ETH zürich

# Provably Correct Memory Management

**Reto Achermann**

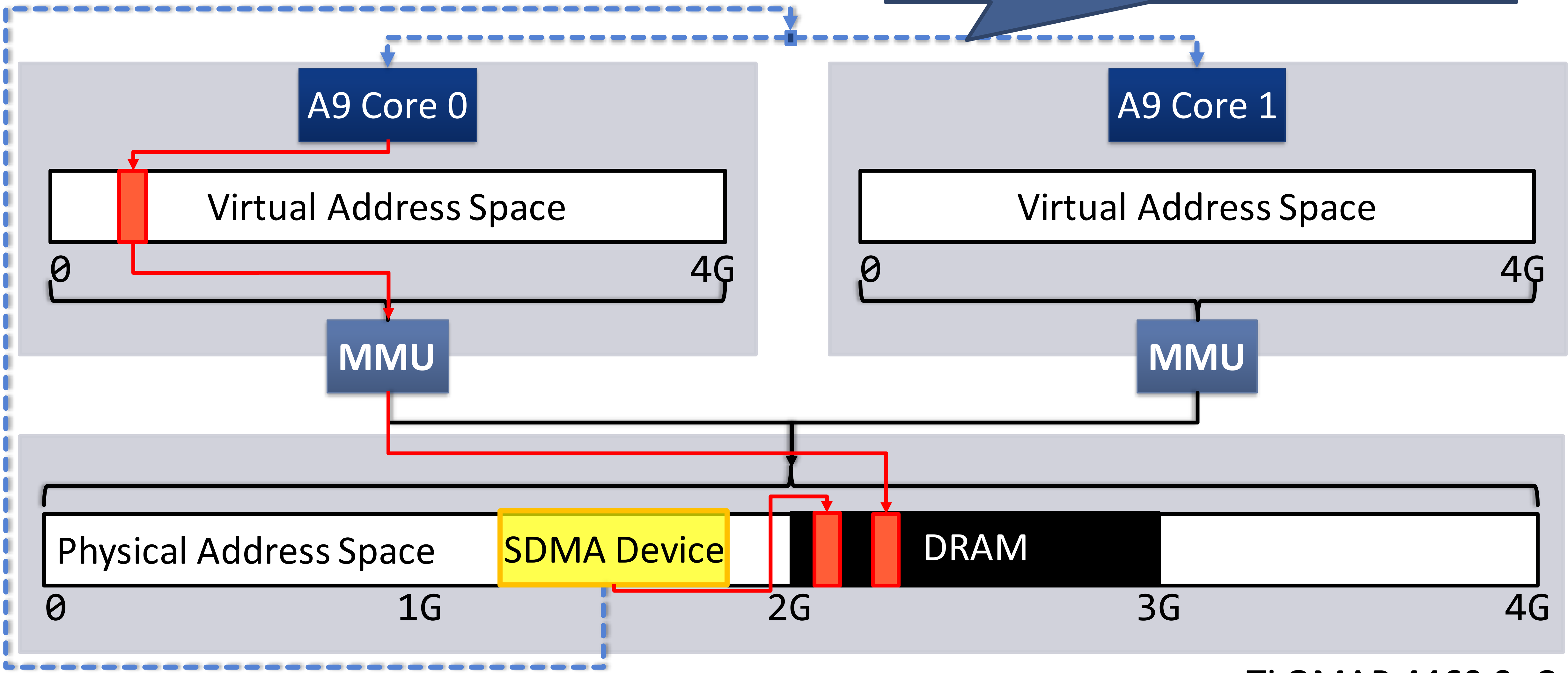
11th EuroSys Doctoral Workshop (EuroDW’17), Belgrade

Systems Group, Department of Computer Science, ETH Zurich

Systems@ETH zürich

# The naïve view of today's systems

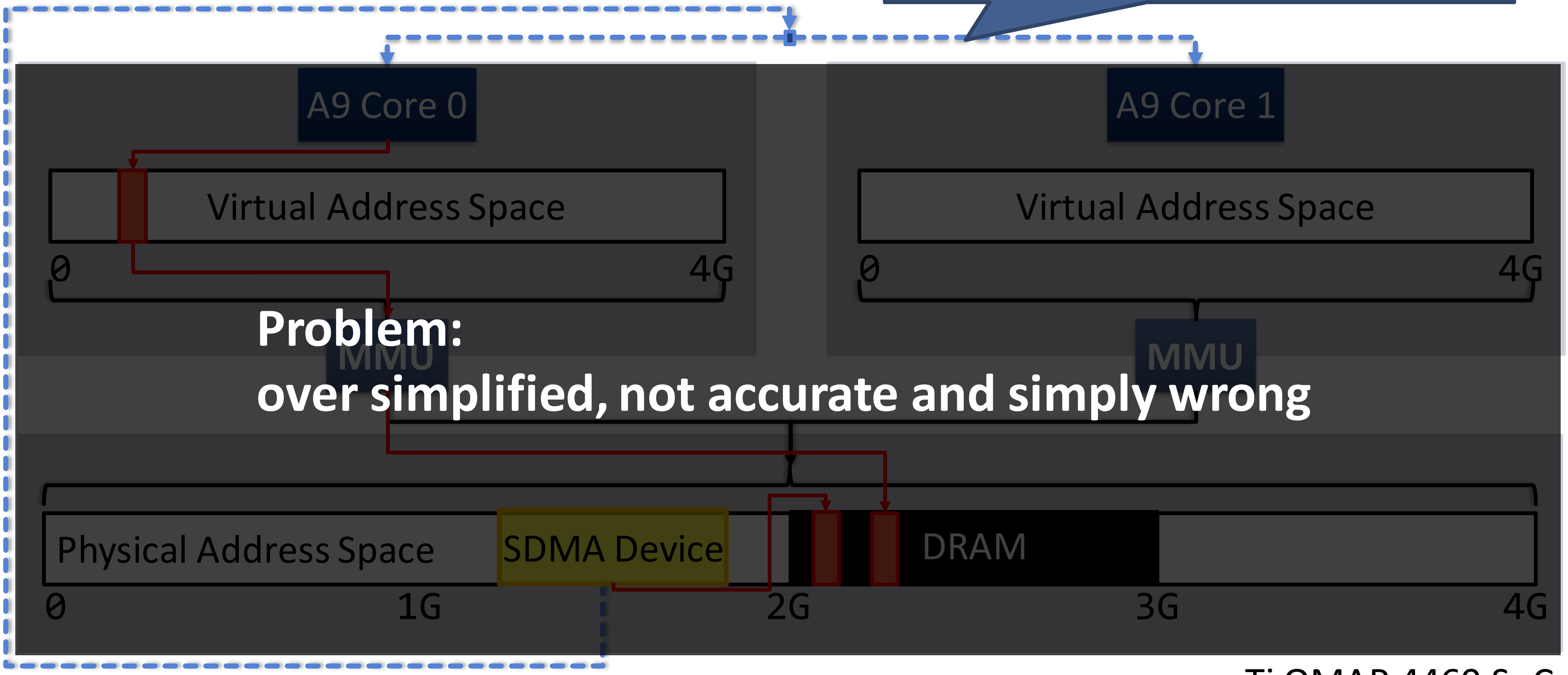
Lukas Humbel "Formalizing Interrupt Routing"



Ti OMAP 4460 SoC

# The naïve view of today's systems

**Lukas Humbel** "Formalizing Interrupt Routing"



**Problem:**
**over simplified, not accurate and simply wrong**

Ti OMAP 4460 SoC

# Reality: The devil is in the details

*Your mobile phone… 5-10 years ago!*

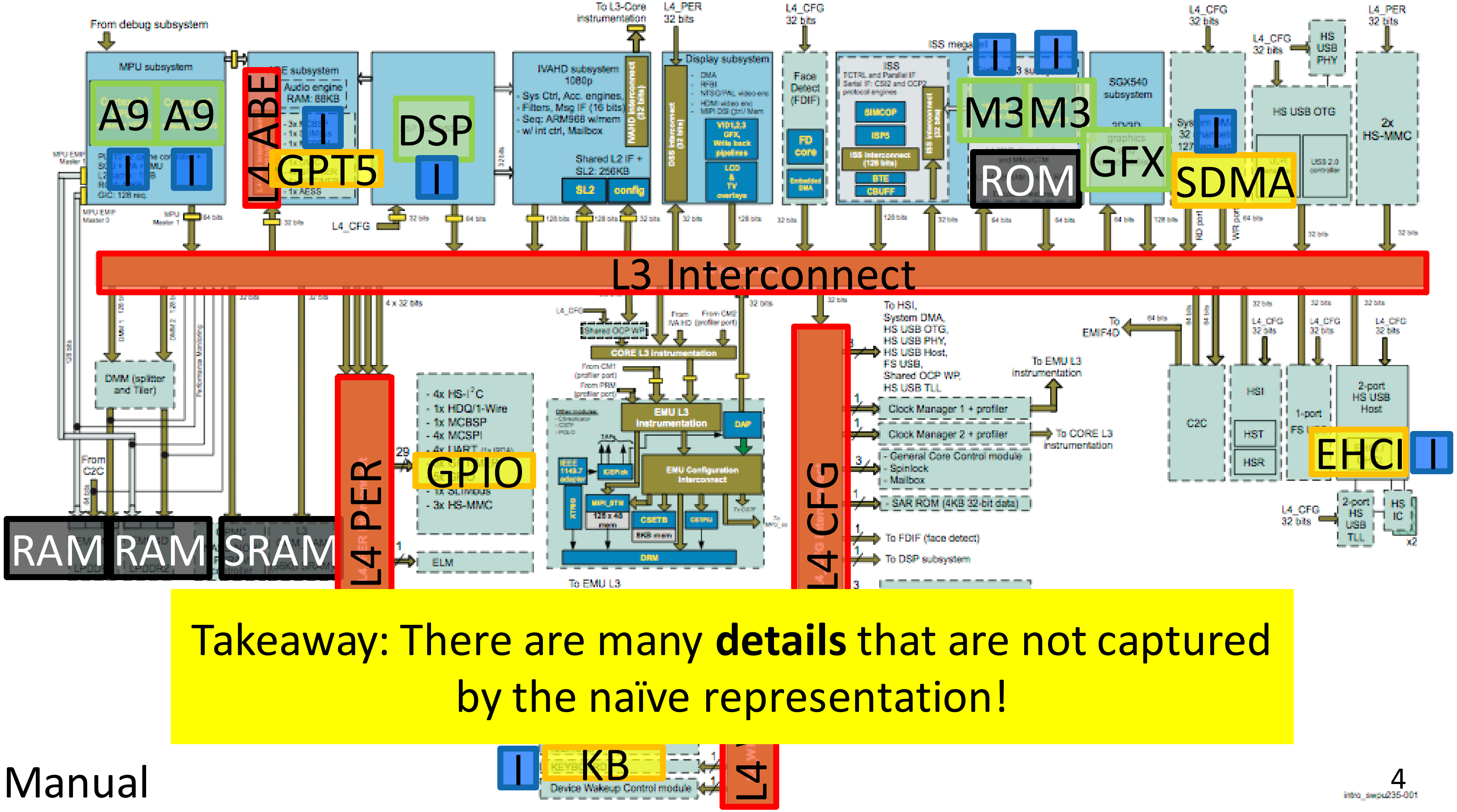
6+ heterogeneous cores

Private and shared memory

5+ Interconnects

Devices attached to different interconnects

Complex interrupt subsystem



Takeaway: There are many **details** that are not captured by the naïve representation!

OMAP 4460 SoC,
Technical Reference Manual

# Reality: The devil is in the details

*Your mobile phone… 5-10 years ago!*

6+ heterogeneous cores

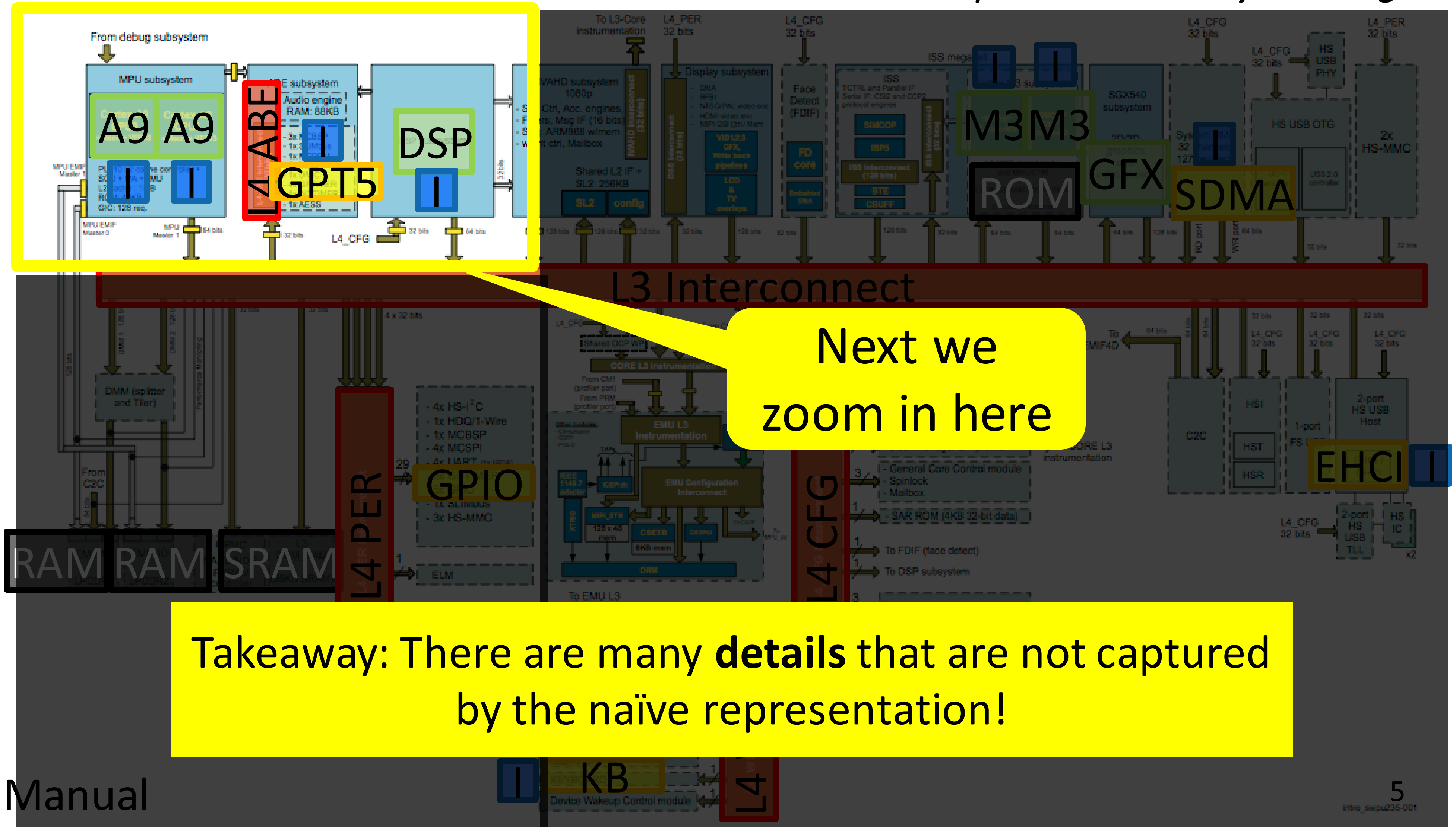
Private and shared memory

5+ Interconnects

Devices attached to different interconnects

Complex interrupt subsystem

OMAP 4460 SoC, Technical Reference Manual

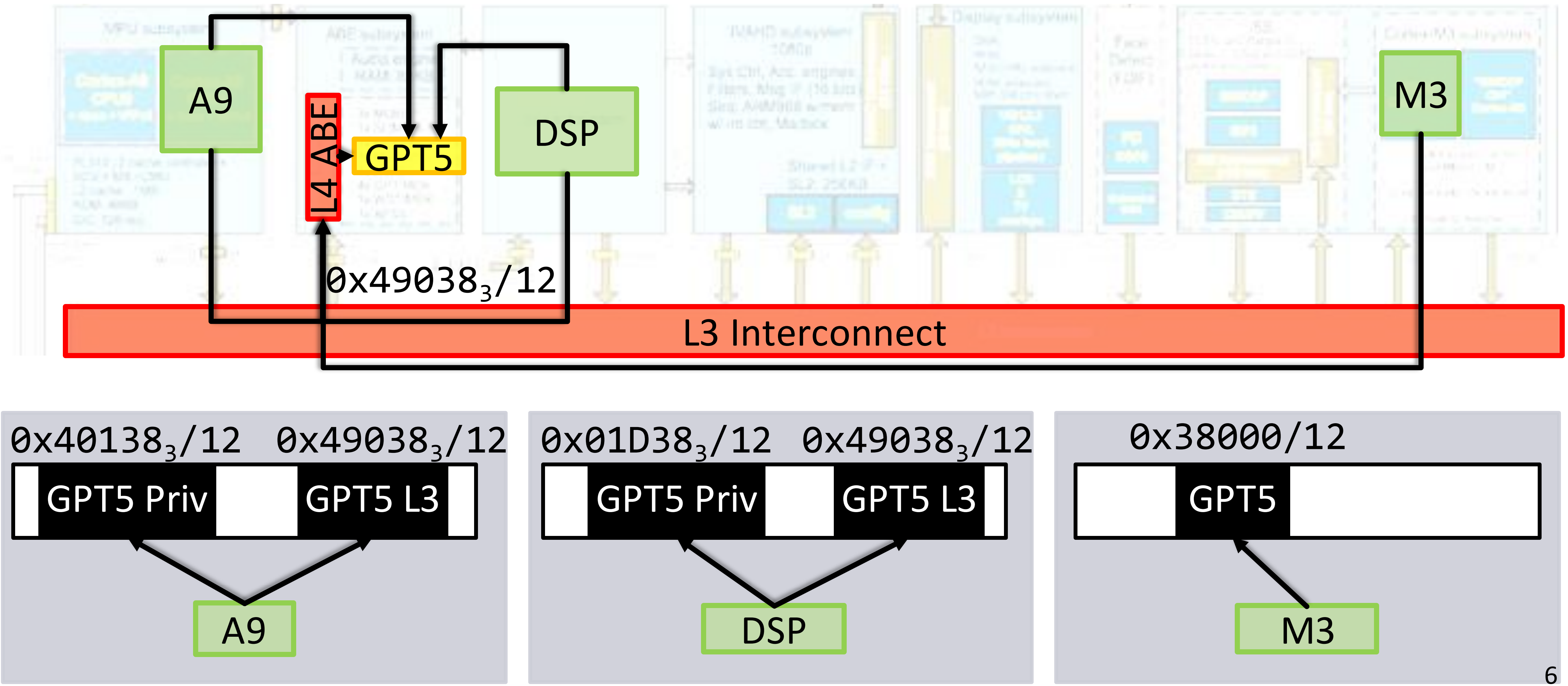


Next we zoom in here

Takeaway: There are many **details** that are not captured by the naïve representation!

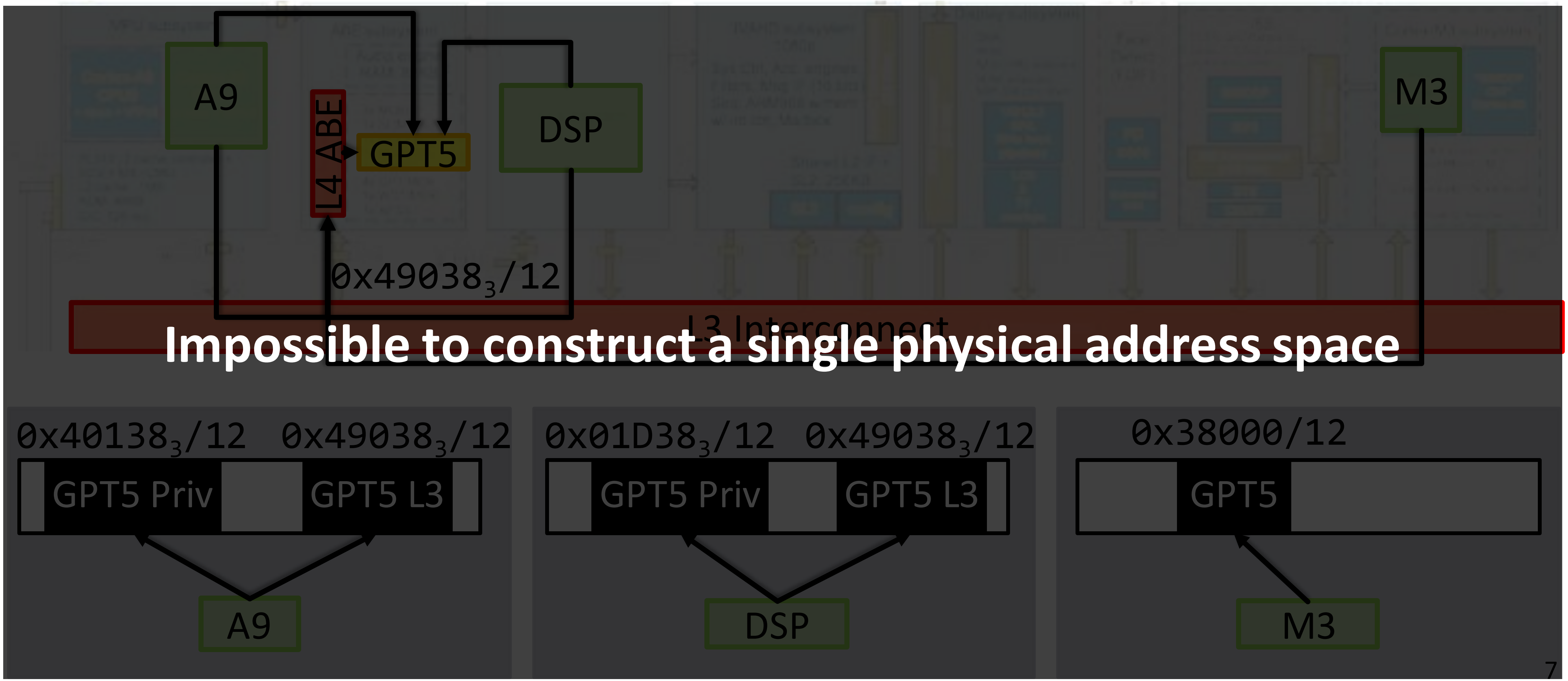
# There is NO uniform view of the system

OMAP 4460 SoC



# There is NO uniform view of the system

OMAP 4460 SoC

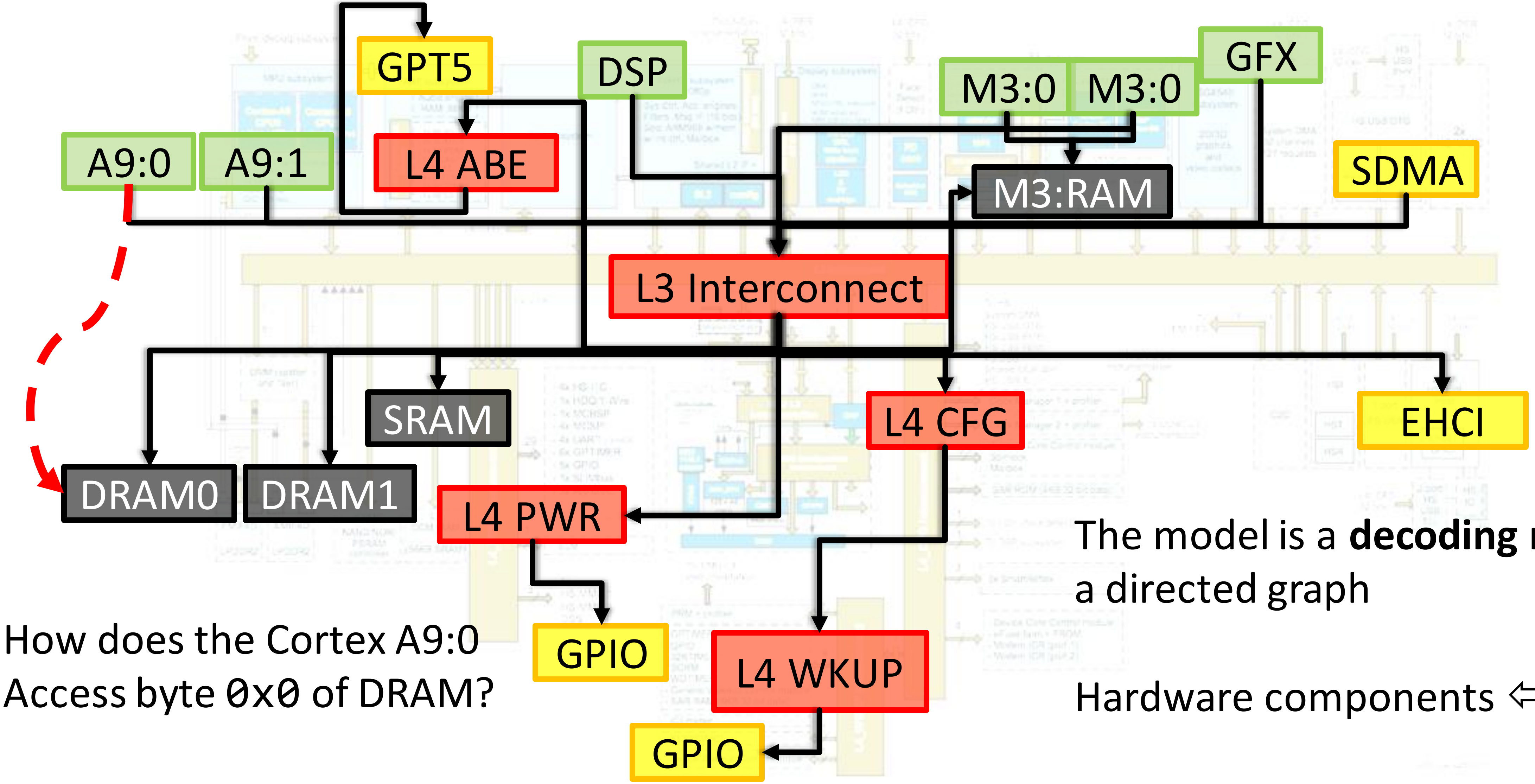


**Impossible to construct a single physical address space**

# Why do we need a formal model for memory accesses ?

- We **build** systems and want to write **correct** systems code
- Experience from the Barrelfish operating system:  
  dealing with this complexity every day.  
  *e.g. PCI programming, heterogeneity, resources, devices, new platforms*
- Problem:
  - Current abstractions make the **wrong** assumptions
  - System software verification requires a sound system hardware description

# A partial decoding net for the OMAP4460

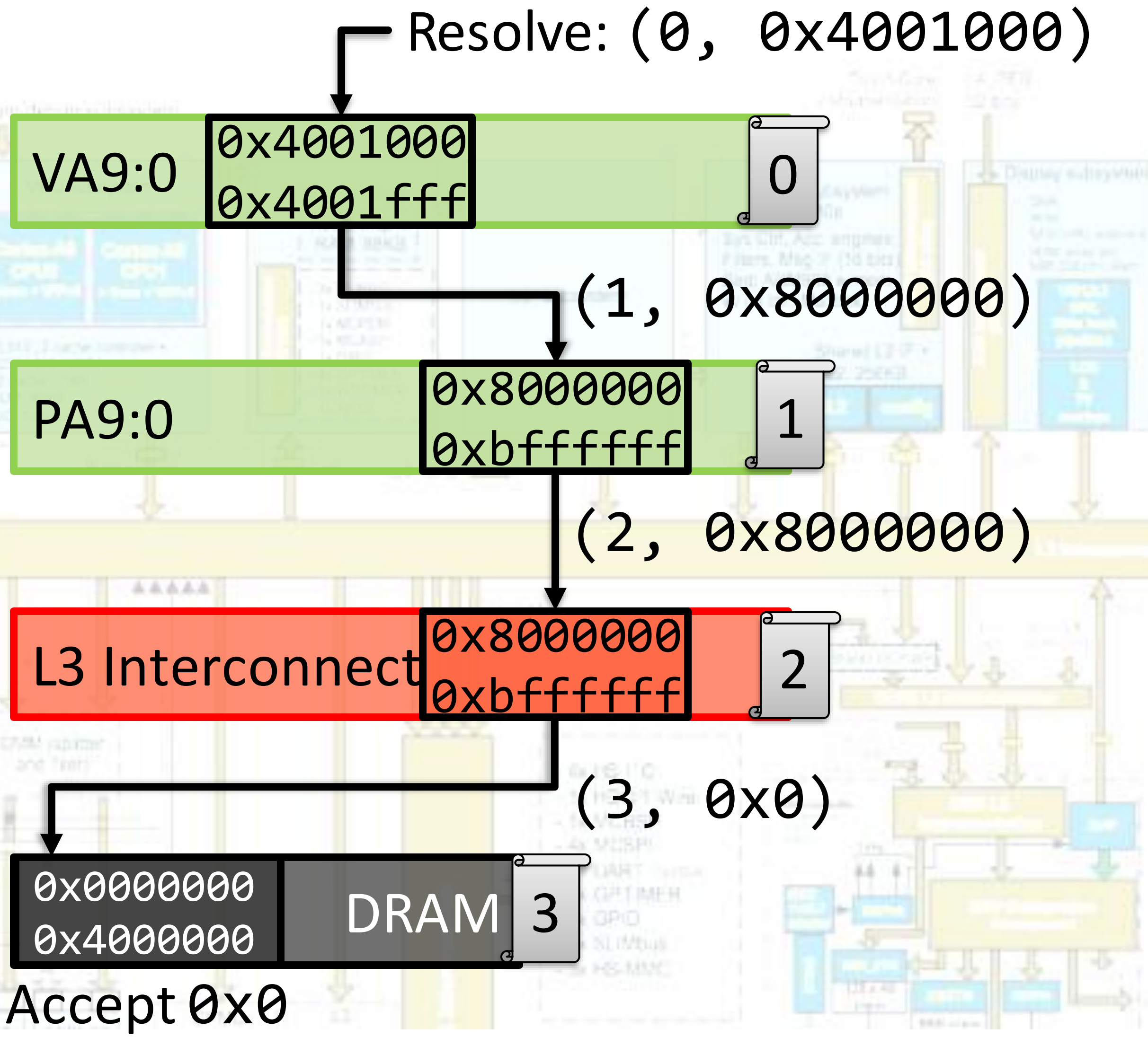


How does the Cortex A9:0
Access byte `0x0` of DRAM?

The model is a **decoding net,**
a directed graph

Hardware components ⇔ **nodes**

# Modelling the access to byte 0x0 of DRAM from an A9 core



Each node has a label

Resolve a **name** (node, address)

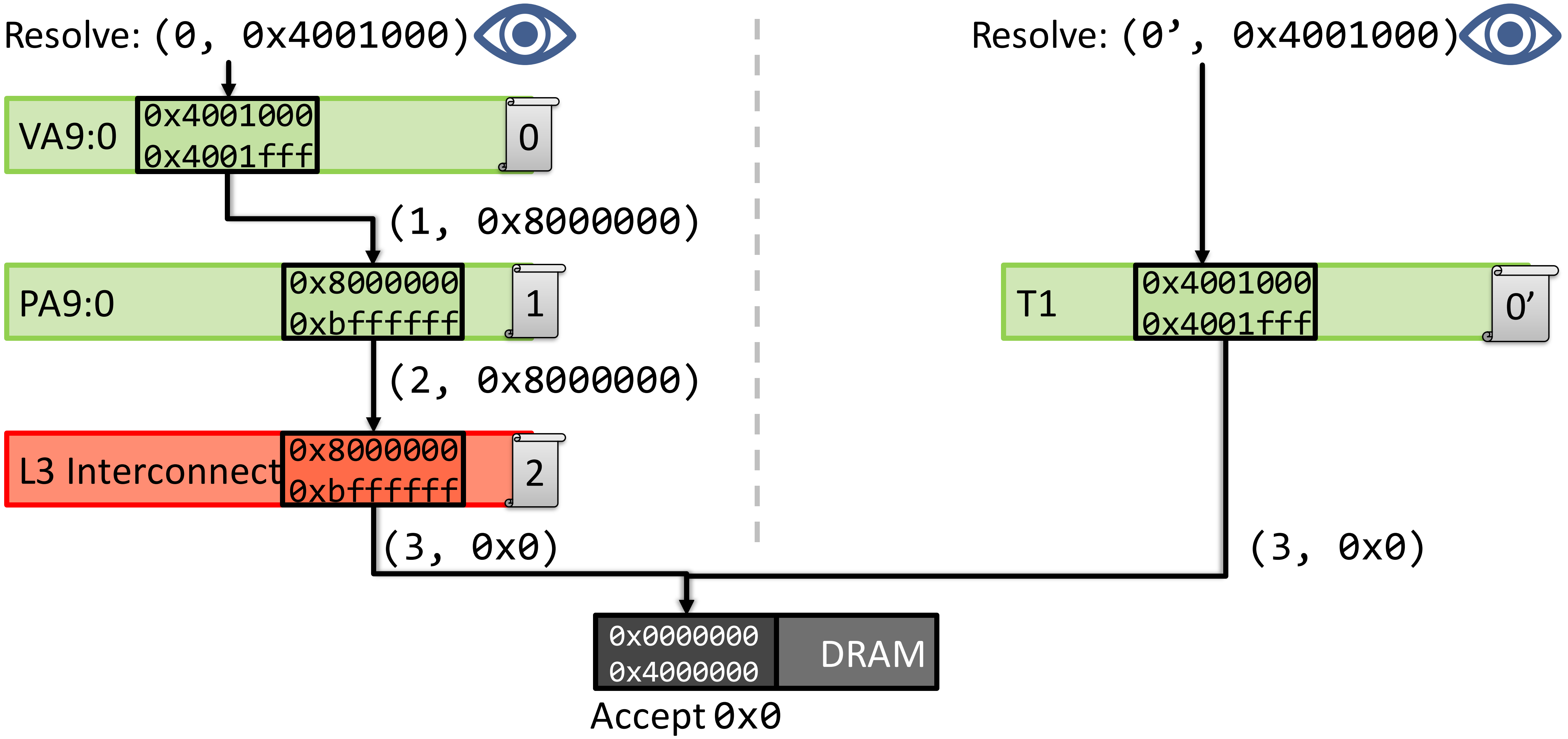
Model of one particular, static configuration state

Nodes have **two properties**:

$accept: \quad node \rightarrow \{\mathbb{N}\}$

$translate: \quad node \rightarrow \mathbb{N} \rightarrow \{name\}$

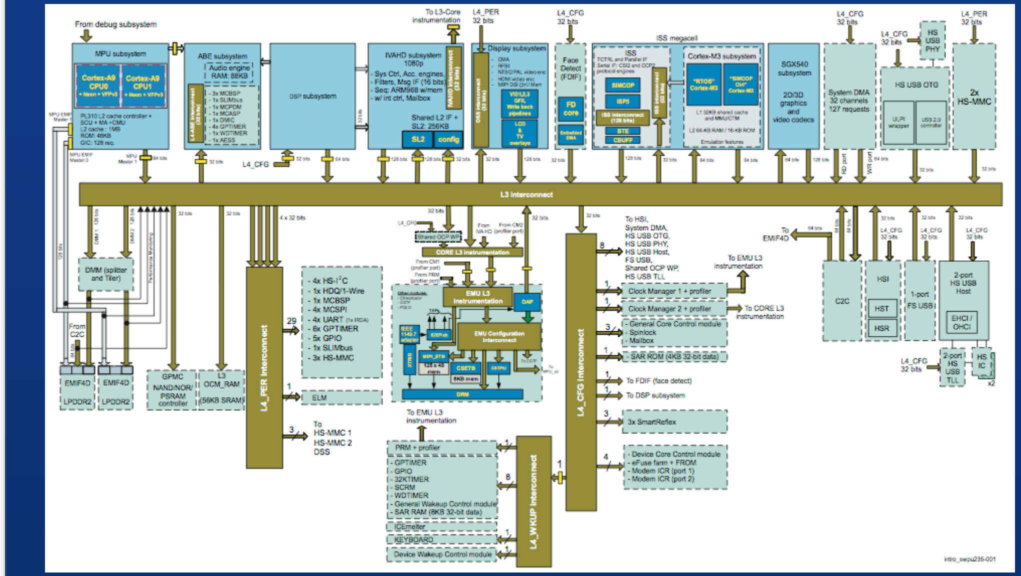
# Flattening using view equivalence preserving operations

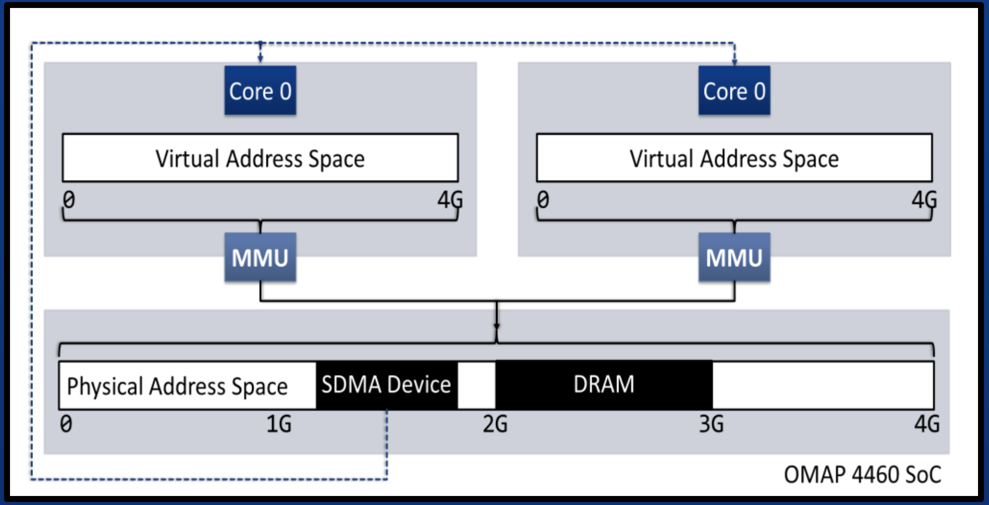


# Flattening using view equivalence preserving operations

Resolve: (0, 0x4001000)

Resolve: (0’, 0x4001000)

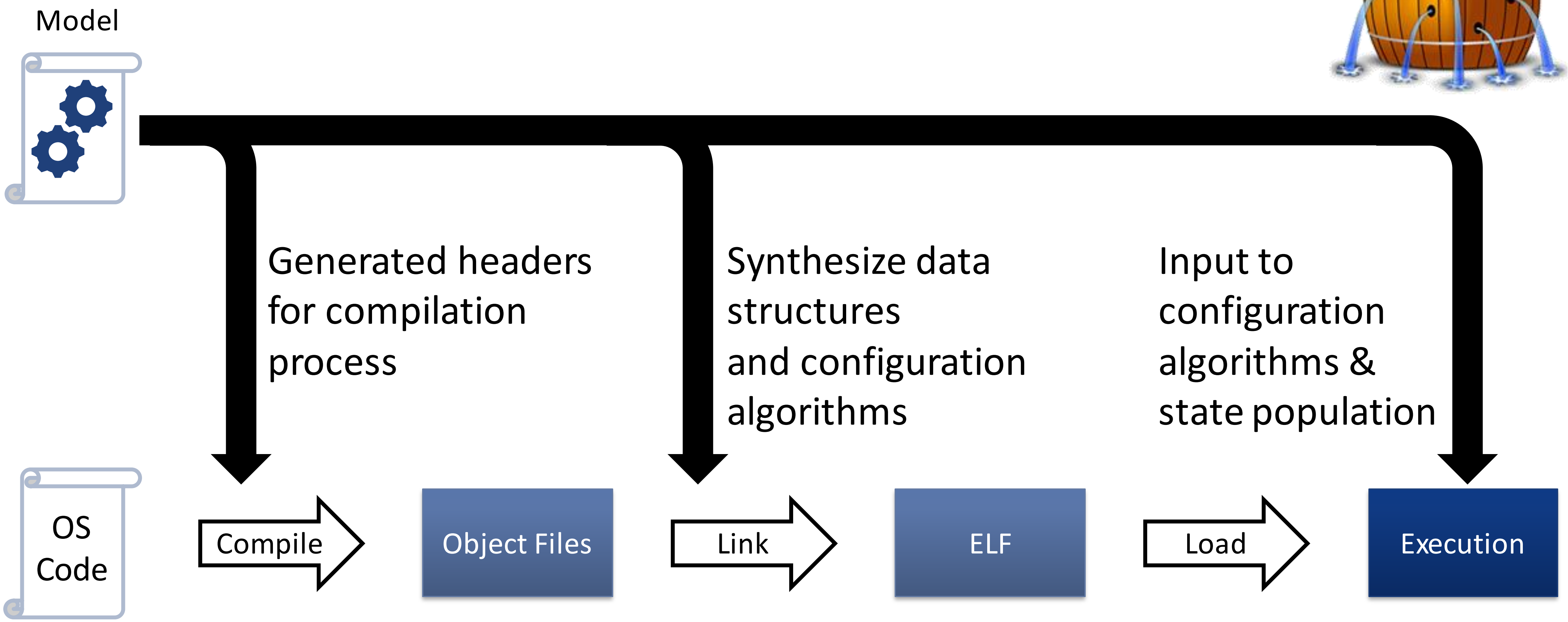




For ONE observer the flattened representation is equivalent to the textbook abstraction

Accept 0x0

# Ongoing work: Using model output at compile and run time



Model

Generated headers for compilation process

Synthesize data structures and configuration algorithms

Input to configuration algorithms & state population

OS Code

Compile

Object Files

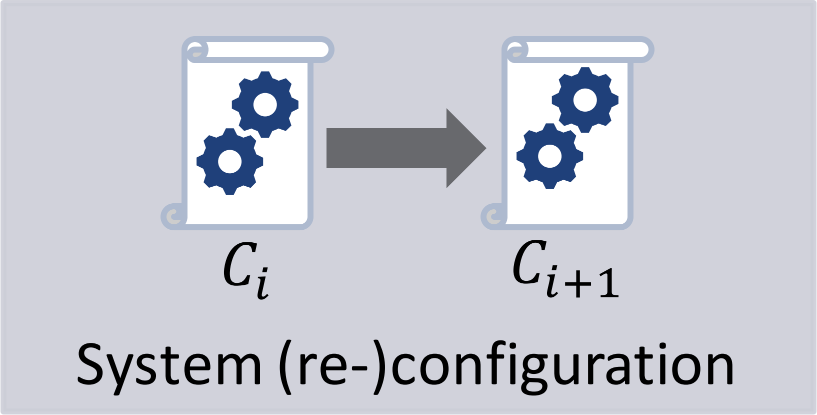
Link

ELF

Load

Execution

# Ongoing Work: Model applications



- Generate system configuration from the model:
  - Kernel page tables
  - Initial capabilities
- **Synthesize** configuration algorithms
- **Transition** between configurations without violation of **invariants**
- **Constraints** on memory accesses

# Future work: Model refinements

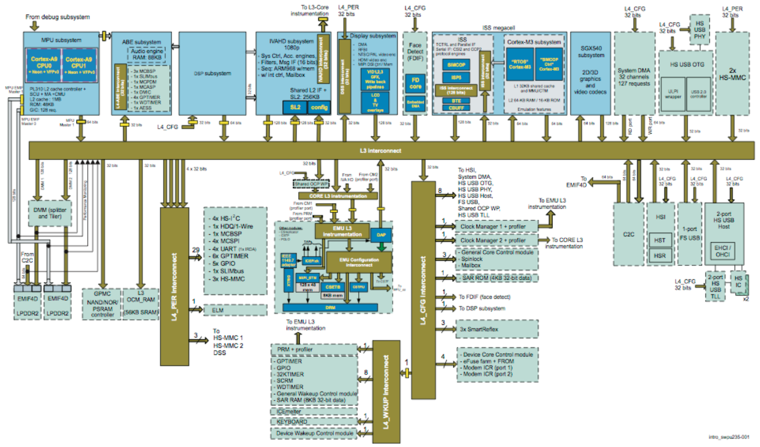
Distinction of Read/Write accesses

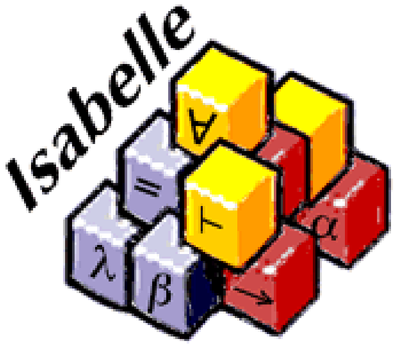
- Reads / writes have different semantics
- Write only / read only regions

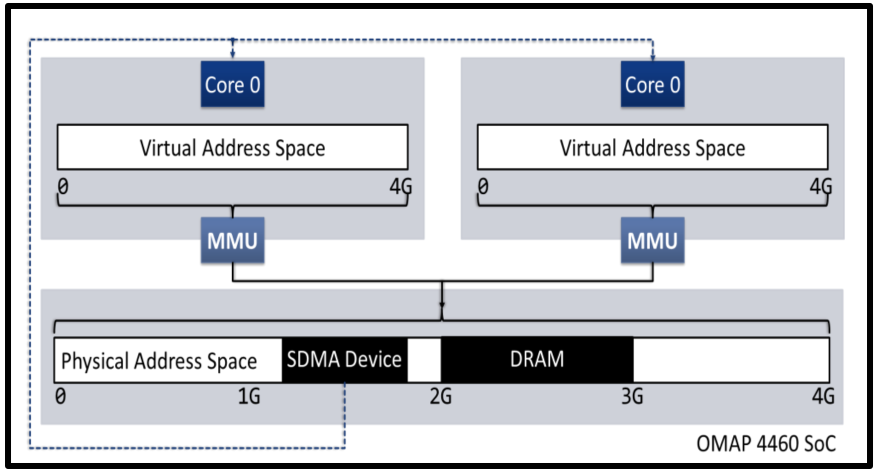
Expressing performance characteristics

- Basis for a performance model.
- Resource allocation & scheduling

# Summary







$$V_{A9:0}\ \textbf{is map}\ [20000_3/12\ \textbf{to}\ P_{A9:0}\ \textbf{at}\ 80000_3]$$

$$P_{A9:0}, P_{A9:1}\ \textbf{are map}\ [40138_3/12\ \textbf{to}\ \mathit{GPT}\ \textbf{at}\ 0]\ \textbf{over}\ L3$$

$$P_{DSP}\ \textbf{is map}\ [\mathtt{1d3e}_3/12\ \textbf{to}\ \mathit{GPT}\ \textbf{at}\ 0]\ \textbf{over}\ L3$$

$$V_{M3}, V_{M3}\ \textbf{are over}\ L1_{M3}$$

$$\mathit{RAM}_{M3}\ \textbf{is accept}\ [55020_3/16]$$

$$\mathit{ROM}_{M3}\ \textbf{is accept}\ [55000_3/14]$$

$$V_{A9:1}\ \textbf{is map}\ [20000_3/12\ \textbf{to}\ P_{A9:1}\ \textbf{at}\ 80000_3]$$

$$V_{DSP}\ \textbf{is over}\ P_{DSP}$$

$$L2_{M3}\ \textbf{is map}\ [0_{30}\ \textbf{to}\ L3\ \textbf{at}\ 80000_3]$$

$$L1_{M3}\ \textbf{is map}\ [0_{28}\ \textbf{to}\ \mathit{MIF}]$$

$$L4\ \textbf{is map}\ [49038_3/12\ \textbf{to}\ \mathit{GPT}\ \textbf{at}\ 0]$$

$$\mathit{GPT}\ \textbf{is accept}\ [0/12]$$

$$\mathit{MIF}\ \textbf{is map}\ [0 - \mathtt{5fffffff}\ \textbf{to}\ L2_{M3}, 55000_3/14\ \textbf{to}\ \mathit{RAM}_{M3}, 55020_3/16\ \textbf{to}\ \mathit{ROM}_{M3}]$$

$$L3\ \textbf{is map}\ [49000_3/24\ \textbf{to}\ L4\ \textbf{at}\ 40100_3, 55000_3/12\ \textbf{to}\ \mathit{MIF}]\ \textbf{accept}\ [80000_3/30]$$

Configuration