The HeartBeat model

A platform abstraction enabling fast prototyping of real-time applications on NoC-based MPSoC on FPGA

Francesco Robino, Johnny Öberg

KTH Royal Institute of Technology

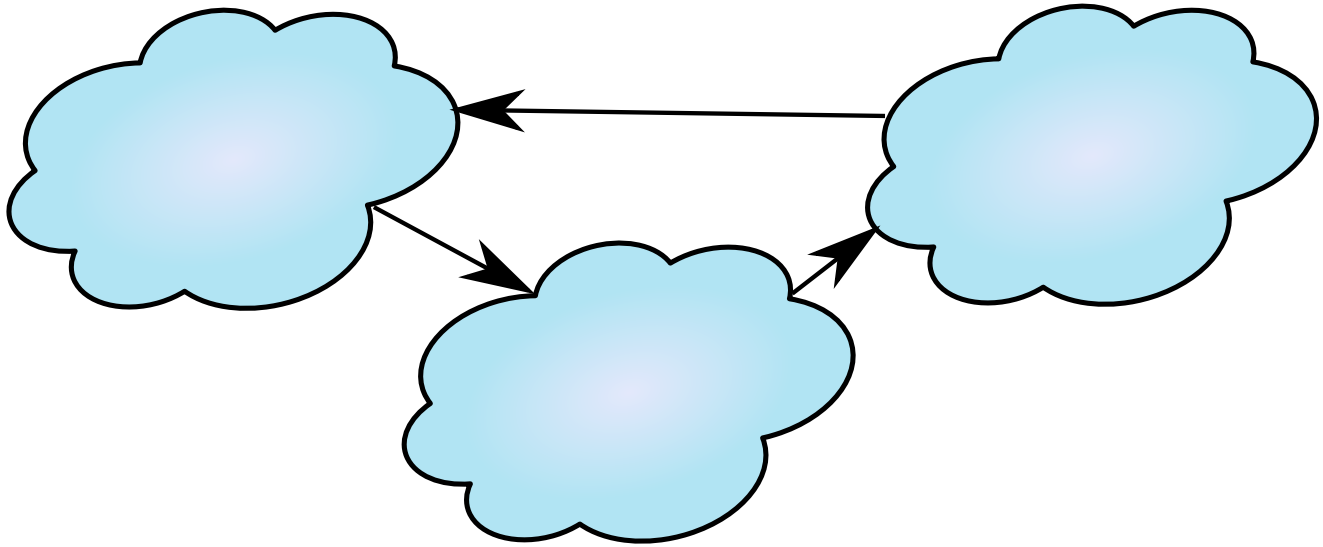
ReCoSoC 2013

Overview of the talk

- Motivation
  - What is the problem?
  - Benefits solving the problem
- Our solution
  - Modeling techniques
  - Our idea
  - From idea to implementation
  - Experimental evidences and examples
- Conclusion

Motivation: Embedded systems modeling

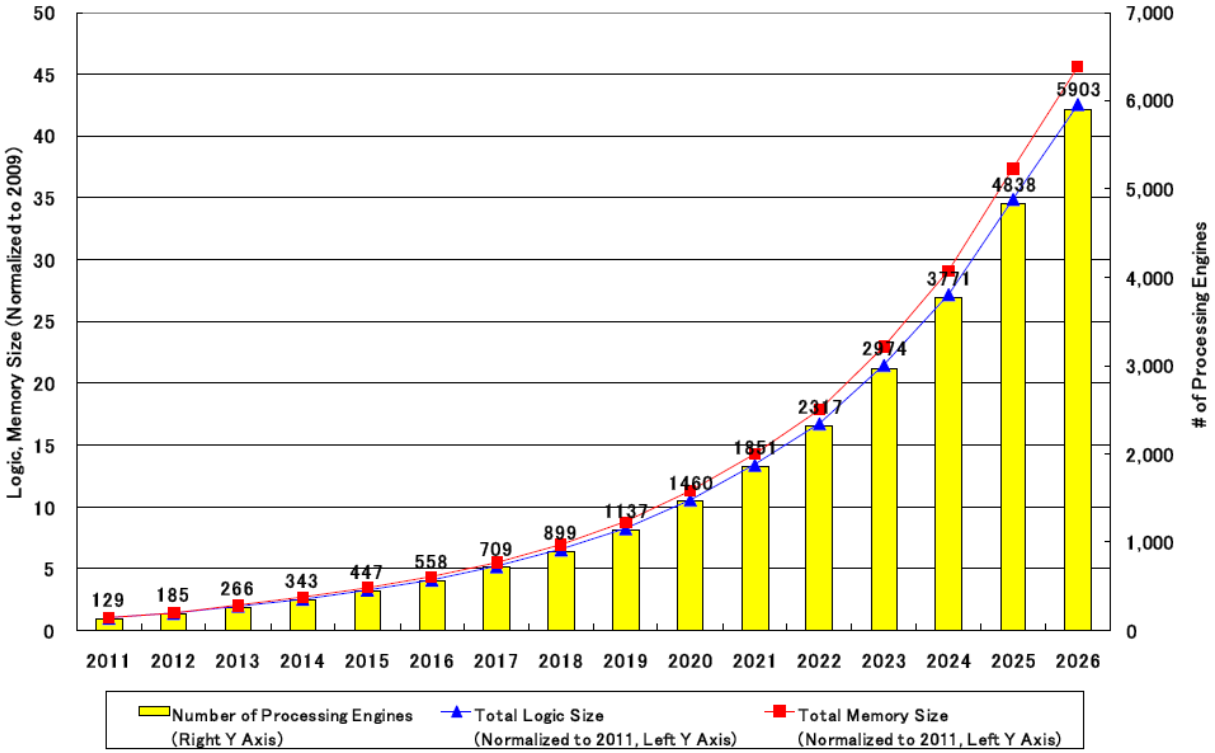
- Embedded systems are getting very complex
- We want to **abstract**, **reduce details** in the system model



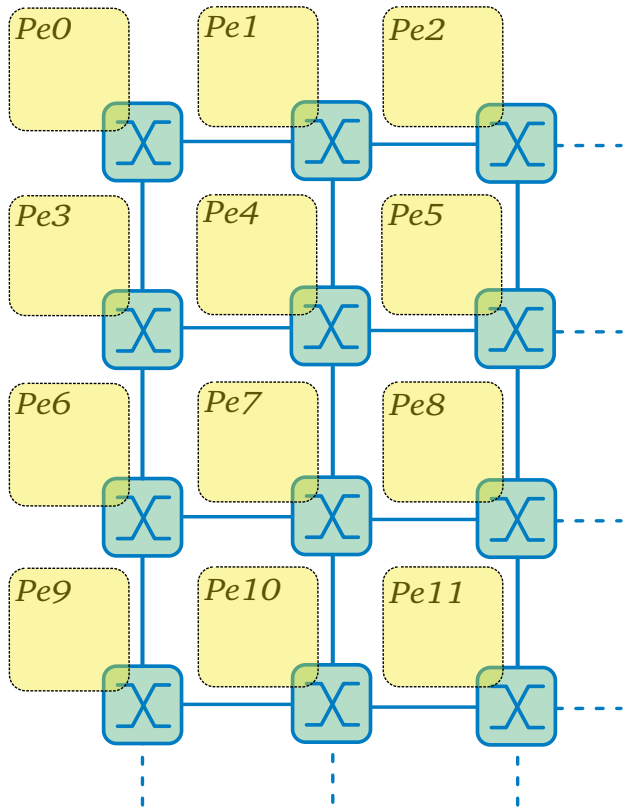
- We want **S**ystem **D**esign **A**utomation to automatically add details
  - it implements the embedded application on **multi-processor** platforms
  - the automated synthesis of the whole system must be **fast**!

Motivation: Embedded system architecture

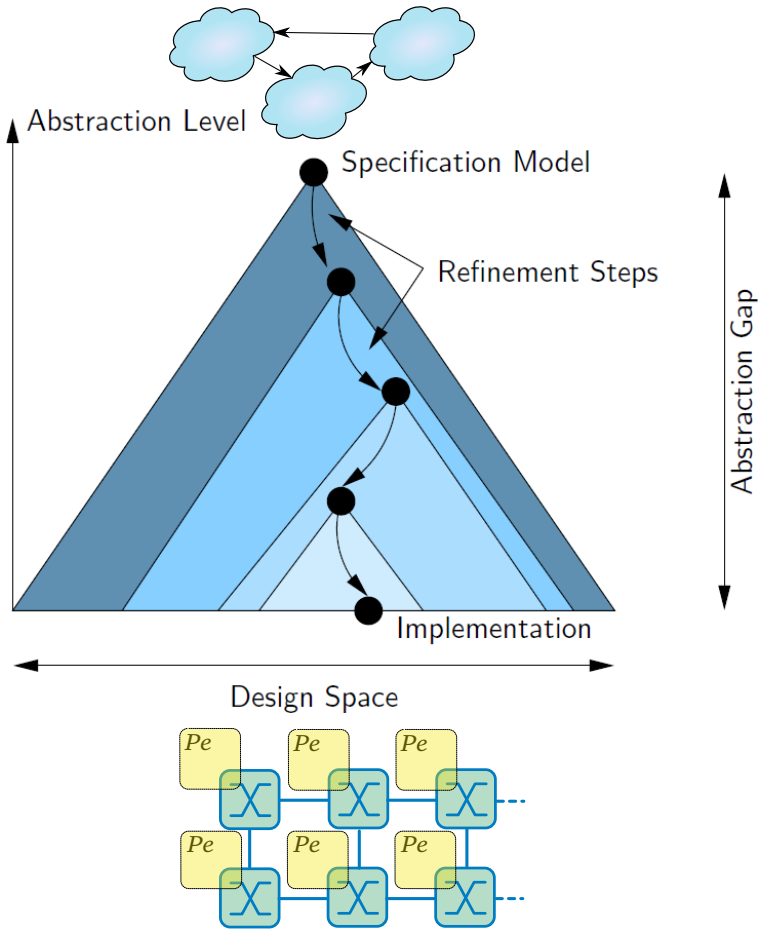
- Embedded *architectures* are getting very complex
- **details** are **increasing!**



- **sea-of-cores / processors**
- NoC-based MPSoC



The problem



- Huge abstraction gap
- Huge design space
  - Platform allocation
  - Processes binding
  - Processes schedule
- Platform:
  - complex to program
  - complex to generate

Filling the gap through design automation: benefits

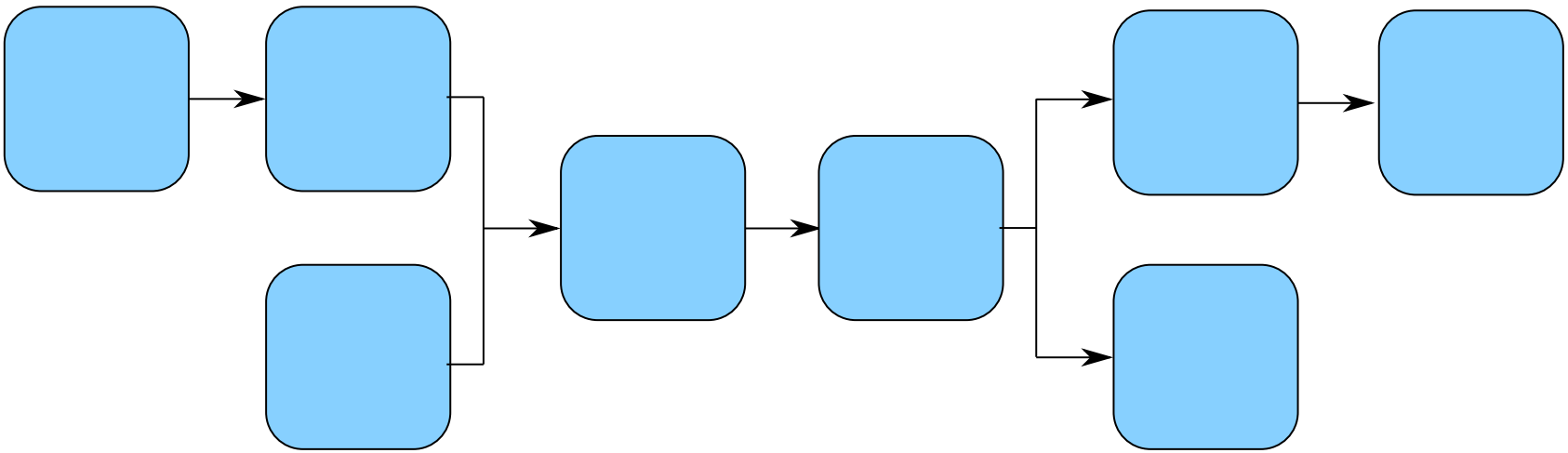
- Reduces time to market, reduces errors, reduces complexity...

Modeling techniques

The theory of models of computation (MoCs)

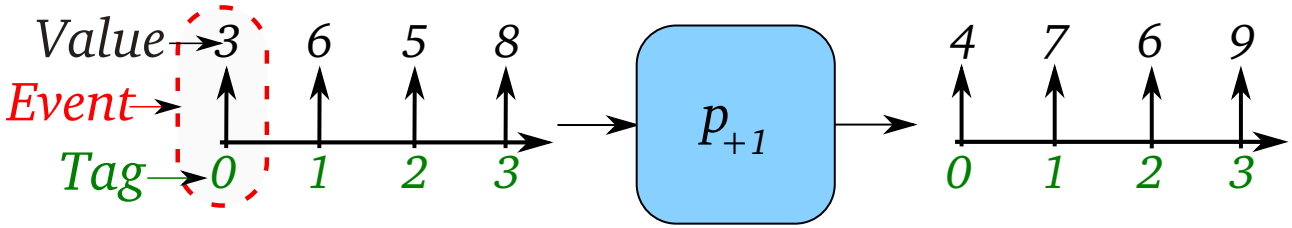
A **MoC** defines **communication and execution rules between nodes executing the operations**.

- The MoC rules can be described through formal notation (math)
- Embedded application modeled as a set of communicating processes

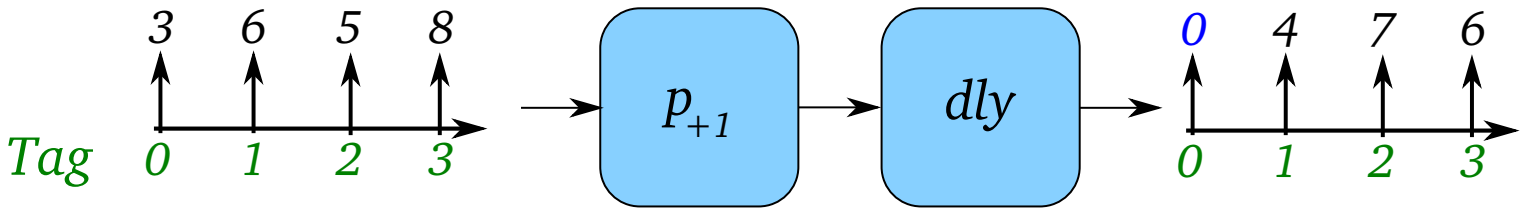


The synchronous MoC

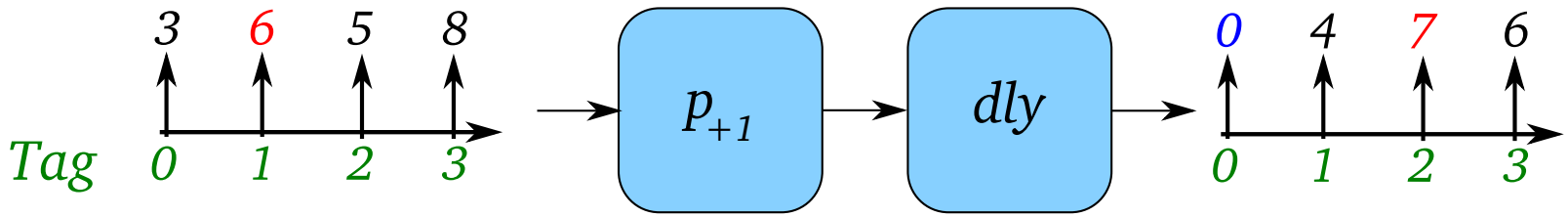
- We use the *synchronous MoC*
- Used in synchronous languages (i.e. Lustre) to guarantee real-time
- Combinatorial application in the synchronous MoC:

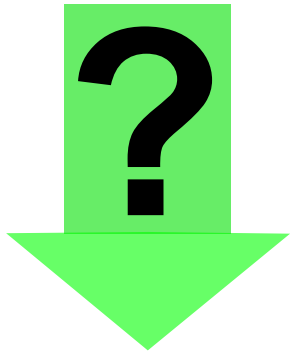


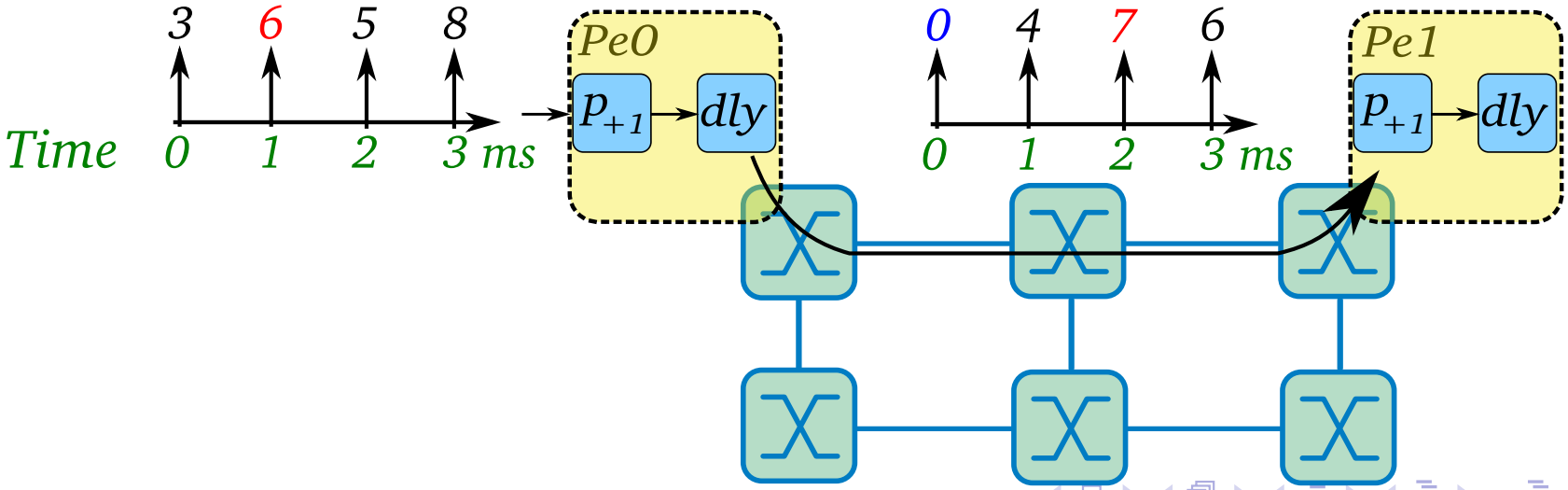
- Application with memory in the synchronous MoC:



From synchronous MoC to NoC-based MPSoC

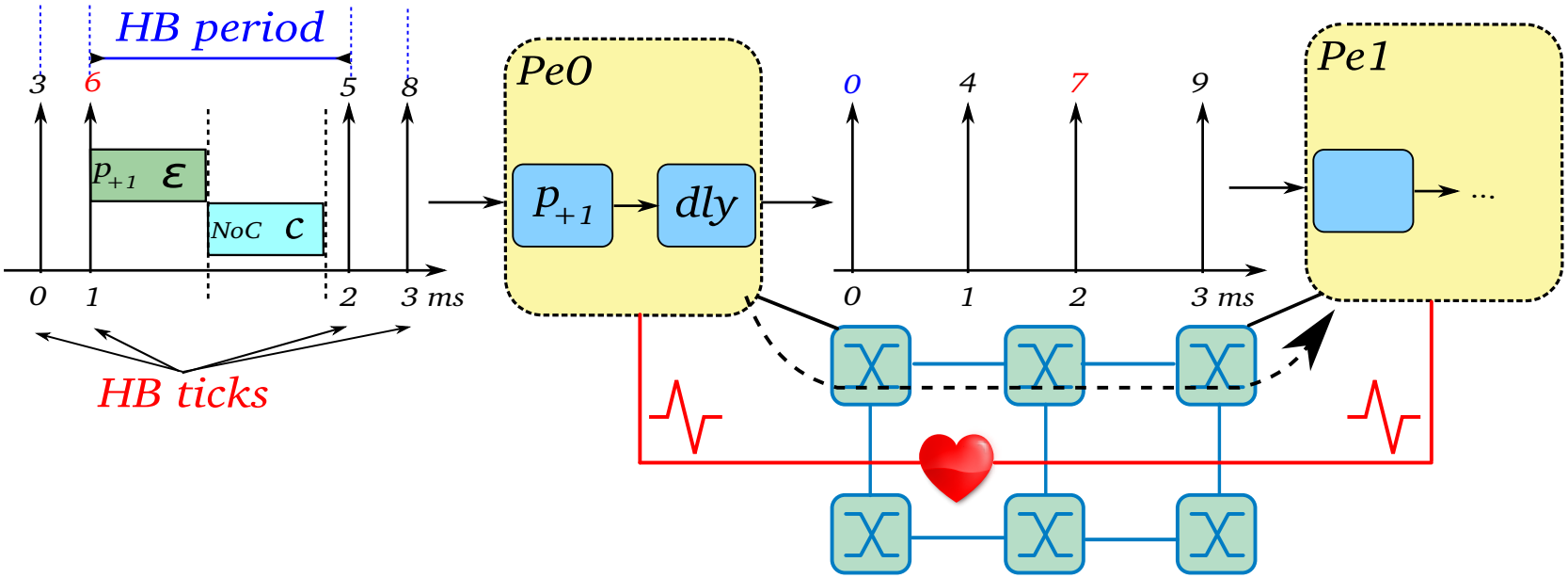






From synchronous MoC to NoC-based MPSoC

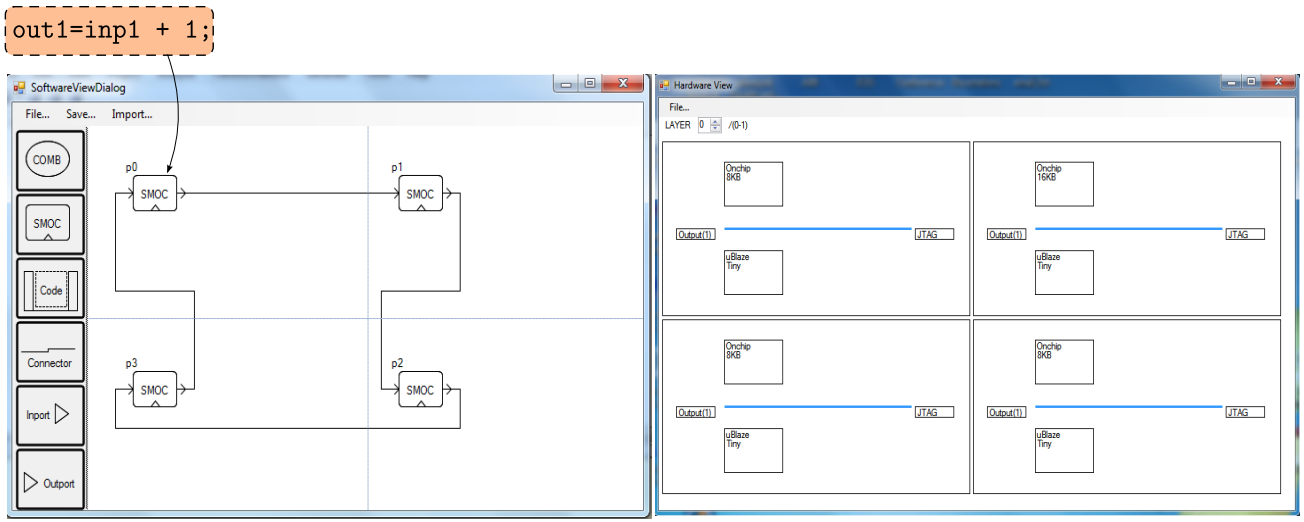
- We define an intermediate layer, the HeartBeat model:
  - fills the abstraction gap adding details about timing and architecture
  - enables the MPSoC to expose the same semantics of the MoC
  - critical path (RTL, sea of gates) $\rightleftharpoons$ HB period (MoC, sea of cores)

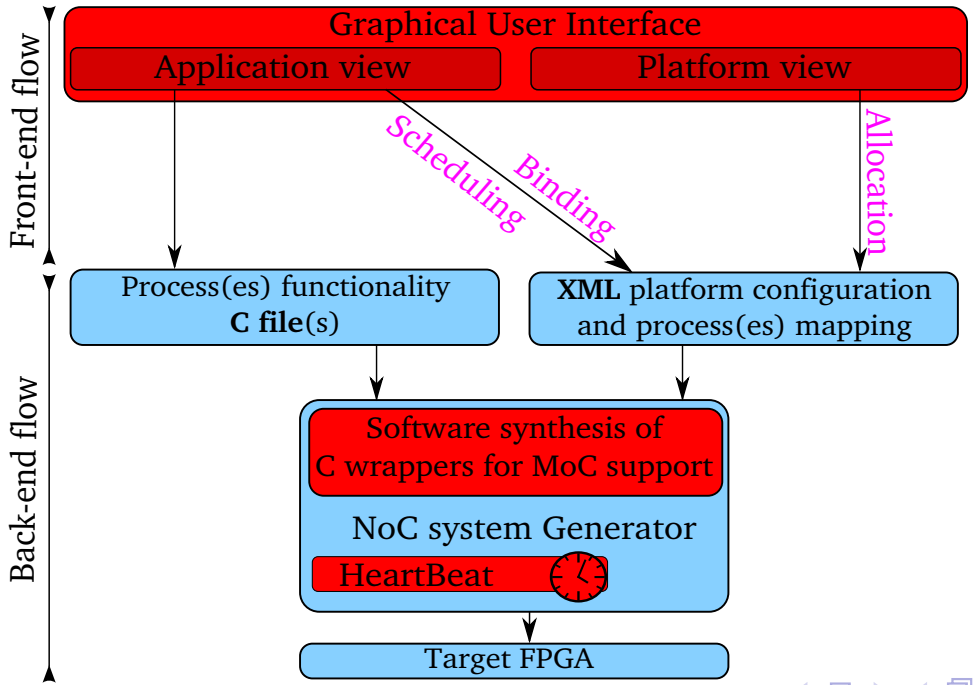


The HeartBeat model (for this simple example)

$$t_{HB} \geq \max_{m=0}^{M}(t_{\epsilon_i} + t_{c_i})$$

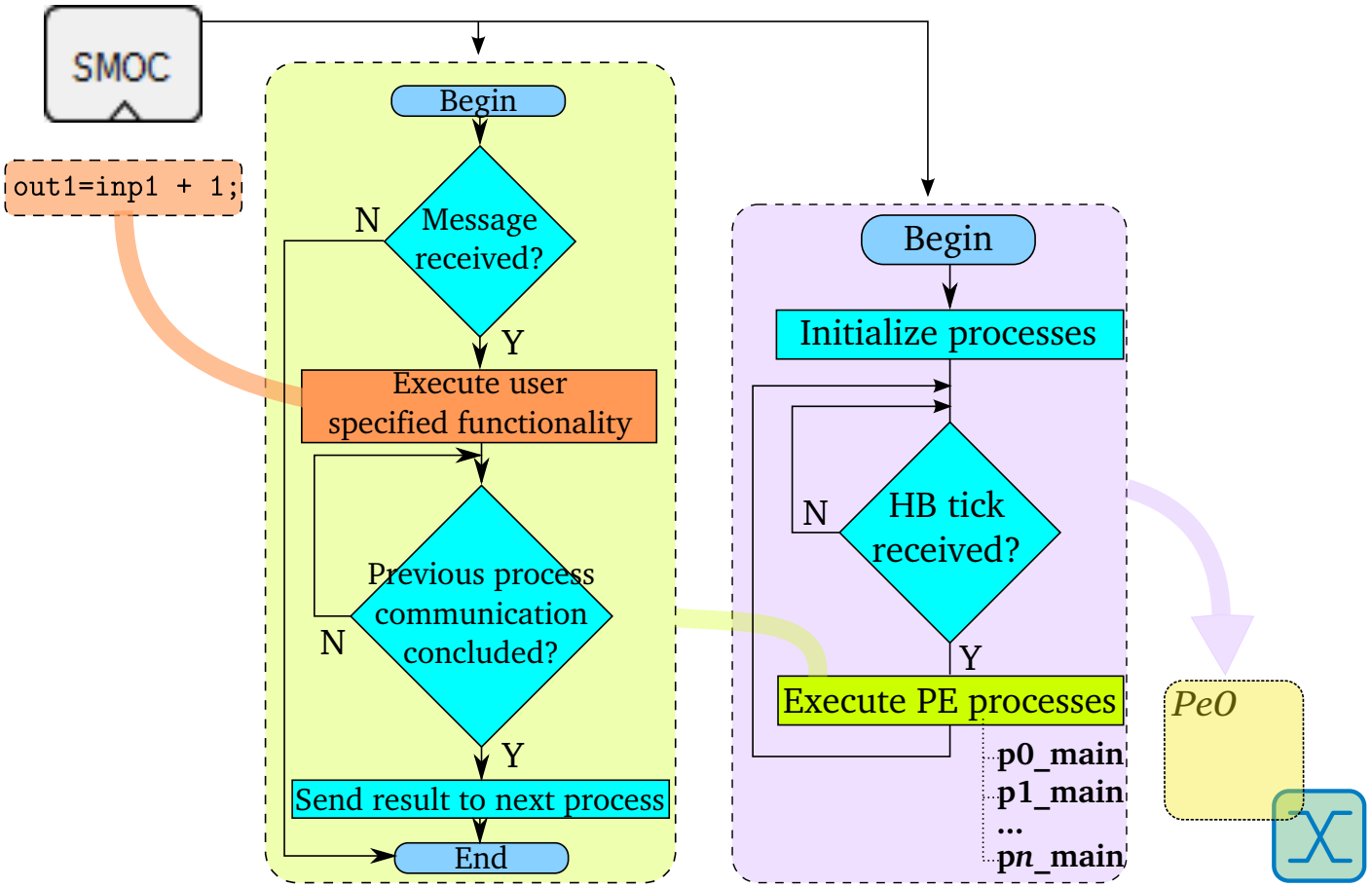
A HW/SW co-design flow based on the HeartBeat model





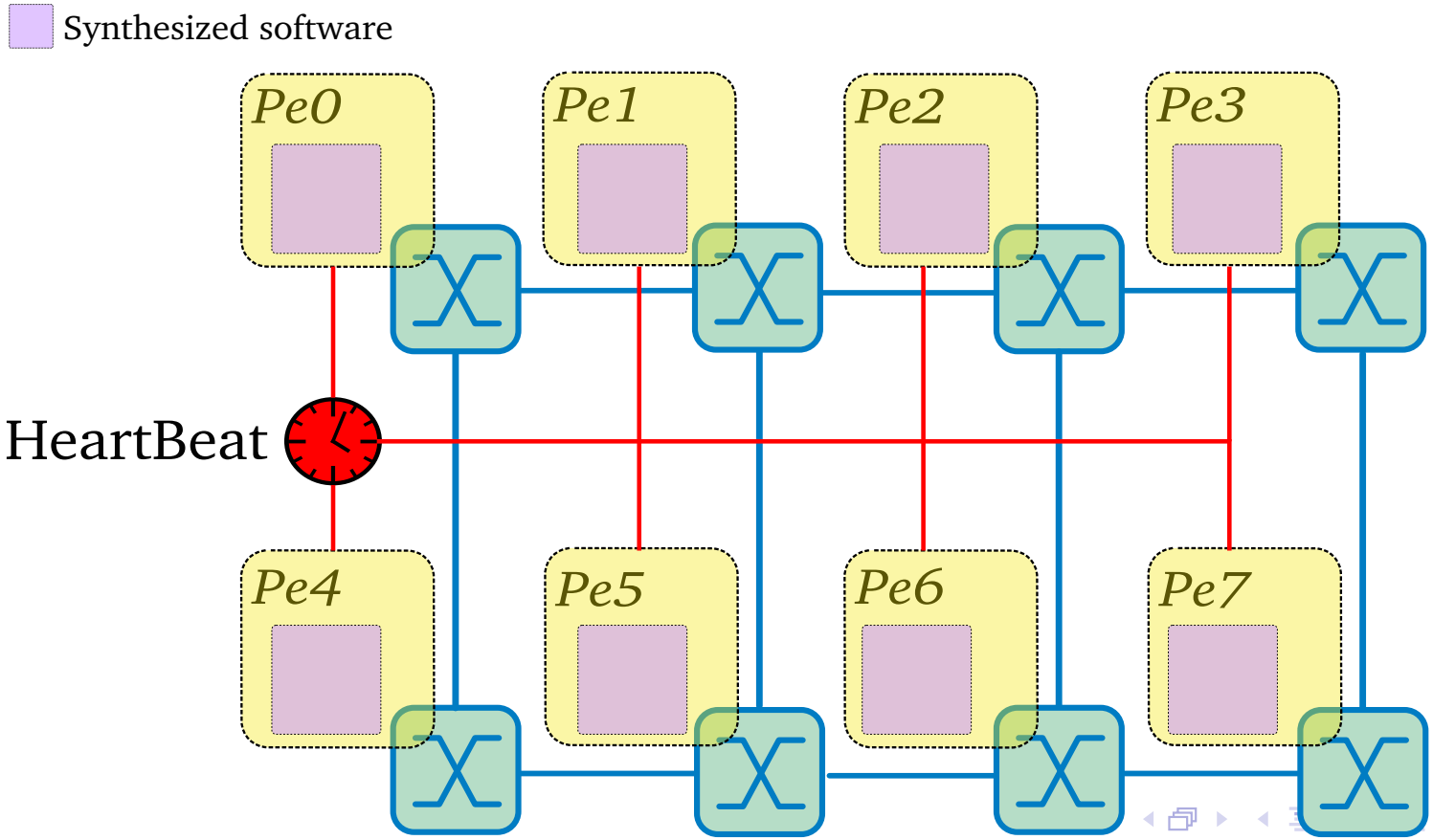
SW synthesis technique

- Application view:
  synthesis of software to access NoC API and synchronize on HB tick



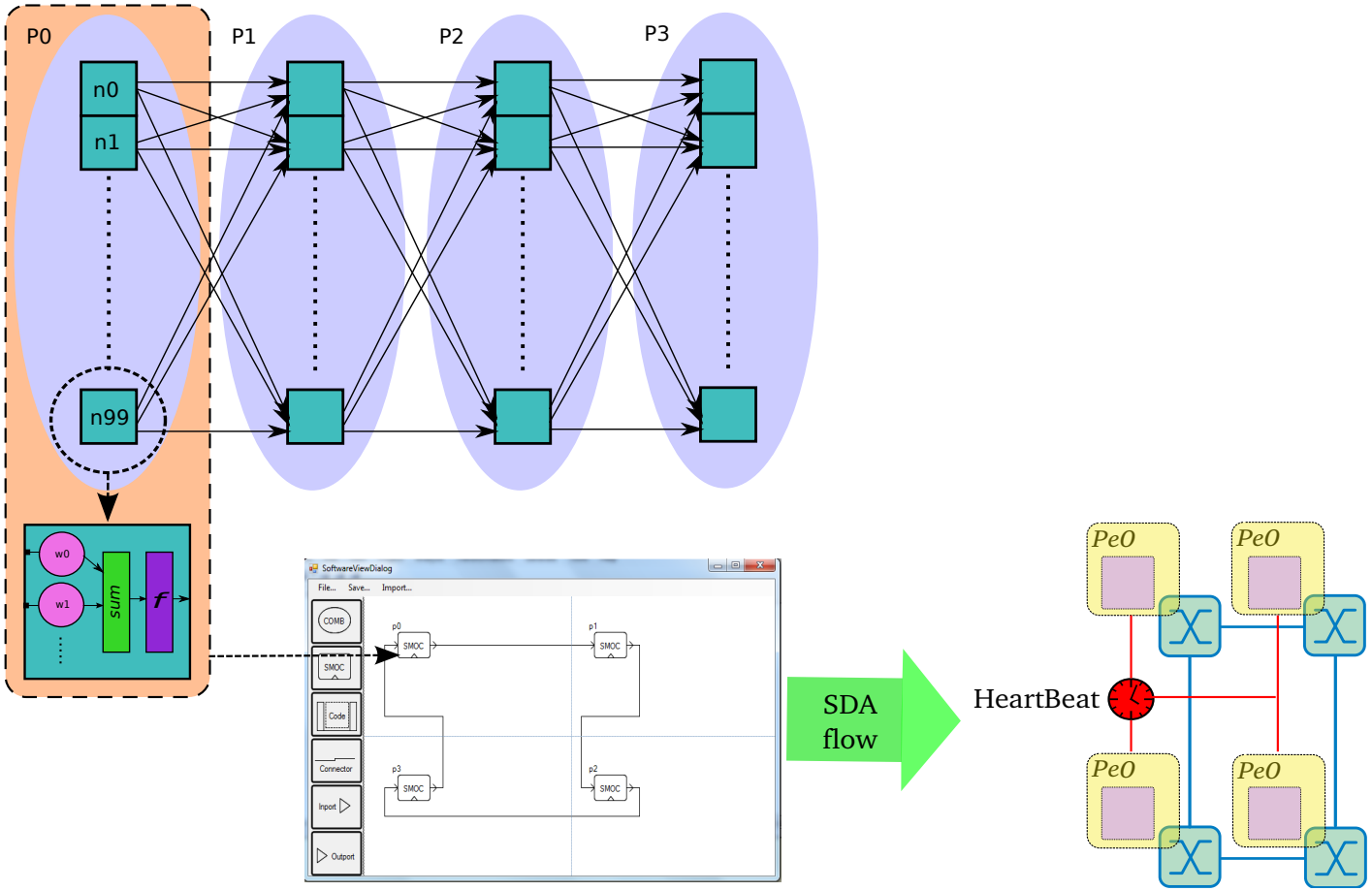
HW synthesis technique

- Platform view:
  synthesis of NoC-based MPSoC with HeartBeat support
  - A HW timer generates HB ticks
  - The HB period (timer period) can be defined by the user, following the Heartbeat model



Case study

- A neural network, 4 layers, each layer 100 neurons
- Each layer on a separate PE (4 layers, 4 PEs, 2×2 NoC)
- Generation of files for prototype synthesis on FPGA < 1 second



Case study

- Optimize the application: find the minimal HB period of the system

The HeartBeat model (for this simple example)

$$t_{HB} \geq \max_{m=0}^{M}(t_{\epsilon_i} + t_{c_i})$$

- t_{ϵ_i} has been found through extensive emulations. Its value is 5,97 ms
- t_{c_i} can be found following models proposed in referenced papers
- The HeartBeat period is totally dominated by t_{ϵ_i}, it is an order of magnitude larger than t_{c_i}
- Concluding the maximal HB frequency of the system is $\frac{1}{5{,}97} \simeq 150$ Hz

Results and related work

Design flow	MAMPS	Our work
Target platform	2 processors	4 processors
Platform interconnection	2×1 NoC	2×2 NoC
Creating system model	manual(SDF)	manual(SY)
Creating application model	manual	semiautomatic
Generating architecture model	1 second	$\sim$100 millisec
Mapping the design	1 minute	manual
Generate Xilinx/Altera project	16 seconds	$\sim$500 millisec

Table: SDA flows steps rapidity

	COMPSOC	Open-Scale	MAMPS	Our work
MoC entry	HSDF	KPN	SDF	Sync
NoC arch.	Mesh	Mesh	Mesh	Mesh
NoC dim.	2D	2D	2D	2D,3D
API generation	Y	N	N	Y
Prototype	Y	Y	Y	Y
OS avail.	Y	Y	N	N

Table: SDA flows properties comparison

Conclusion

- The *HeartBeat model* is an intermediate platform model bridging the abstraction gap between the synchronous MoC and a NoC-based MPSoC platform
- The *HeartBeat model* enables a SDA flow to synthesize an embedded application onto a NoC-based MPSoC on FPGA, provided that the timing constraints of the HeartBeat model are met
- The design flow reduces the complexity of designing embedded systems for NoC-based MPSoCs through a GUI
- The automated flow speeds up the design process of embedded systems.

More info and tutorials

https://forsyde.ict.kth.se/noc_generator/