

System Level Design Methods and Tools for Modeling and Development of MPSoCs/CMPs-Heterogeneous

Antonio Núñez



University of Las Palmas de Gran Canaria

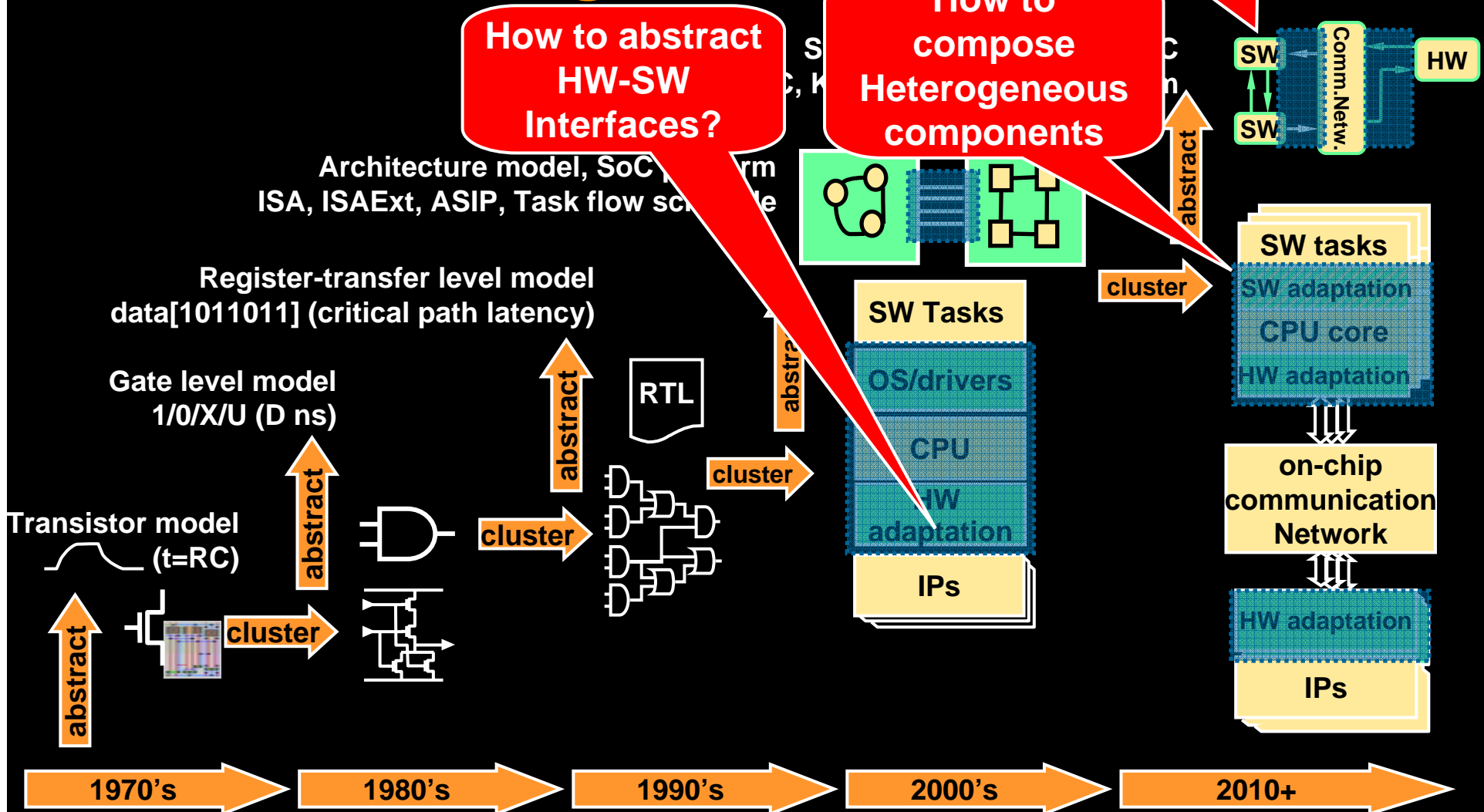
Institute for Applied Microelectronics



ASIC is down, the future is MIP

CMP/Hetero Design and Core

from Wires to Higher level interconnect, TCP



How to abstract HW-SW Interfaces?

How to compose Heterogeneous components

How to compile & sync ||tasks for Heterogeneous processors

Architecture model, SoC, Arm ISA, ISAExt, ASIP, Task flow sc...

Register-transfer level model data[1011011] (critical path latency)

Gate level model 1/0/X/U (D ns)

Transistor model ($t=RC$)

RTL

SW Tasks

OS/drivers

CPU

HW adaptation

IPs

SW tasks

SW adaptation

CPU core

HW adaptation

on-chip communication Network

HW adaptation

IPs

1970's

1980's

1990's

2000's

2010+

Adapted from F. Schirmer (Cadence Design Systems Inc.)

System-Level Design and Verification



- Narrowing the design and verification gap requires methods and tools which work at the system-level
 - From transistors to cells (80's)
 - From cells to RTL (90's) -> VHDL and Verilog
 - From RTL to IP blocks, cores, interconnects -> SystemC
 - From cores to multicore synthesis and modeling -> ?
- Electronic System Level (ESL) tools and massive IP-reuse are the keys to overcome the complexity of future MPSoCs [Gartner]
 - Shorten design cycles
 - Reduce design risk
 - Increase design productivity
 - Deal with verification complexity
 - Deal with parallel programming embedded support

1 Introduction

Methodology

Application Modeling

Architecture Modeling

Mapping, Analysis & Refinement

CASSE

Case Studies

Conclusions

System-Level Design and Verification



- Most current ESL tools bases on Virtual Prototyping:
 - Software model of the complete system (executable specification)
 - Faster simulations with 100% accurate functionality and timing
 - IP libraries developed with SystemC (processors, busses, etc)
 - Early software development and architectural analysis
- Better... but still not enough
 - Typical VP ESL tools provide results when projects are already at >70% of the development time [CoFluent]
 - VP needs accurate HW models and embedded SW to be available (takes some effort)
 - VP does not help in the HW/SW integration
 - VP does not help in taken the right architectural decisions at the beginning of the design cycle

1 Introduction

Methodology

Application Modeling

Architecture Modeling

Mapping, Analysis & Refinement

CASSE

Case Studies

Conclusions

IUMA goals for HiPEAC cluster



- To develop new system-level methods and tools which
 - cope with the **complexity of current MPSoCs**
 - **Parallel application development**
 - Architecture-independent
 - **Fast and simple architecture modeling**
 - Early architectural exploration
 - Early SW development
 - **Application to architecture mapping**
 - Ease HW/SW integration
 - Ease design space exploration
 - **Path to implementation**
 - Progressive composition
 - Progressive refinement
 - can be used **very early in the design process**
- IUMA CASSE SystemC TLM Tool a starting point

- 1 Introduction
- Methodology
- Application Modeling
- Architecture Modeling
- Mapping, Analysis & Refinement
- CASSE
- Case Studies
- Conclusions