DATE 14
Design, Automation & Test in Europe
24-28 March, 2014 - Dresden, Germany
The European Event for Electronic System Design & Test

Coupling TDM NoC and DRAM Controller for Cost and Performance Optimization of Real-Time Systems

Manil Dev Gomony[1], Benny Akesson[2] and Kees Goossens[1]

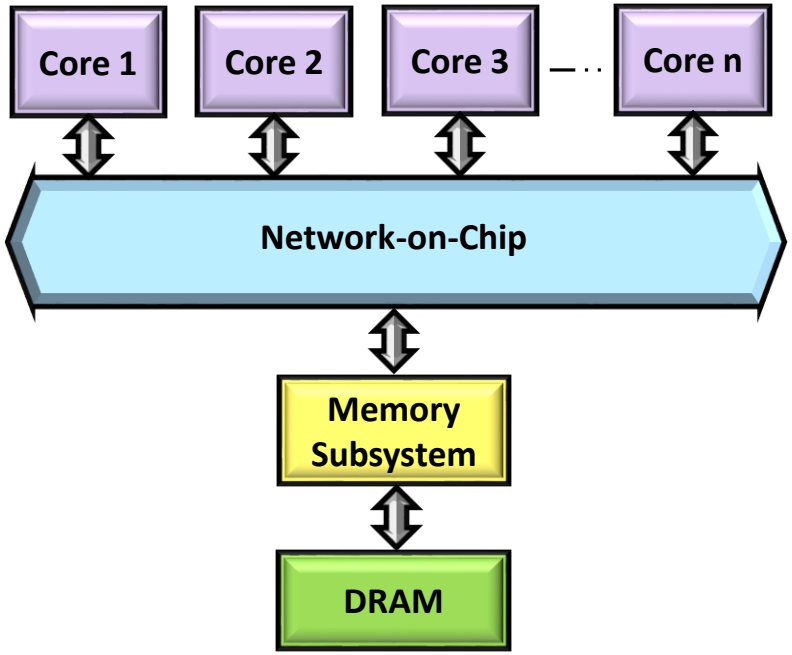
[1] Eindhoven University of Technology, The Netherlands

[2] Czech Technical University in Prague, Czech Republic

TU/e Technische Universiteit Eindhoven University of Technology
ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE
T-CREST
BENEFIC
ARTEMIS
COBRA – CA104
NEST
OPENES

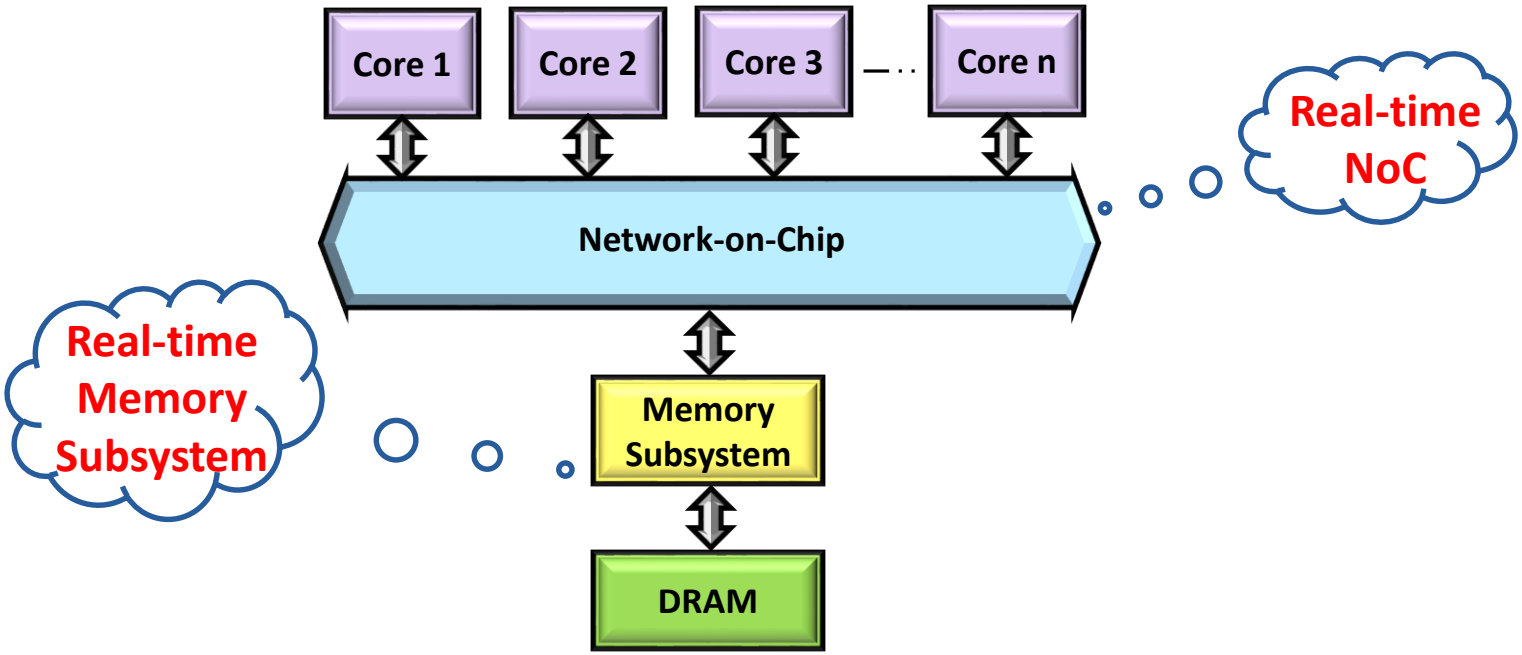
Multi-processor platforms

- **Network-on-Chip is used as an interconnect**
  - ***For scalability as opposed to bus-based interconnects***
- **Main memory (DRAM) is a shared resource**
  - ***For cost and communication reasons***



In real-time systems

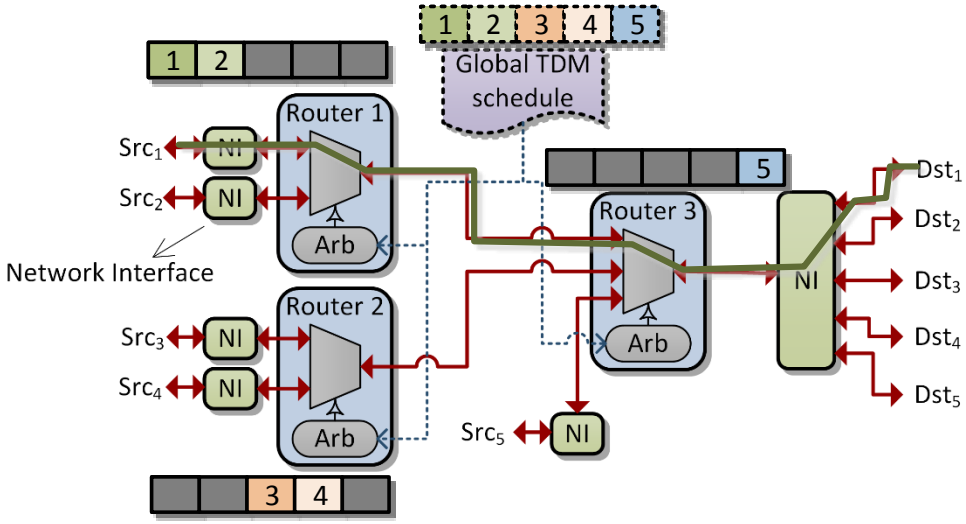
- **Memory clients comes with real-time requirements on memory bandwidth and/or latency**
  - ***Real-time NoCs and Memory Subsystems***



- **Minimize cost and maximize performance!**

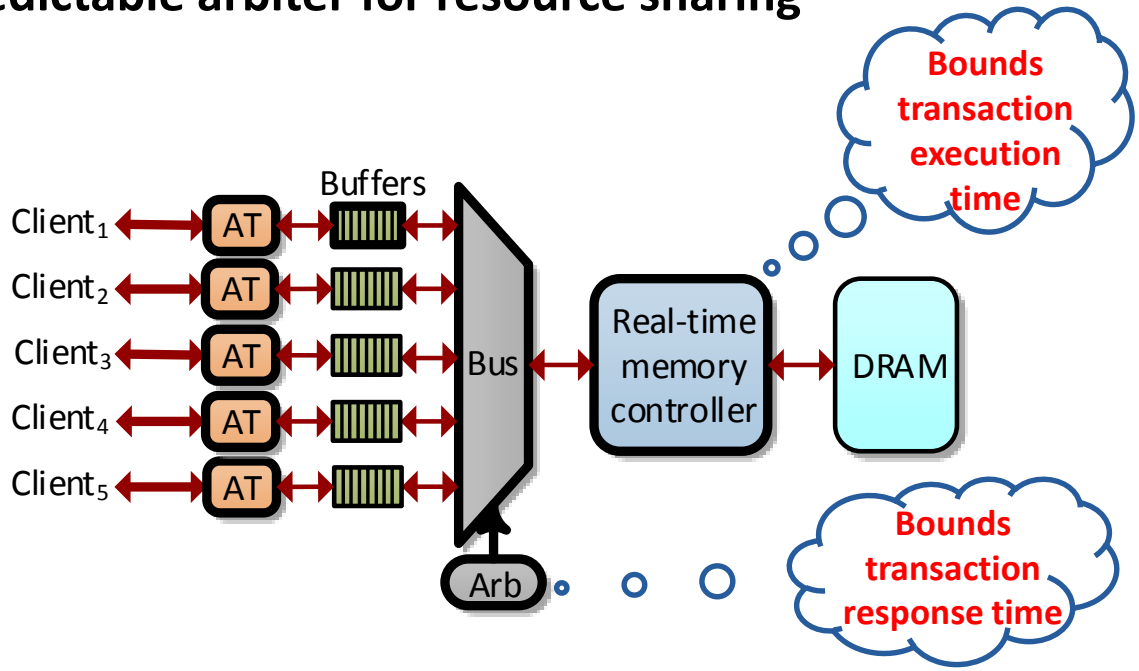
Real-time NoC

- **Statically scheduled Time Division Multiplexed (TDM) NoC**
  - ***Single global TDM schedule***
- **Provide end-to-end guarantees on latency and bandwidth between a source and destination**
  - ***Dedicated virtual channel/circuit***



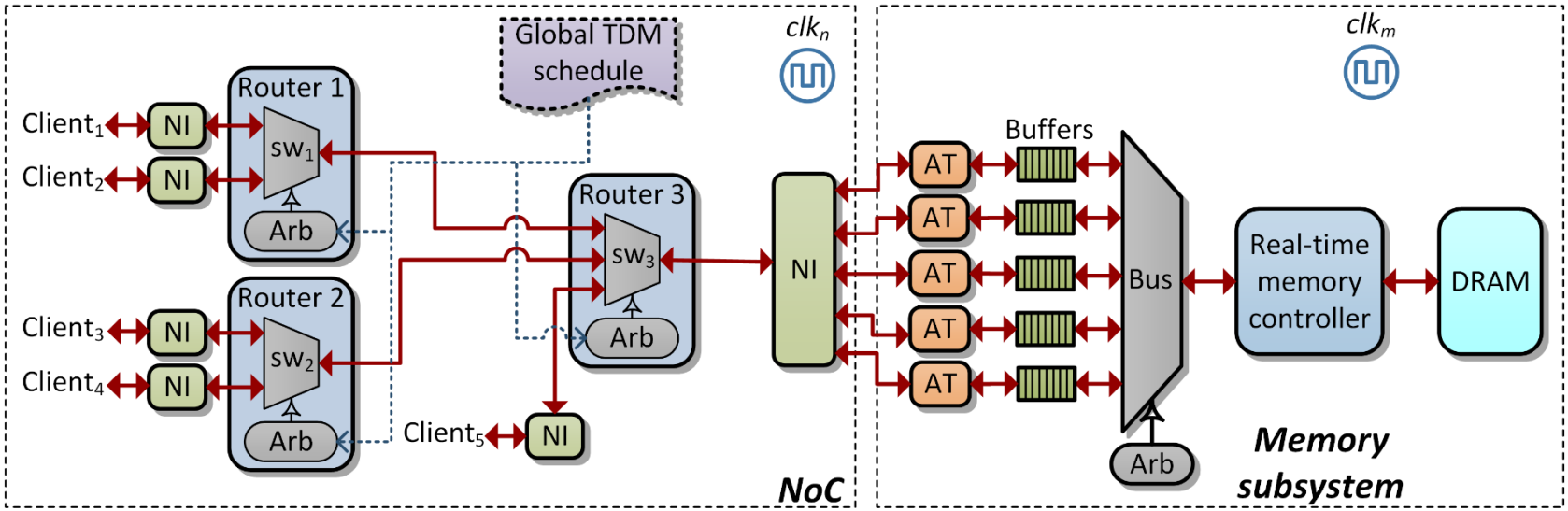
Real-time memory subsystems

- **Provide guarantees on memory bandwidth and latency to a memory client**
  - ***Real-time memory controller*: Fixed set of memory access parameters (burst size, page-policy)**
    - **Atomizers (AT) split larger transactions to smaller sized *service units***
  - **Predictable arbiter for resource sharing**



Motivation

- **Existing *TDM NoCs* and real-time *memory subsystems* are optimized independently**



- **Multiple arbitration points!**
- **Reducing to a single arbitration**
  - ***Reduces worst-case latency***
  - ***Destination NI needs a single output port***

Our contributions

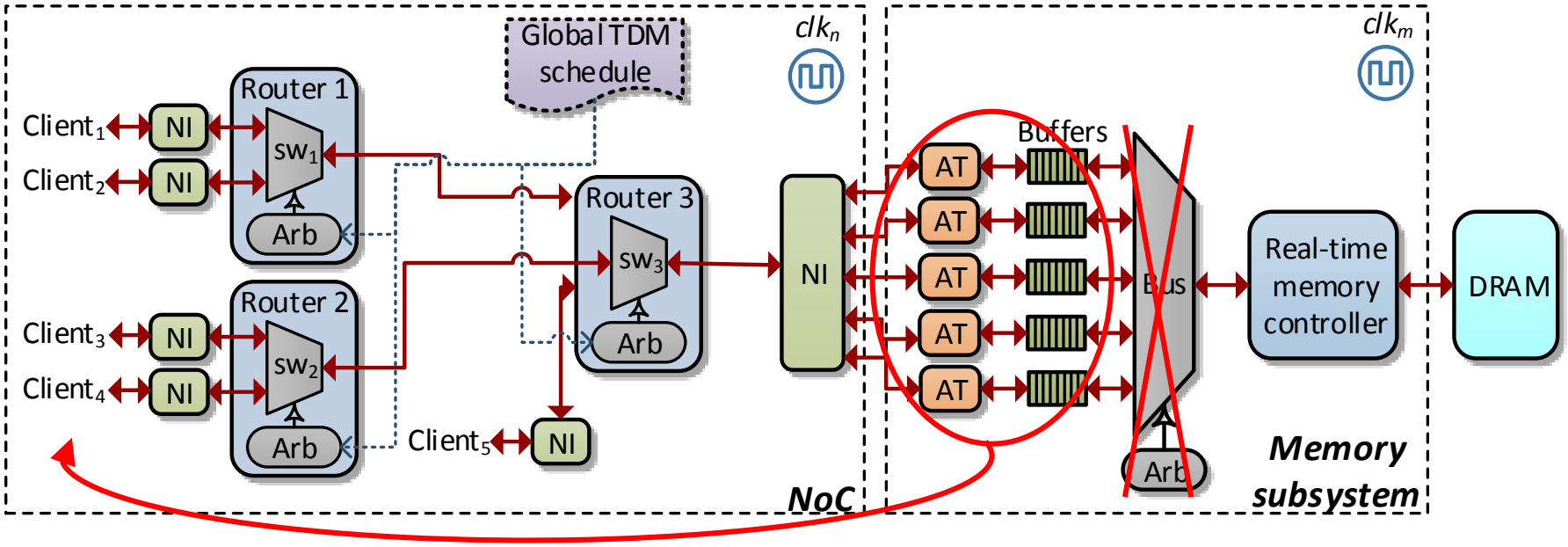
- **Novel methodology to couple any existing TDM NoC with a real-time memory subsystem**
  - ***Compute the different NoC configurations for minimal area and/or power consumption***
- **Trade-off between area and power consumption**
  - ***For different NoCs and memory devices***
- **Comparison of coupled and decoupled architectures by synthesizing the designs**
  - ***In 40nm technology***

Outline

- Introduction
- **Coupling TDM NoC and memory subsystem**
- **Dimensioning the NoC**
- **Experiments**
- **Conclusions**

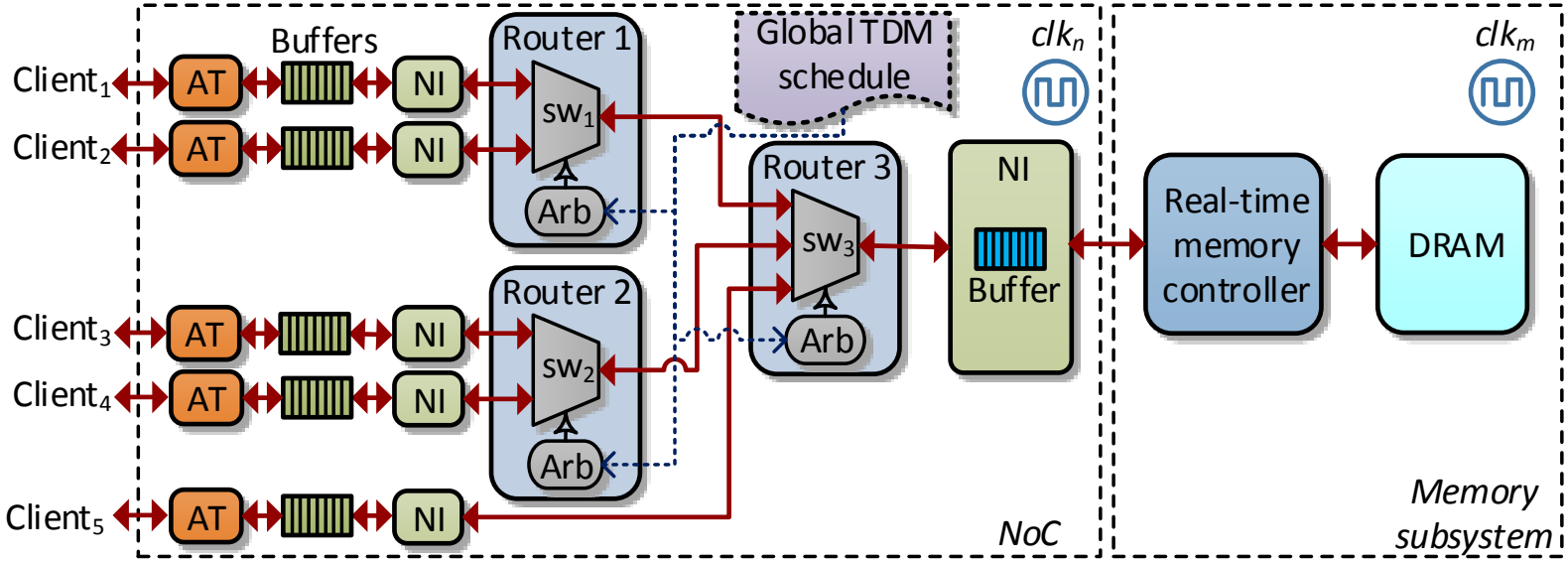
Coupling NoC and memory subsystem

- **Remove the bus-based arbitration in the memory subsystem**
- **Move atomizers and decoupling buffers to the client side**
- **Perform memory arbitration in the NoC**
  - ***Lose flexibility in selecting different arbitration for memory!***



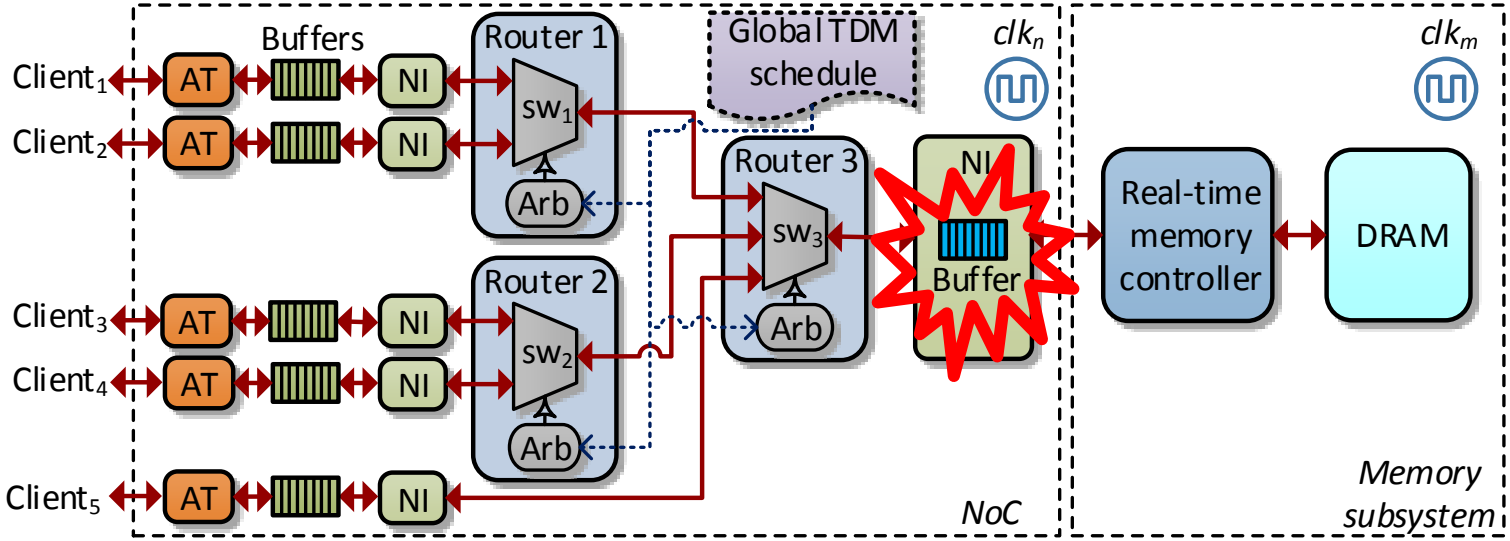
Need to address..

- **Different clock domains, transaction granularities**
  - ***What should be the buffer size?***
  - ***How to select NoC interface width and operating frequency?***
  - ***How to guarantee real-time performance to clients?***



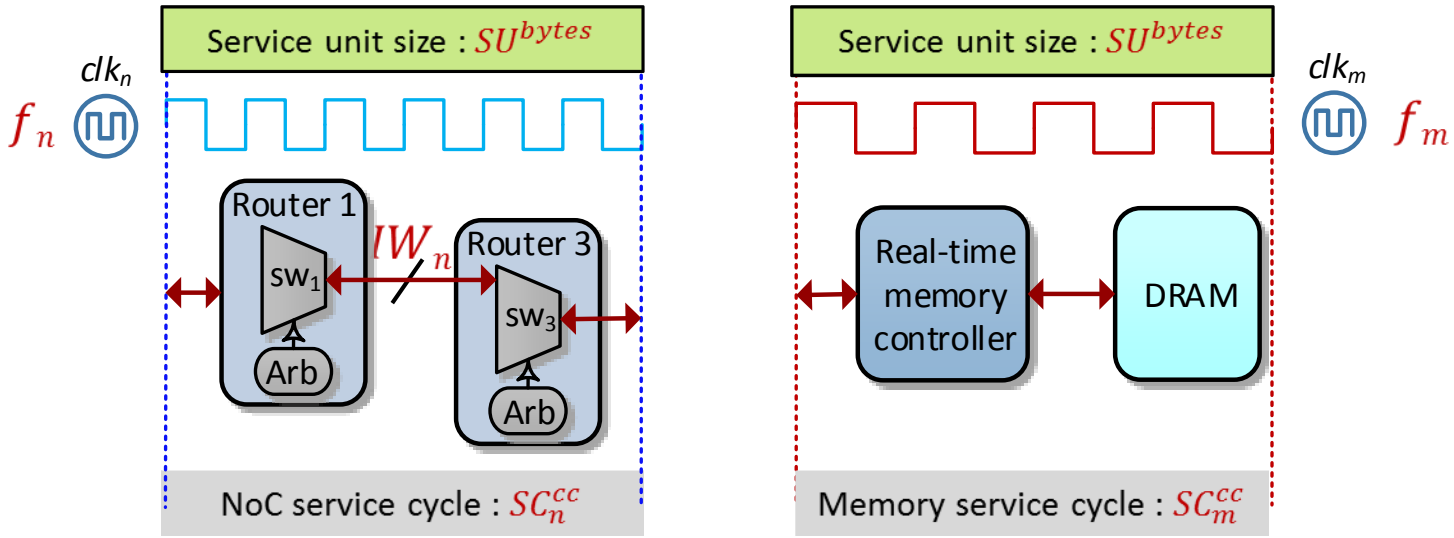
NoC configuration

- **NoC transports *flits* and the memory controller executes service units**
  - ***Configure NoC flit size = service unit size***
- **Buffer of size equal to the service unit is required at the destination NI**
  - ***Complete service unit need to be buffered***



Bandwidth matching

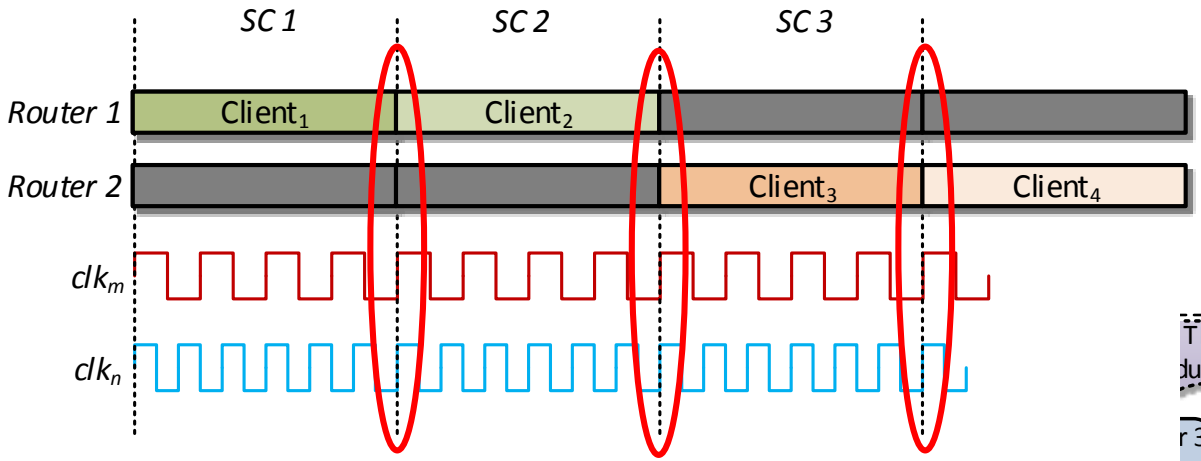
- **NoC link bandwidth must be same as the memory subsystem in a service cycle**

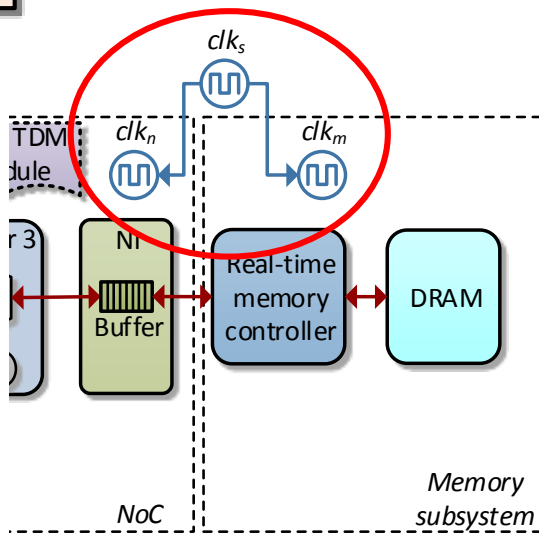


$$f_n \times \frac{IW_n}{8} \times \frac{SC_n^{cc} - \delta_{ov}}{SC_n^{cc}} = f_m \times \frac{SU^{bytes}}{SC_m^{cc}}$$

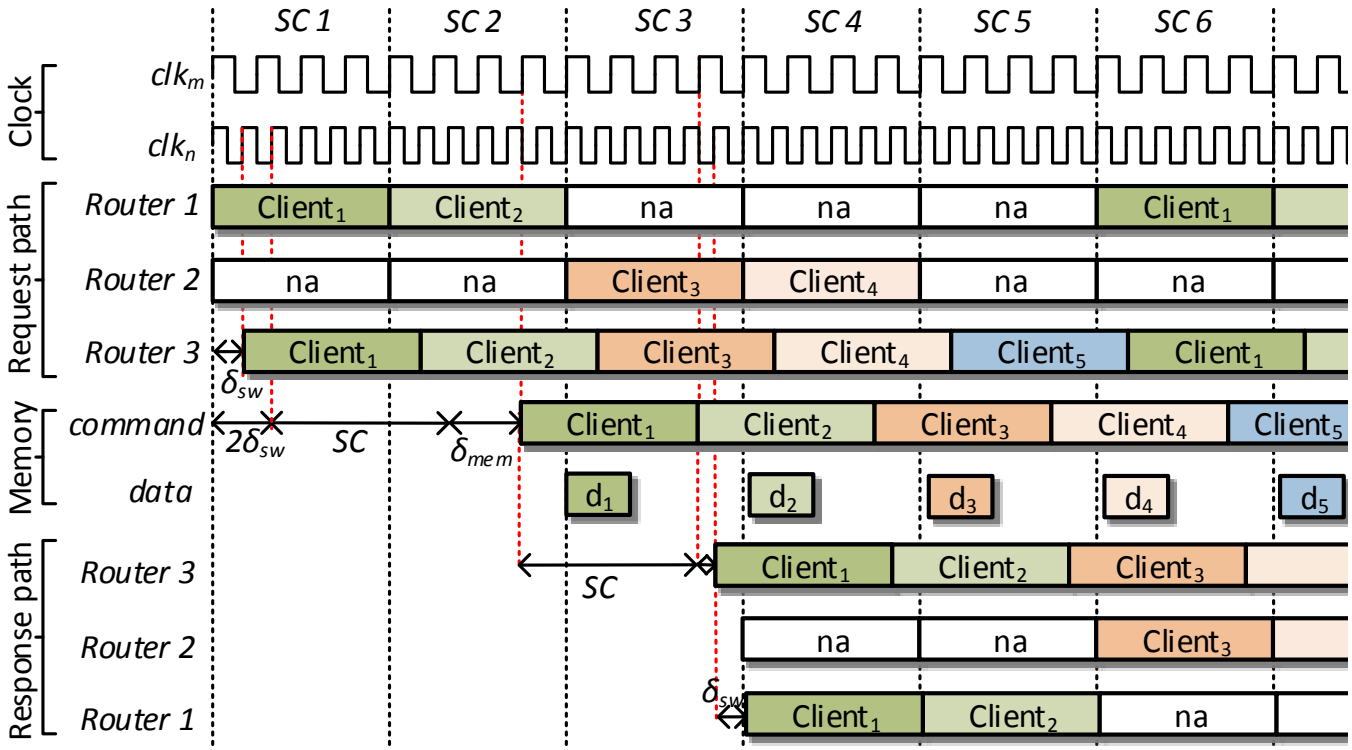
Clock alignment

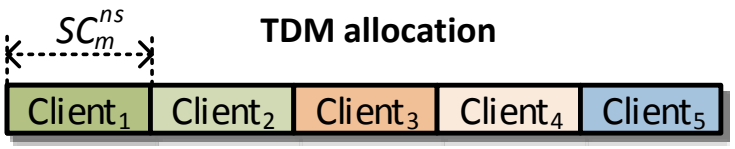
- **Clock edges of NoC and memory subsystem must be aligned at the service cycle boundaries**
  - ***Single clock source***

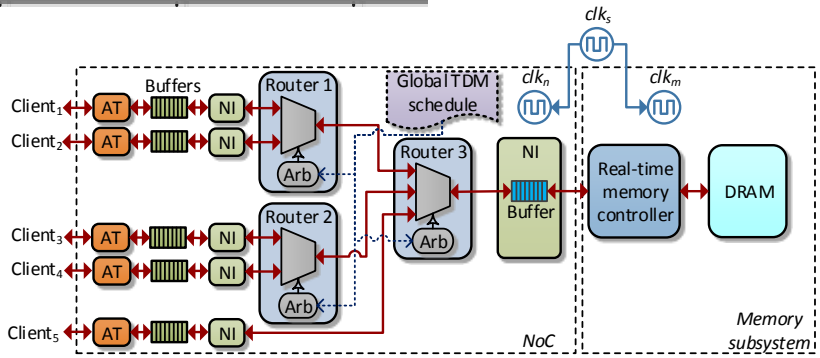




Coupled architecture - operation







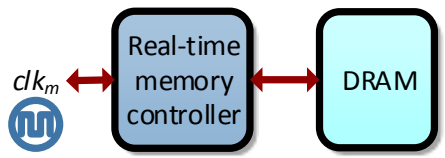
Outline

- Introduction
- Coupling TDM NoC and memory subsystem
- **Dimensioning the NoC**
- **Experiments**
- **Conclusions**

Design parameters

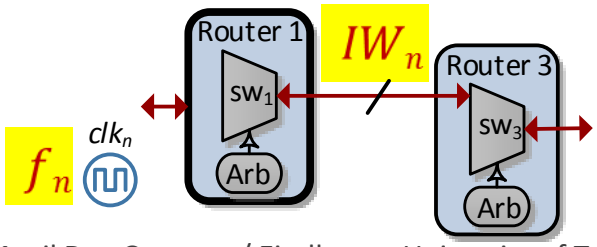
- The **service cycle duration** of a given memory device can be **computed** using state-of-the art methods

DRAM device	Memory frequency (f_m)
LPDDR-266	133
LPDDR-416	208
LPDDR2-667	333
LPDDR2-1066	533
DDR3-800	400
DDR3-1600	800



Service unit size (SU^{bytes})	Service cycle (SC_m^{cc})
16	13
32	13
64	19
128	35

- The **interface width** and **operating frequency** of the NoC need to be selected based on area vs. power trade-off



NoC dimensioning

- Given a memory device with frequency f_m, service unit size of SU^{bytes} with service cycle SC_m^{cc}
- Determine all (f_n, IW_n) combinations which satisfies the hardware constraints

  Step 1: Compute all possible values of f_n that are integer multiples and common fractions of f_m

  Step 2: Select the values of f_n such that the clocks will be aligned at the boundaries of the service cycles

$$(SC_m^{cc} \times f_n) \; mod \; f_m = 0$$

  Step 3: Compute the values of IW_n corresponding to the different f_n using the bandwidth matching equation

$$f_n \times \frac{IW_n}{8} \times \frac{SC_n^{cc} - \delta_{ov}}{SC_n^{cc}} = f_m \times \frac{SU^{bytes}}{SC_m^{cc}}$$

Outline

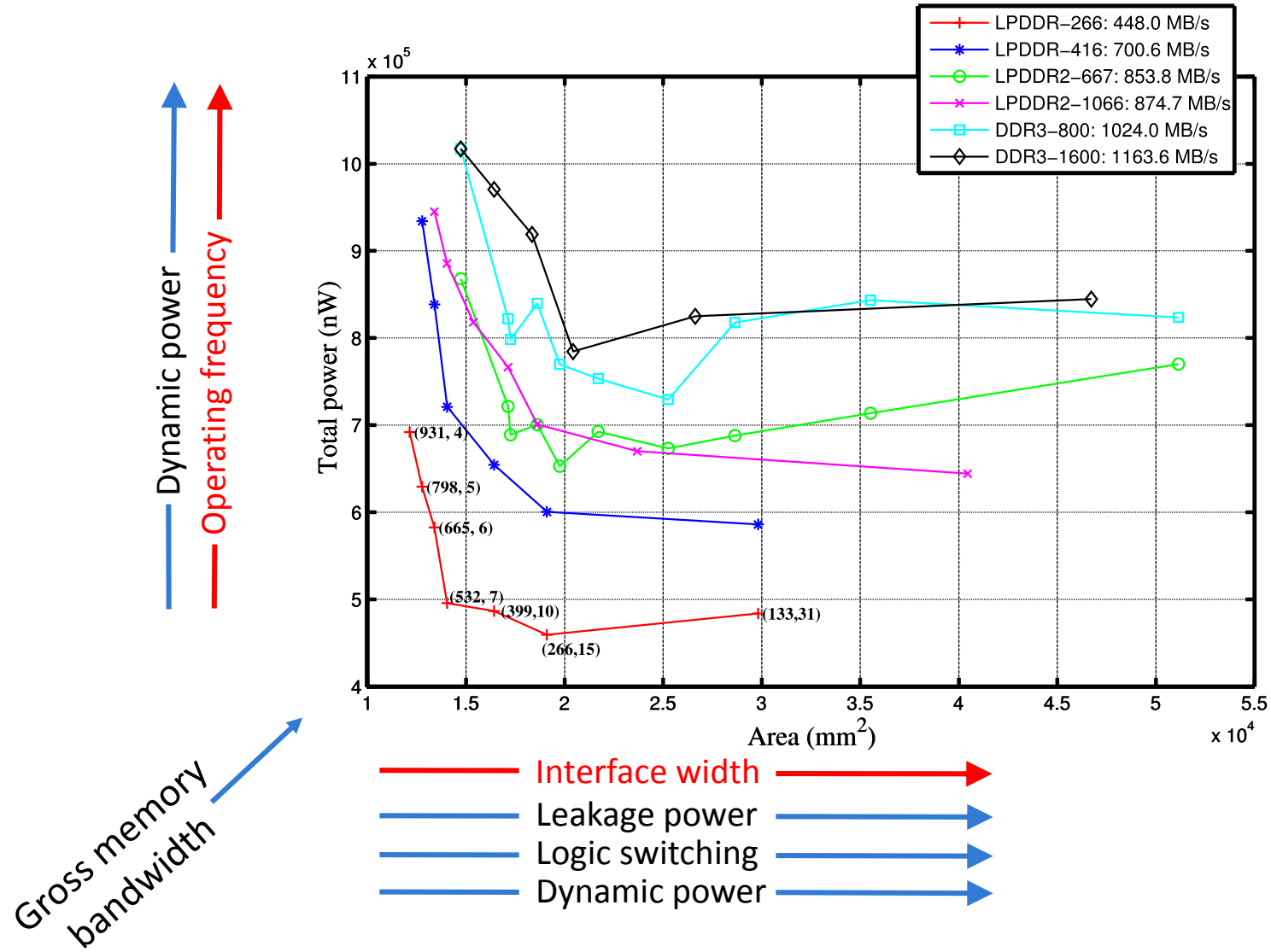
- Introduction
- Coupling TDM NoC and memory subsystem
- Dimensioning of TDM NoC
- **Experiments**
- **Conclusions**

Experimental setup

- **RTL-level implementations of**
  - ***Router and NI of two different TDM NoC types***
    - ***Packet switched : Aelite***
    - ***Circuit switched : Daelite***
  - ***TDM arbiter***
  - ***Bus using the Device Transaction Level (DTL) protocol***
    - ***Comparable to AXI and OCP protocols***

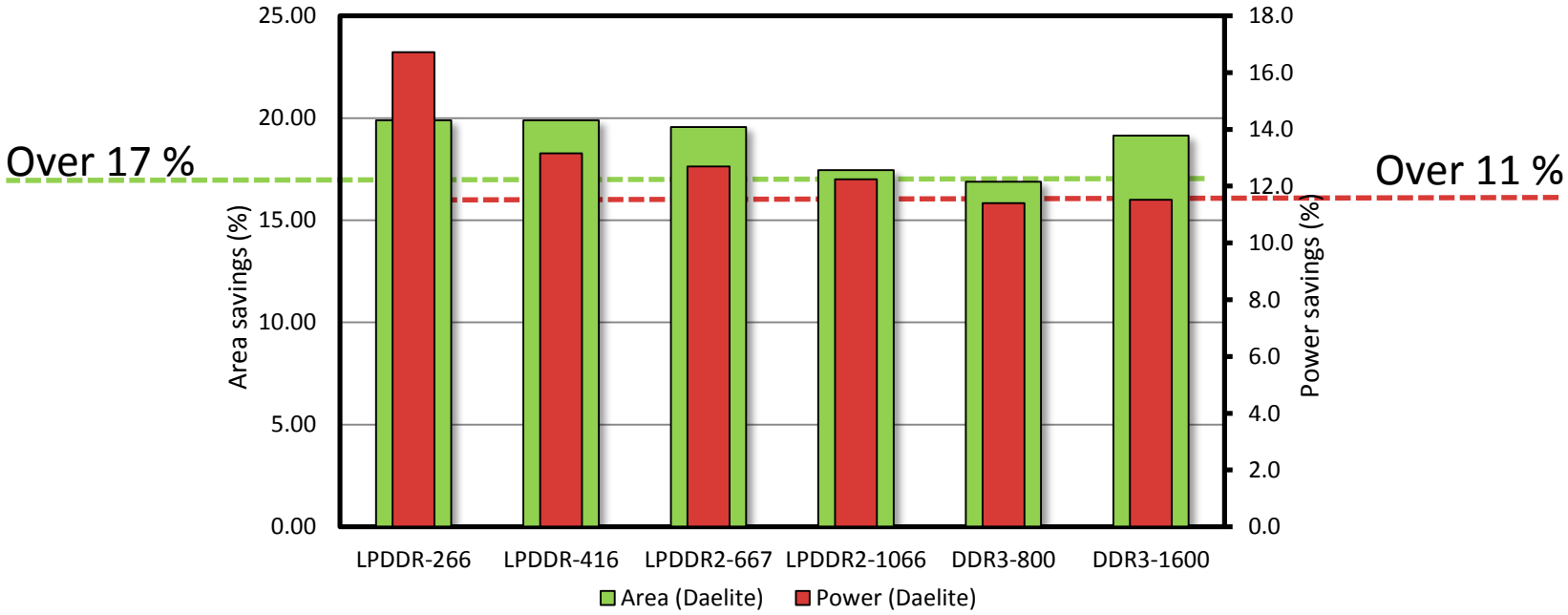
- **Power/area estimation using the Cadence Encounter RTL compiler**
  - ***40 nm nominal Vt CMOS standard cell library***

Area vs. power trade-off



Area/power savings

- **16 memory clients → 16 ports in destination NI**
- **Four-stage NoC tree consisting of 15 routers and NIs**
- **Service unit size 64 B → NI buffer size**



Worst-case latency savings

- **16 clients with same bandwidth allocated to all clients**
  - **Assuming the same TDM allocation for the NoC in both architectures**

Memory	**NoC frequency (MHz)**	**Interface width (bits)**	***Worst-case latency (ns)***	
			Decoupled	***Coupled***
LPDDR-266	266.0	15	4.81	2.65
LPDDR-416	416.0	15	3.08	1.70
LPDDR2-667	399.6	16	2.51	1.38
LPDDR2-1066	355.3	22	2.45	1.35
DDR3-800	480.0	23	2.11	1.17
DDR3-1600	400.0	17	1.83	1.00

Over 44 %

Outline

- Introduction
- Coupling TDM NoC and memory subsystem
- Dimensioning of TDM NoC
- Experiments
- **Conclusions**

Conclusions

- **Existing NoCs with a global TDM schedule can be coupled with a real-time memory controller by**
  - ***Configuring NoC flit size equal to the service unit size***
  - ***Selecting NI buffer size equal to the service unit size***
  - ***Using a single clock source***
- **We proposed a methodology for computing the NoC parameters for power vs. area trade-off**
- **For a system with up to 16 clients, coupling the NoC with the memory subsystem saves over**
  - ***17% area***
  - ***11% power consumption***
  - ***44% worst-case latency***
- ***Give up flexibility in selecting different arbiters for memory!***

Questions?