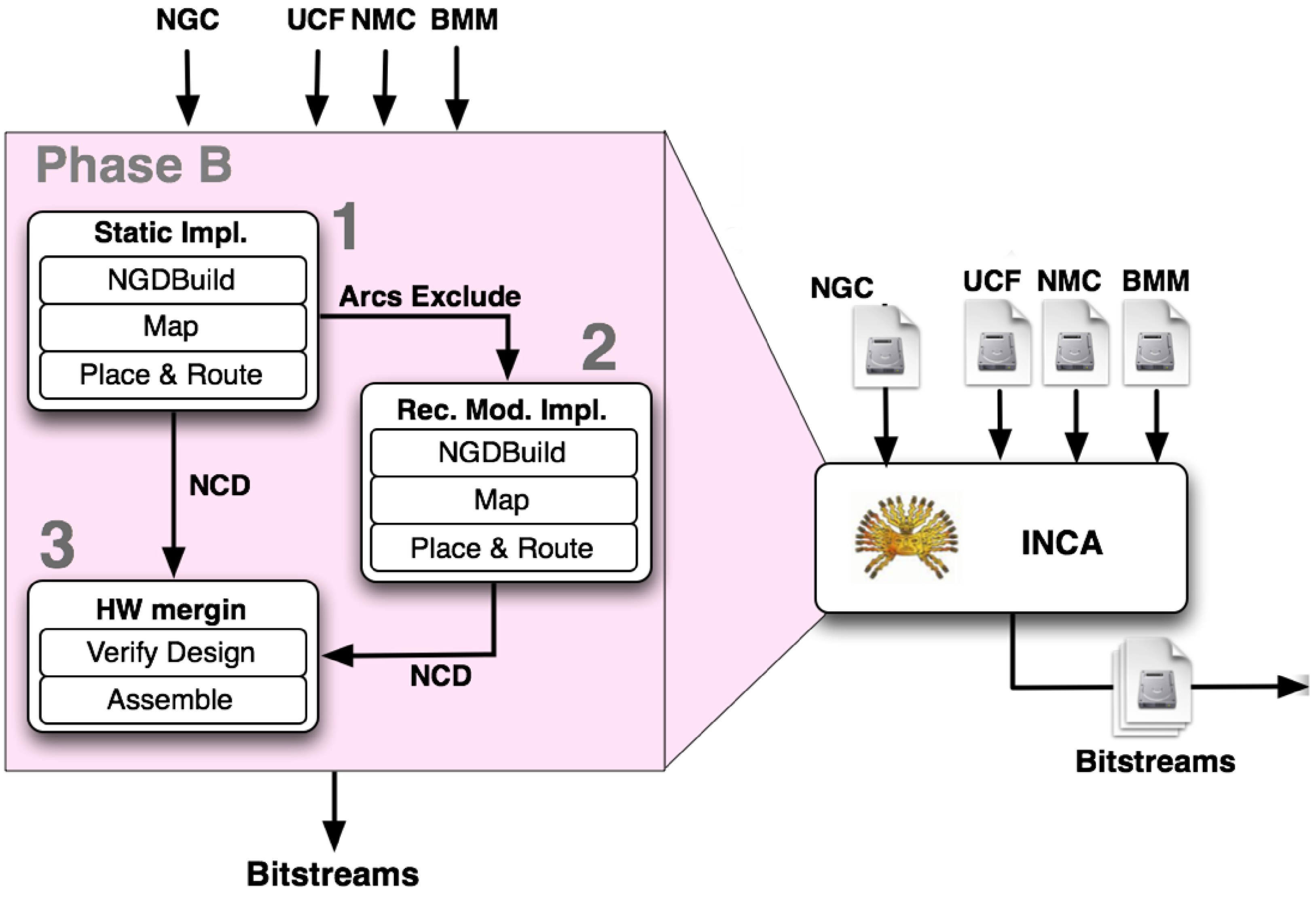
- IPGen



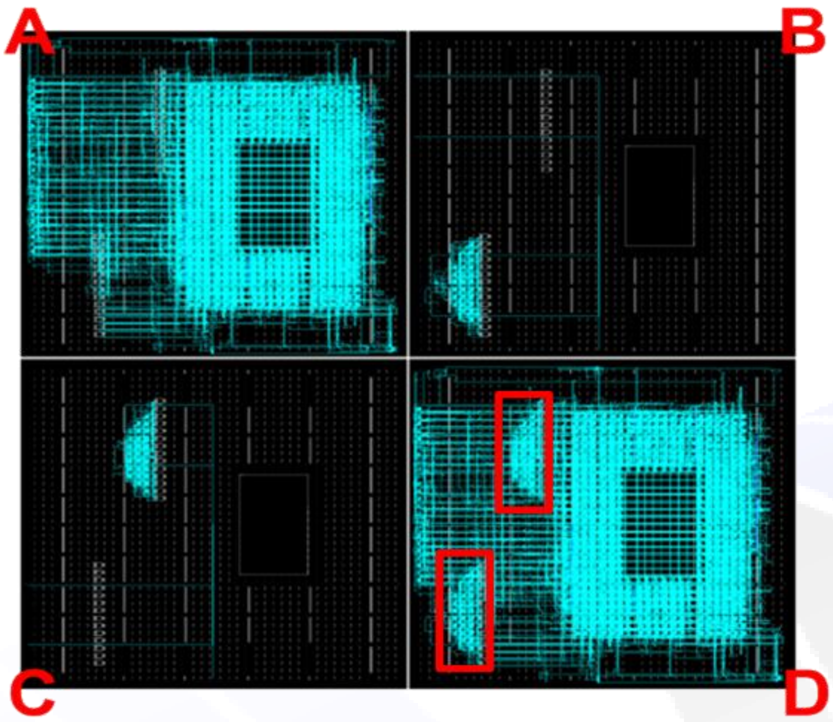
- IPGen Tests

IP-Core	4-input LUTs	Ratio	Slices	Ratio	ar	d	Time(s)
IrDA	15		11		136	0.081	
	146	9.73	103	9.36	136	0.758	0.045
FIR	273		153		272	0.562	
	308	1.13	173	1.13	272	0.636	0.058
RGB2YCbCr	848		913		952	0.959	
	1028	1.21	940	1.03	952	0.987	0.063
Complex ALU	1750		950		952	0.997	
	2089	1.19	1079	1.14	1088	0.991	0.071

Phase B - INCA


NGC
UCF
NMC
BMM
Phase B
1
Static Impl.
NGDBuild
Map
Place & Route
Arcs Exclude
2
Rec. Mod. Impl.
NGDBuild
Map
Place & Route
NCD
3
HW mergin
Verify Design
Assemble
NCD
Bitstreams
NGC
UCF
NMC
BMM
INCA
Bitstreams

Phase B - Results



- General Reconfiguration Approach

- INCA Results

Phase B	s200		VP7		FX12	
(#)	(s)	%	(s)	%	(s)	%
1	133,86	18.8%	224.91	18.0%	1060.28	39.0%
2	110.72	14.1%	165.69	13.3%	249.72	9.1%
3	466.19	65.5%	852.13	68.5%	1407.36	51.8%
Total	710.77	100.00%	1242.73	100.00%	2717.36	100.00%

GNU/Linux OS

- The Software Architecture allows the former two phases to be integrated
- Reconfiguration process handled has a standard OS feature
- The */dev* directory contains the information on the *Reconfigurable Devices*
- *The /dev/icap* has to be used to handle the physical reconfiguration processes

Shining Results

Modules	Bus	Speedup
Software		1
4 pixels per time	32bit OPB	0.97
Pipelined Core	32bit OPB	1.99
Pipelined Core	32bit PLB	2.08
Pipelined Core	64bit PLB	2.41

Canny Algorithm

Sobel Convolution

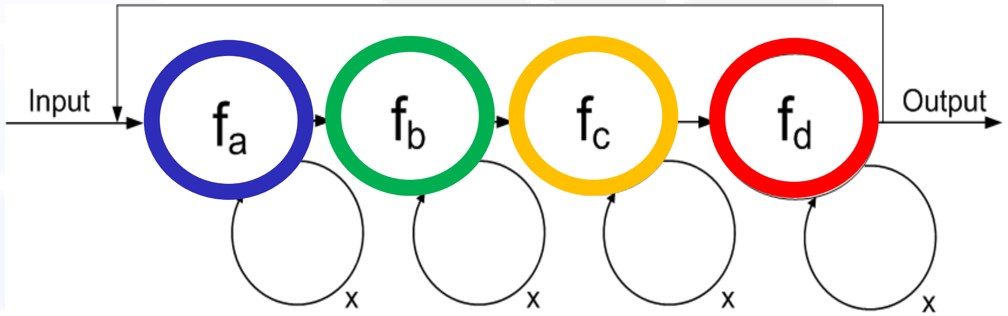
Modules	Bus	Speedup
Software		1
4 pixels per time	32bit OPB	0.79
Pipelined Core	32bit OPB	1.73
Pipelined Core	32bit PLB	1.74
Pipelined Core	64bit PLB	1.68

Modules	Bus	Speedup
Software		1
4 pixels per time	32bit OPB	0.65
Pipelined Core	32bit OPB	1.33
Pipelined Core	32bit PLB	1.41
Pipelined Core	64bit PLB	1.9

Laplace Convolution

Case Study: Digital Image Processing

- The canny edge detector is used to detect the edges in a given input image i [Kb]
- 4 functionalites
  - Image smoothing → remove the noise
  - Gradient operator → highlight regions with high spatial derivative
  - Non-maximum suppression → reveal the edges
  - Hysteresis → remove false edges
- Each functionality has to be executed using an input of j [Kb]
  - $j \leq i$ and $x = i/j$



Case Study: Digital Image Processing

- Time analysis to identify a first partition in HW core and SW core
  - Non-maximum suppression implemented as a SW core
  - Image smoothing, Gradient operator and Hysteresis implemented as HW cores

Applications Functions	Slices	Percentage
Static side and non-maximum suppression	2662	54
Image smoothing	245	4
Gradient computation	2168	44
Hysteresis threshold	5343	108

Conclusion and Future Work

- The Shining methodology provides an effective and low cost approach to the partial dynamic reconfiguration and mixed HW-SW execution problems
- Introduces dynamic reconfiguration features at design time
- The proposed flow organizes the input specification into three different components: hardware, reconfigurable hardware and software, managed by proper portion of the methodology
- The complete design flow has to be automated to help the designer in a more effective way

Any Question?

- Thank you...