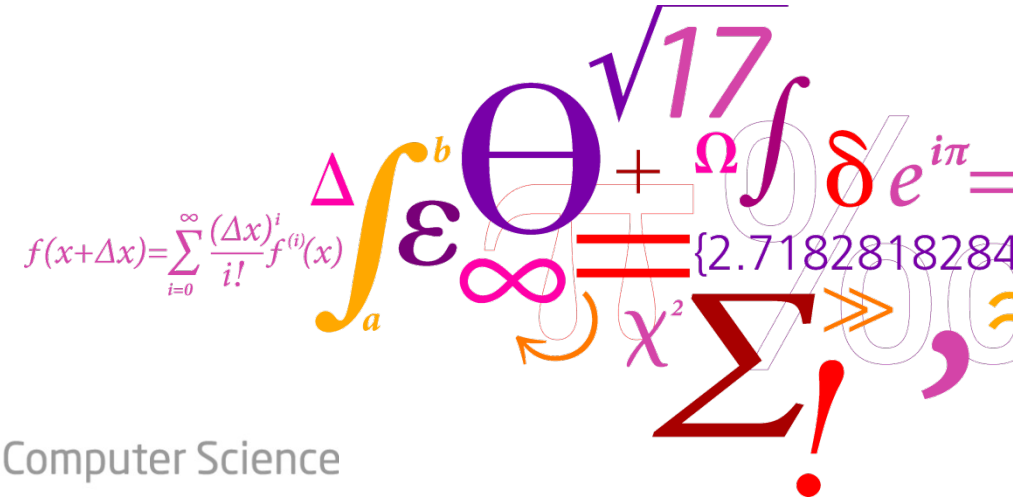
Networks-on-Chip for Hard Real-time Systems

Jens Sparsø
jspa@dtu.dk
www.compute.dtu.dk



DTU Compute
Department of Applied Mathematics and Computer Science

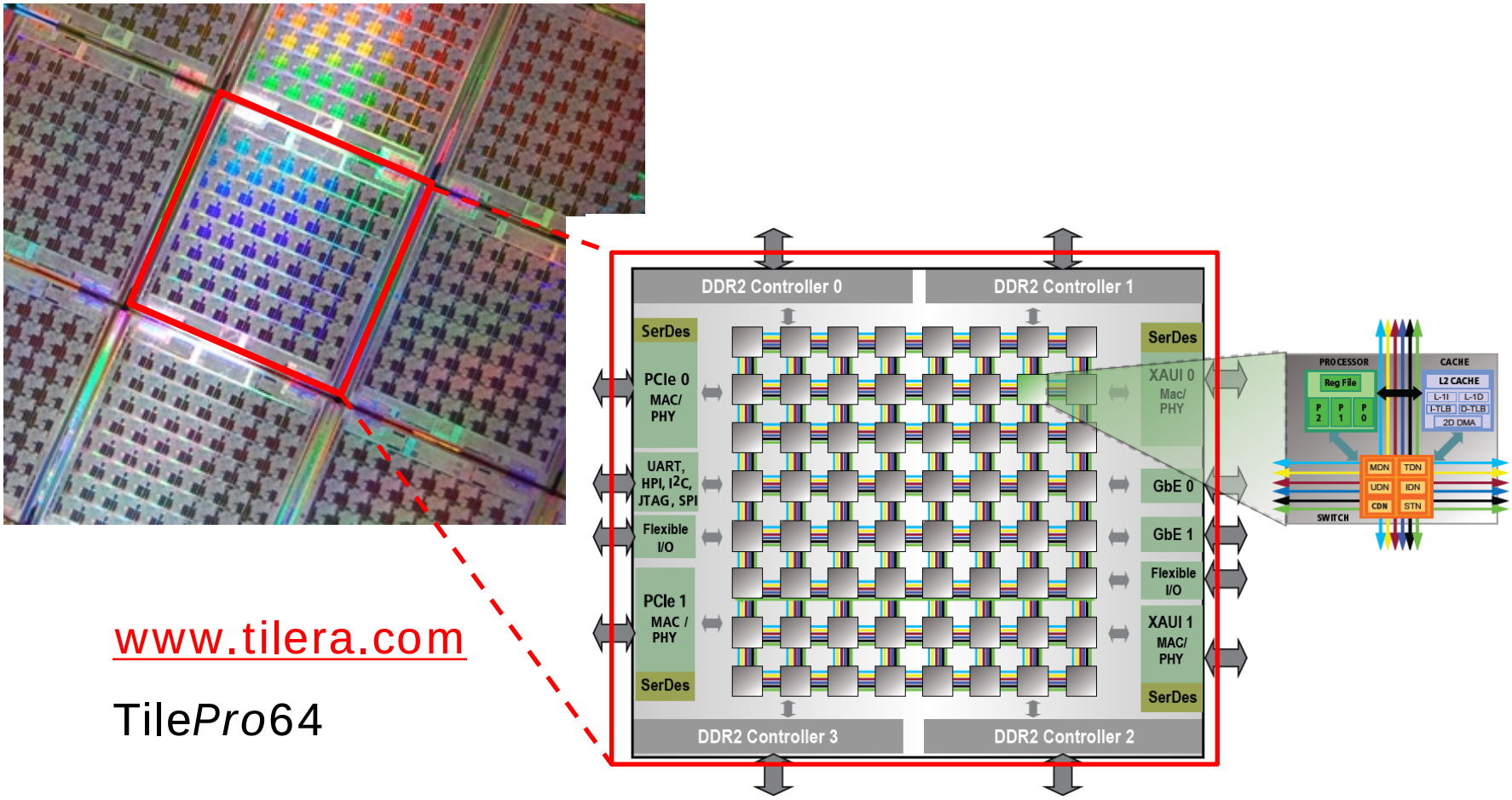
Outline

1. **Introduction**
   - Multi-core platforms and networks on chip
2. **NOC tutorial**
   - Motivation and basics
3. **NOCs for real time systems**
   - End-to-end (virtual) circuits
4. **The T-CREST NOC**
   - Scheduling of traffic
   - HW design (work in progress)
     - Router node
     - Network interface
5. **Conclusion**

Today: Everything is multi-core and NOC-based

- **General purpose computing**
  - Chip Multi-Processors **(CMP)**
  - Speed is convenience (*no guarantees*)
  - Identical processors (*homogeneous*)
- **Embedded systems**
  - Multi-Processor Systems-on-Chip **(MPSoC)**
  - *Application specific* systems
  - Often *heterogeneous* processors (inc. HW-acc., DSP)
  - *Real time* requirements (hard or soft)
- **Architecture of CMP's and MPSoC's is converging:**
  - "Cores" communicating through a
  - packet switched network-on-chip (NOC)

A typical chip photo (CMP)

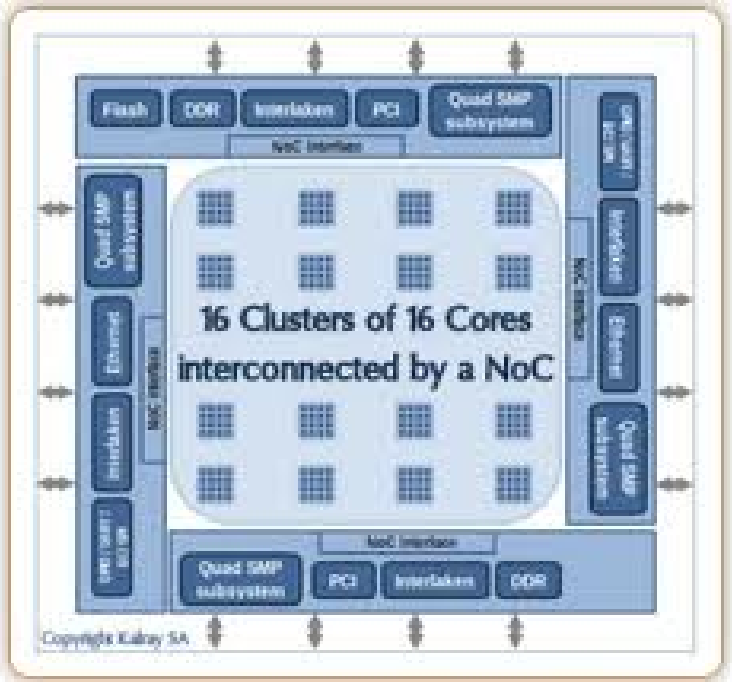


www.tilera.com

Tile*Pro*64

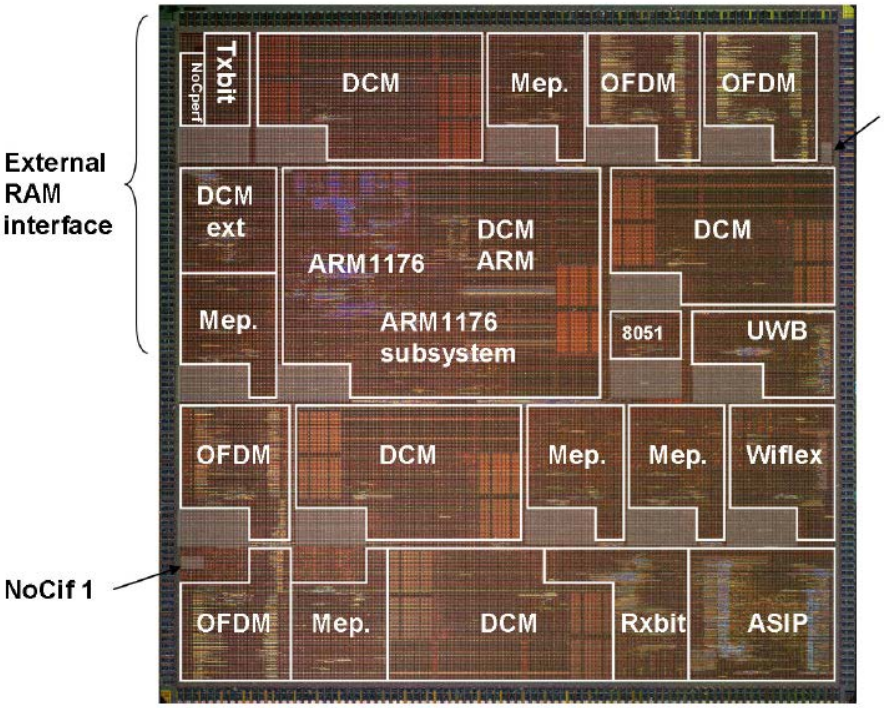
Kalray MPPA (Multi Purpose Processor Array)

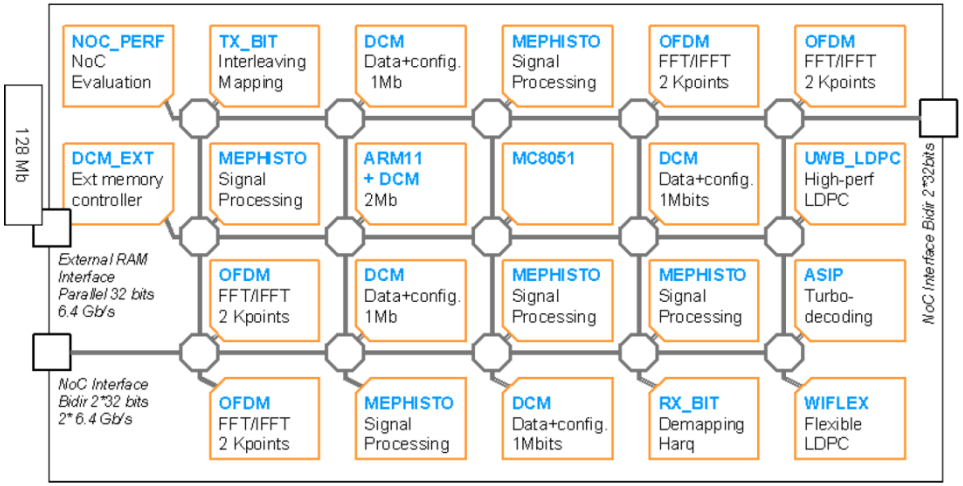
www.kalray.eu



- 256 VLIW cores per chip
- 16 clusters of 16 cores,
- High bandwidth Network-on-Chip

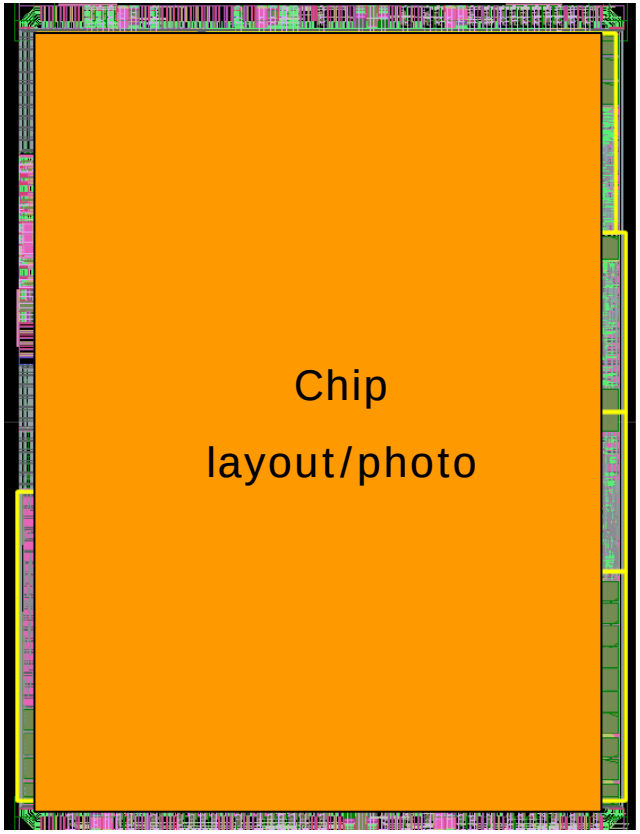
A typical chip photo (MPSoC)





Clermidy, F. and Bernard, C. and Lemaire, R. and Martin, J. and Miro-Panades, I. and Thonnart, Y. and Vivet, P. and Wehn, N., **A 477mW NoC-based digital baseband for MIMO 4G SDR**, Proc. IEEE International Solid-State Circuits Conference, pp. 278-279, **2012**

A typical chip photo (MPSoC)



Keynote at:

6th ACM/IEEE International Symposium on Networks-on-Chip May 7-9, 2012 (hosted by DTU).

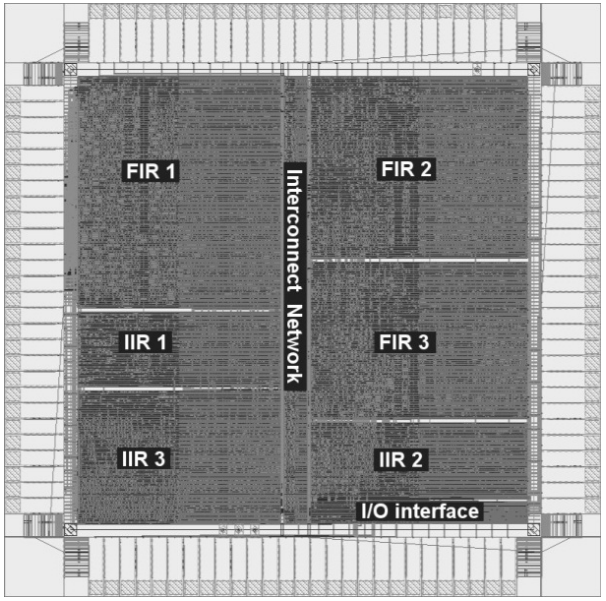
M. Balsby:

“Oticon hearing aids - highly optimized multi-processor systems-on-chip”

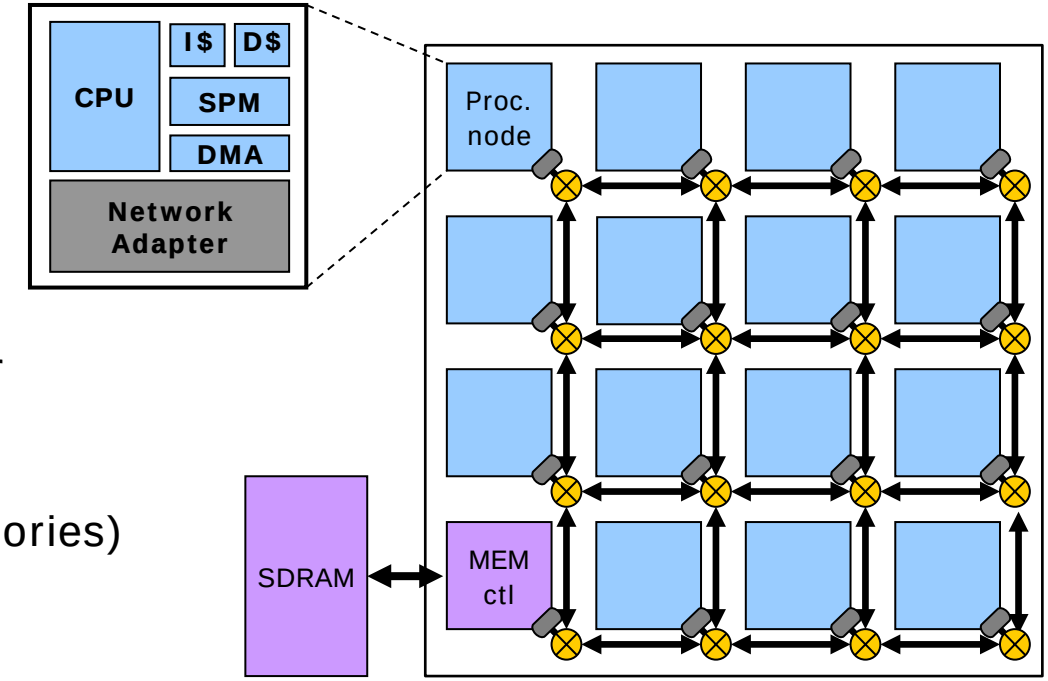
Early work on networks on chip

- Some key papers:
  - W.J. Dally and B. Towles, “Route Packets, Not Wires: On-Chip Interconnection Networks”, DAC 2001
  - L. Benini and G. De Micheli, "Networks on chip: a new SoC paradigm", IEEE Computer, vol. 35, no. 1, Jan. 2002
- Roots in:
  - Interconnection networks for parallel computers built from single-chip microprocessors (1980-1990)
    - Caltech Mosaic, Torus routing chip
    - Intel Hypercube …
- At DTU

  Ö. Paker, J. Sparsø, N. Haandbæk, M. Isager, L.S. Nielsen, “A Heterogeneous Multiprocessor Architecture for Low-Power Audio Signal Processing Applications”, 2001
  - 6 small instruction set processors
  - Interconnection network
  - 0.25 µm CMOS std. cells 520 K transistors



T-CREST platform

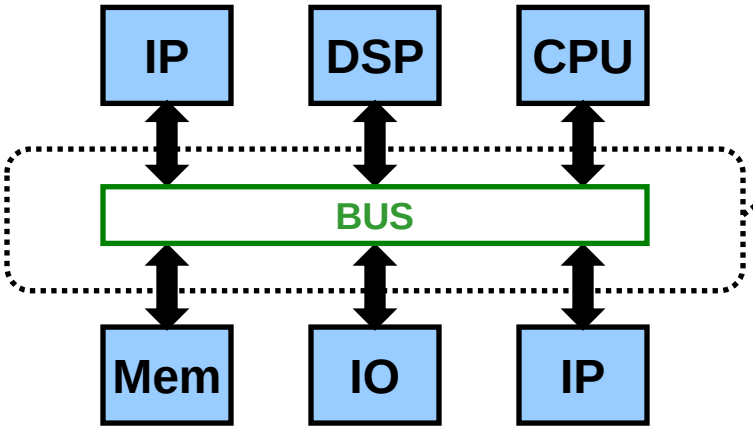


- Processor node:
  - Time predictable processor (Patmos)
  - Local memories (caches, scratch-pad memories)
  - DMA-controllers
- Memory controller node:
  - Time-predictable access to a shared off-chip SDRAM
- Network-on-chip:
  1. Communication between processor nodes
     Scratch-pad (SPM) to scratch-pad driven by DMA
  2. Communication between processor nodes and memory (controller)
     Scatch-pad to memory-controller

Outline

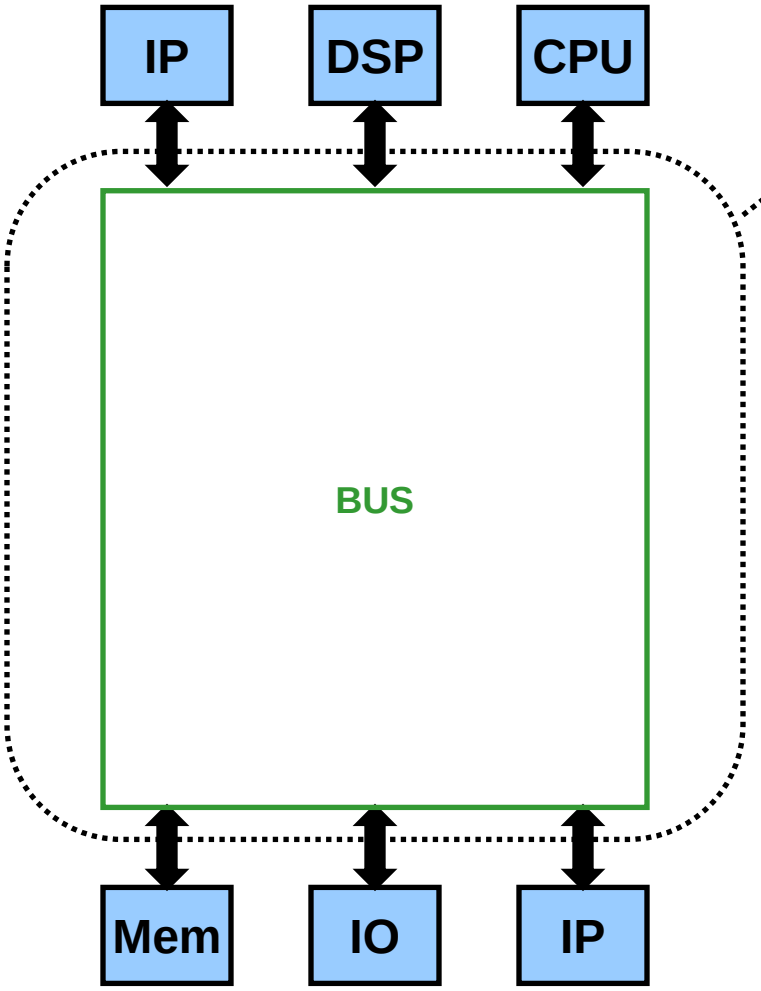
1. **Introduction**
   - Multi-core platforms and networks on chip
2. **NOC tutorial**
   - Motivation and basics
3. **NOCs for real time systems**
   - End-to-end (virtual) circuits
4. **The T-CREST NOC**
   - Scheduling of traffic
   - HW design (work in progress)
     - Router node
     - Network interface
5. **Conclusion**

A bus-based system



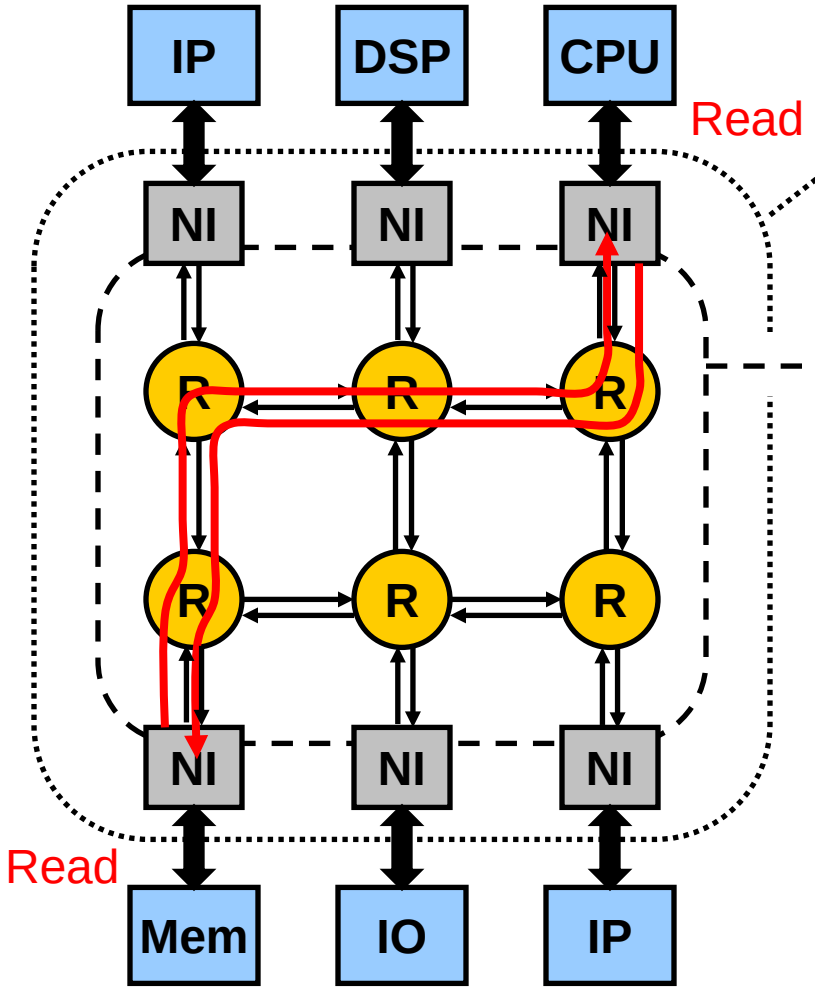
Distributed shared address space
- Std. interface (OCP, AXI; …)
- Read and write transactions

A bus-based system



Distributed shared address space
- Std. interface (OCP, AXI; …)
- Read and write transactions

A NoC based system



Distributed shared address space
- Std. interface (OCP, AXI; …)
- Read and write transactions

Packet switched network

IP — IP core

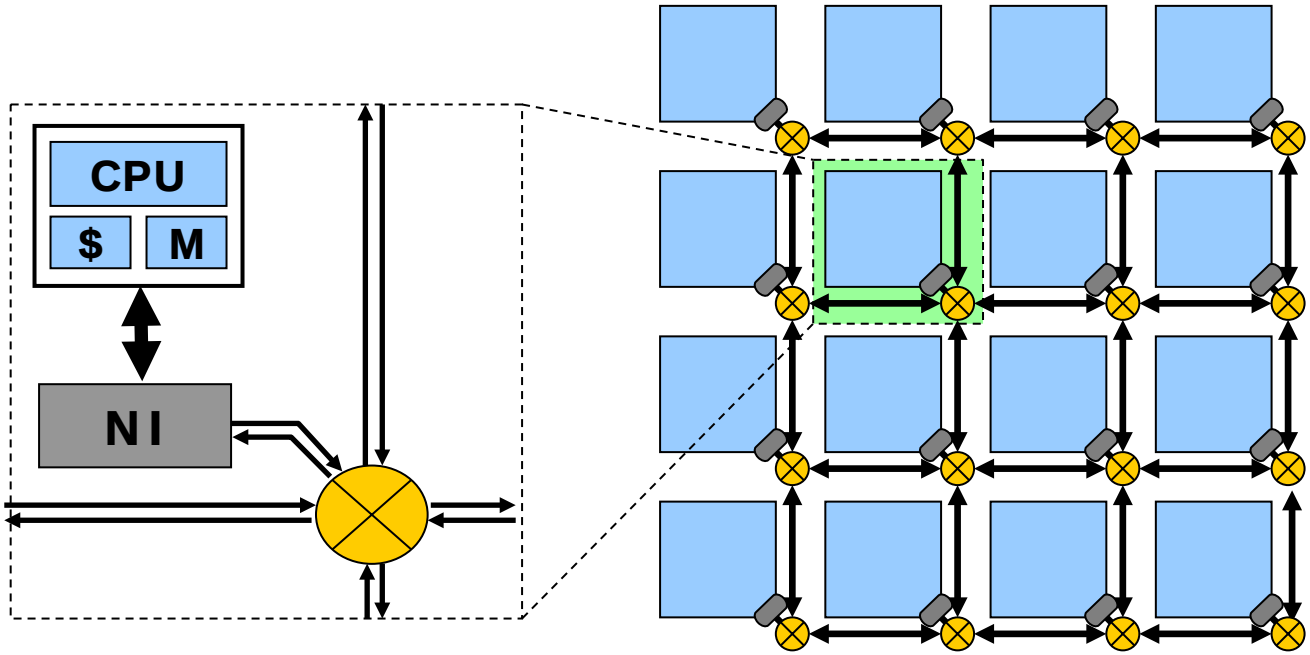
NI — Network interface (NI)

R — Router (R)

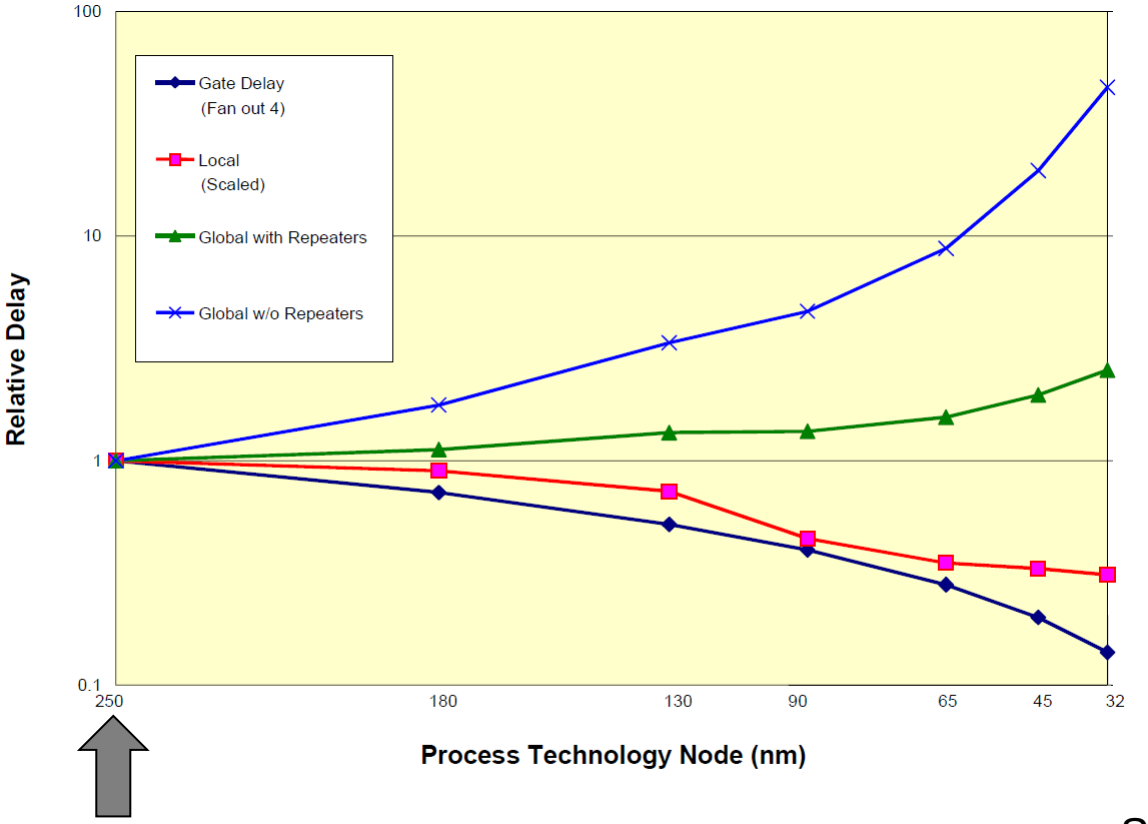
Link

Motivation for NOC-based multi-core

- Solution to a range of challenges including:
  - Design methodology (“Lego bricks”, up-scaling of BW)
  - Wires / Technology / Timing organization



Scaling of wires and logic



Design / Chip

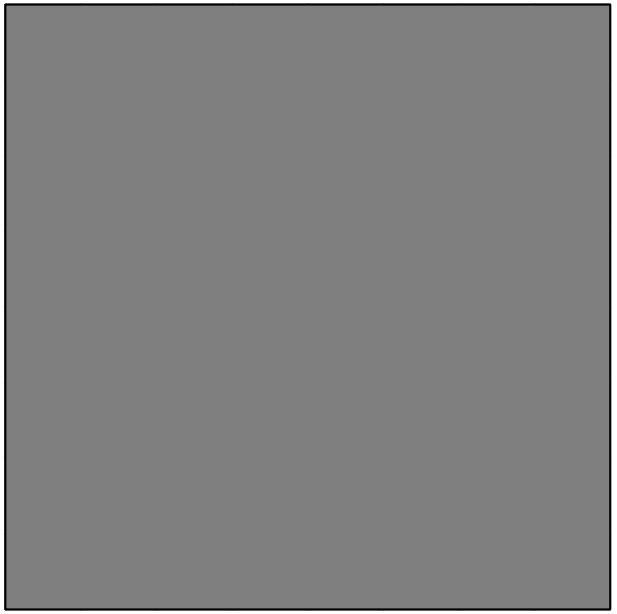
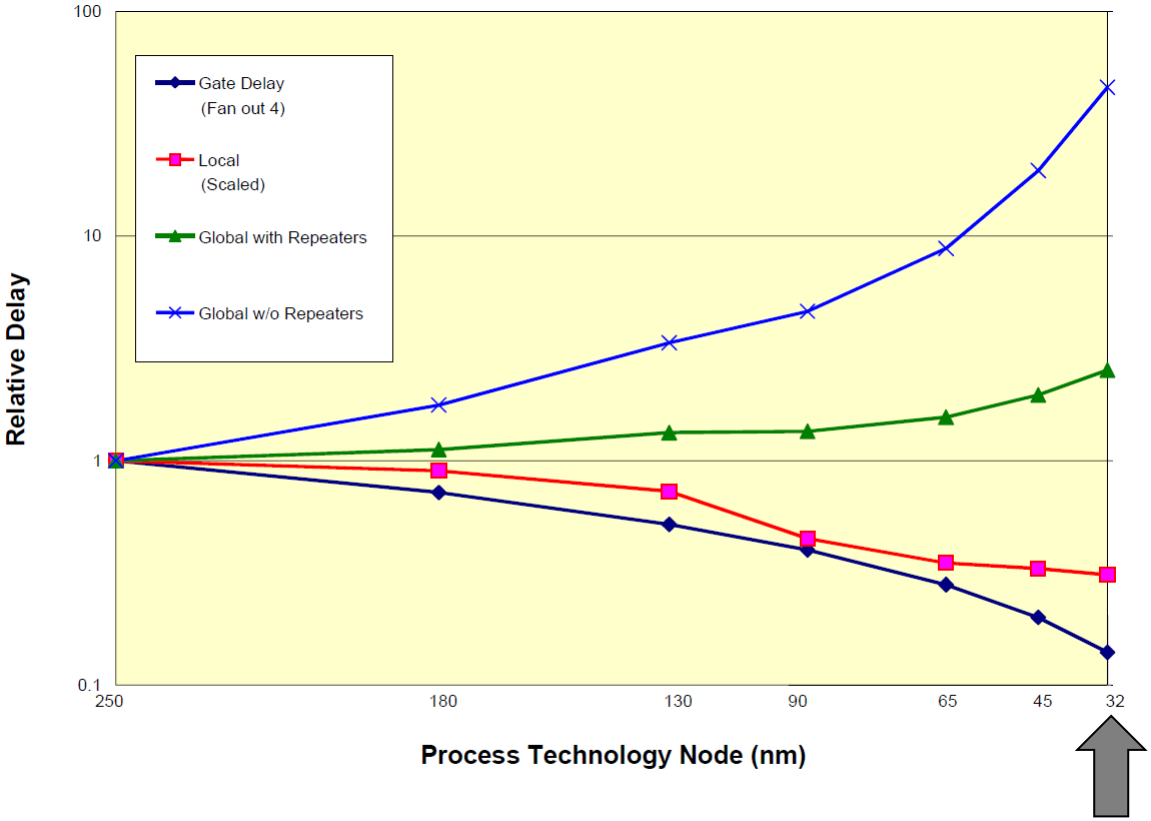
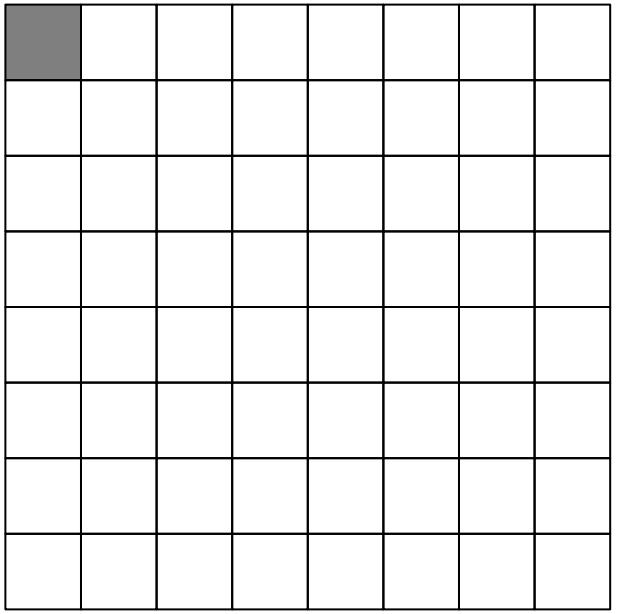


Figure 69 Delay for Metal 1 and Global Wiring versus Feature Size

Source: SIA ITRS 2005

Scaling of wires and logic

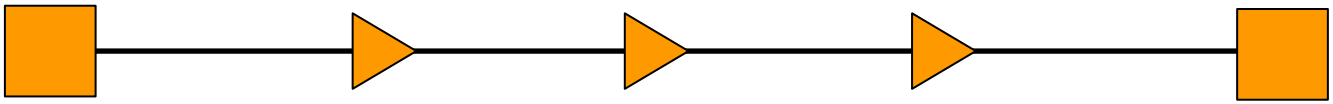




- All delay and energy consumption is in wires
- Gates/transistors are ideal

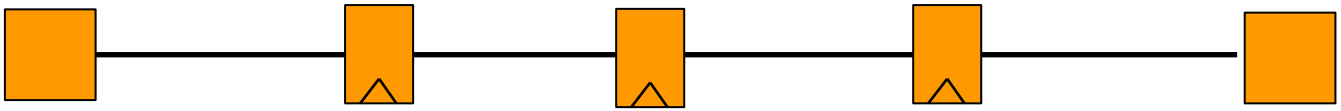
Motivations for NoC - Technology

- Long wires → repeaters



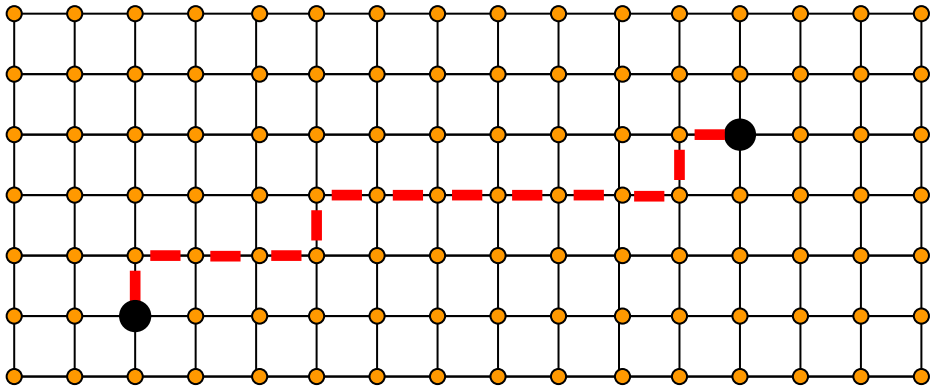
T=1/L

- Register as repeater → pipelined wire



T=3/L

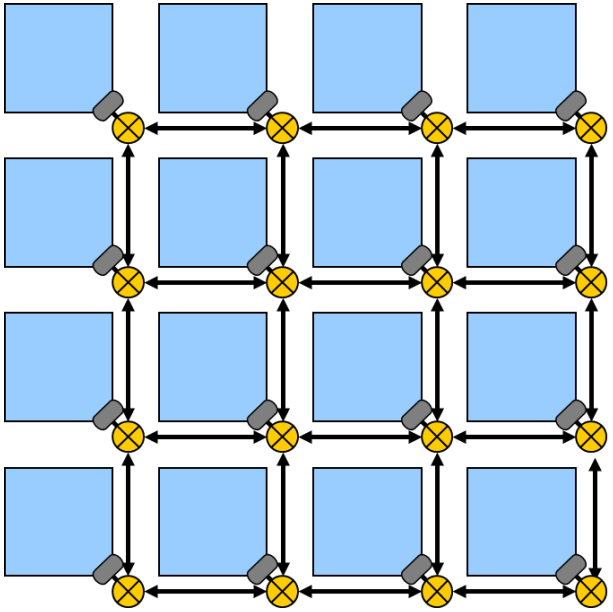
- Adding X-Y switching → Pipelined mesh; i.e. a NoC.



Motivations for NOC – Timing organization

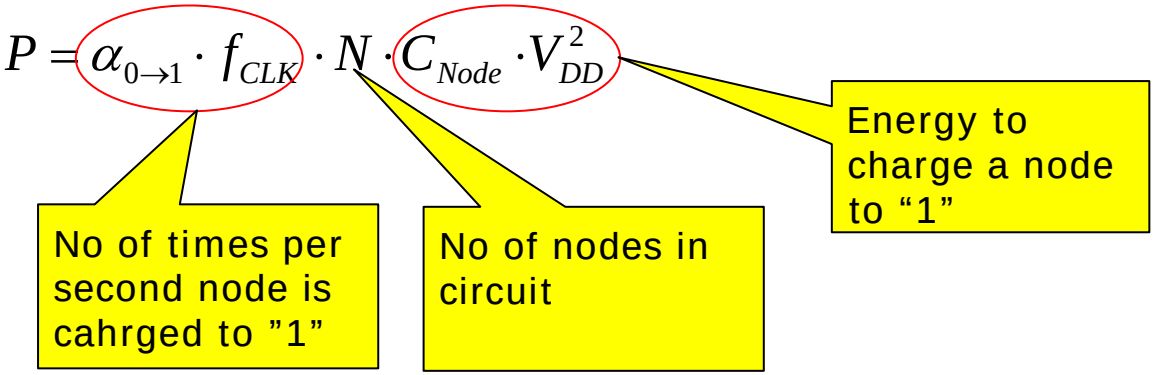
- Options (entire chip):
  - ~~Globally-synchronous~~ (single clock)
  - Globally-asynchronous locally-synchronous (GALS), i.e., independently clocked IP-cores and clock domain crossings
- How about NOC?
  - ~~Synchronous~~
  - Mesochronous

    Same oscillator but unknown phase difference.
  - Asynchronous
    - Clock domains and clock domain crossings
    - Fully asynchronous



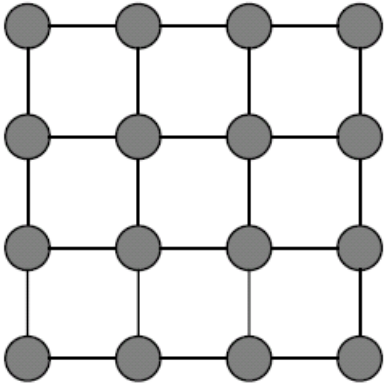
Motivations for NOC – Low power

- Minimizing power consumption:

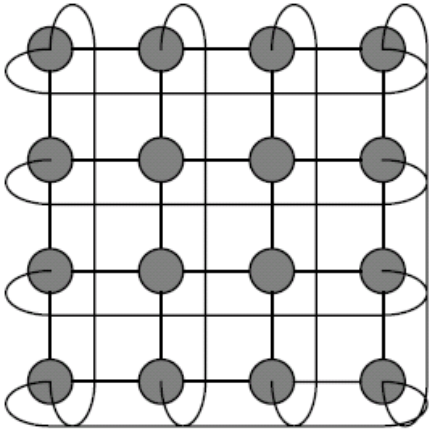


- Many simple (i.e. small) processors are better than few advanced (i.e. big) processors:
  - Speculation is overhead (do only the computation required)
  - “Small” may reduces N and C_{Node}
  - “Many” enables
    - Lowering of f_{clk} and V_{DD} (individual processors)
    - Heterogeniety (processors optimized for the job)

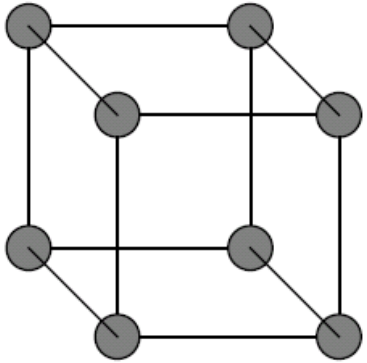
Topologies



Mesh

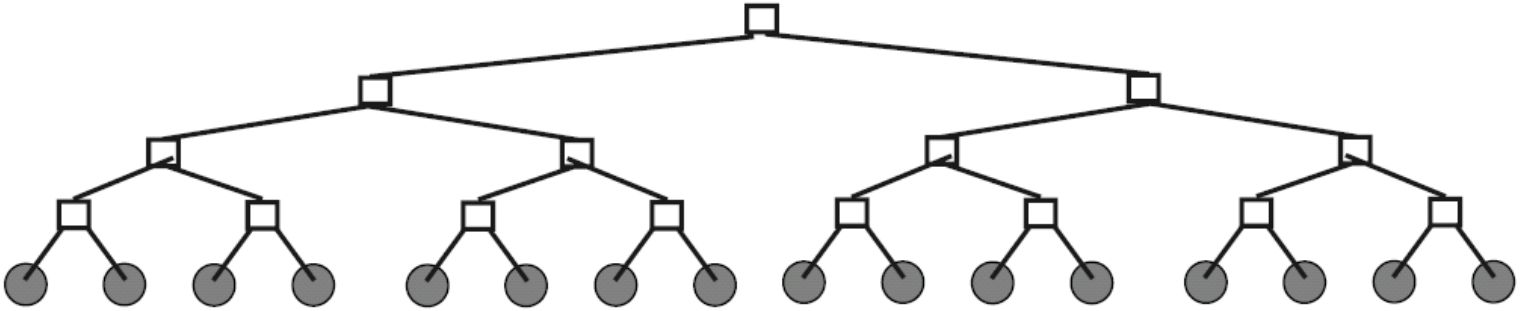


Torus

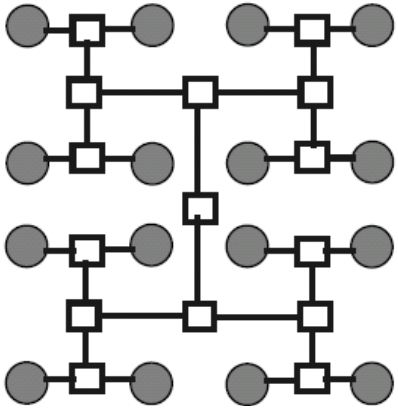


Cube

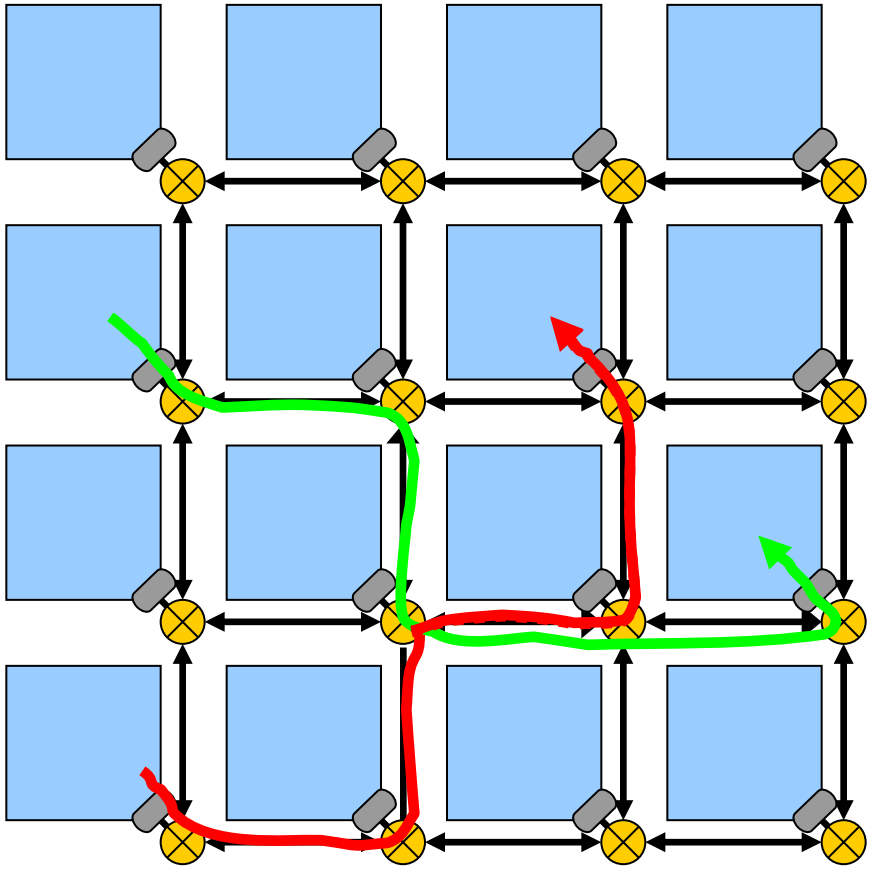
More topologies ...



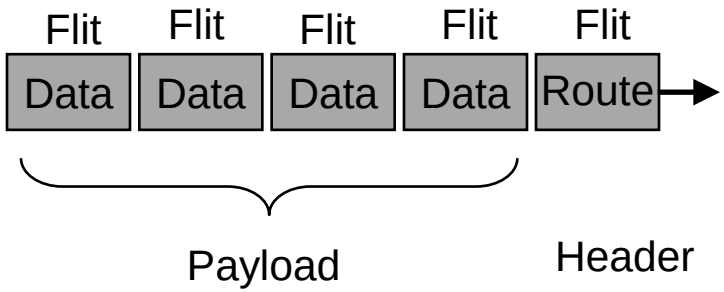
Binary Tree



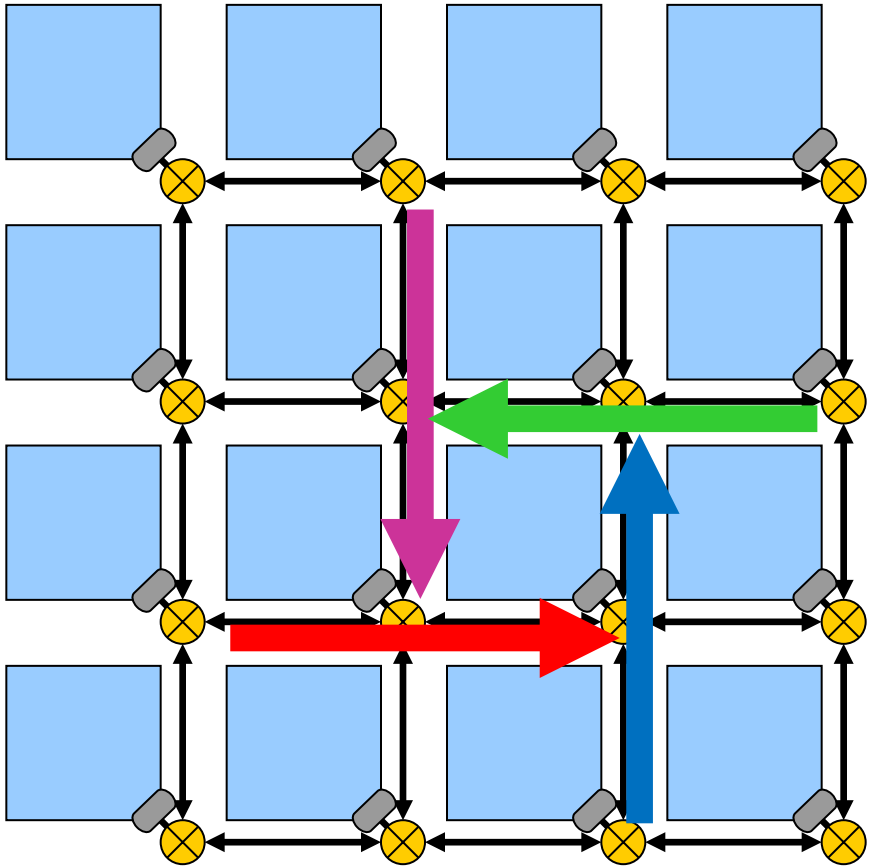
Routing



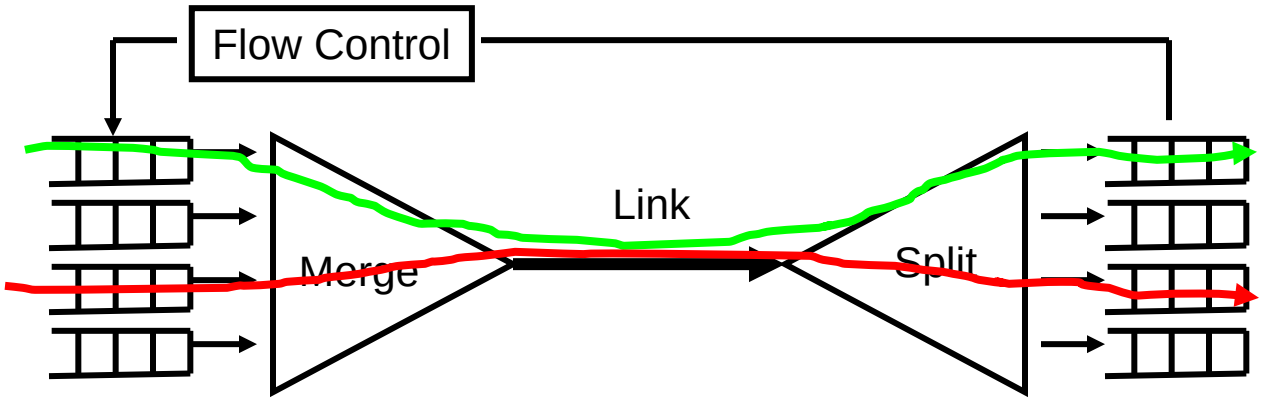
Packet:



Deadlock

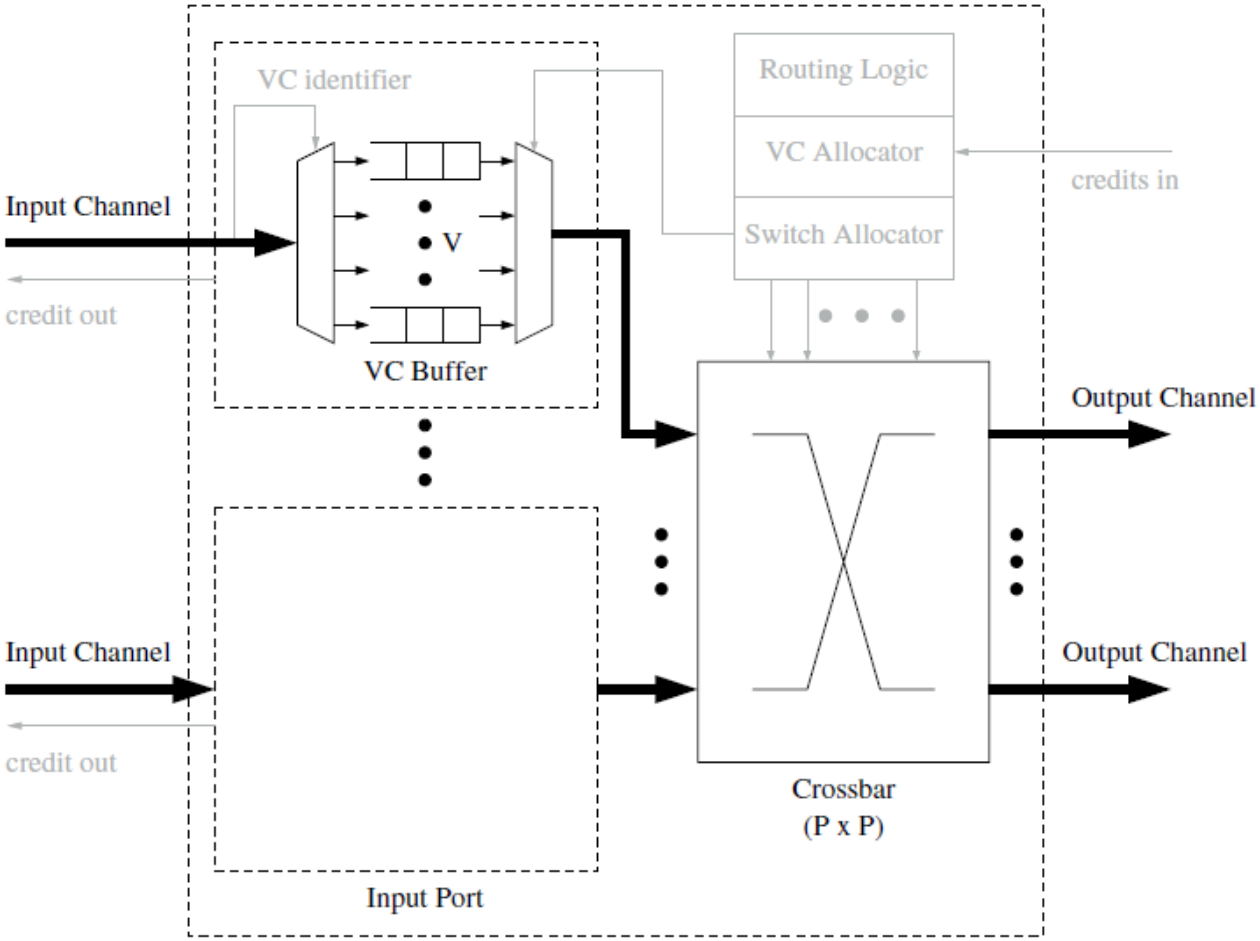


Virtual channels



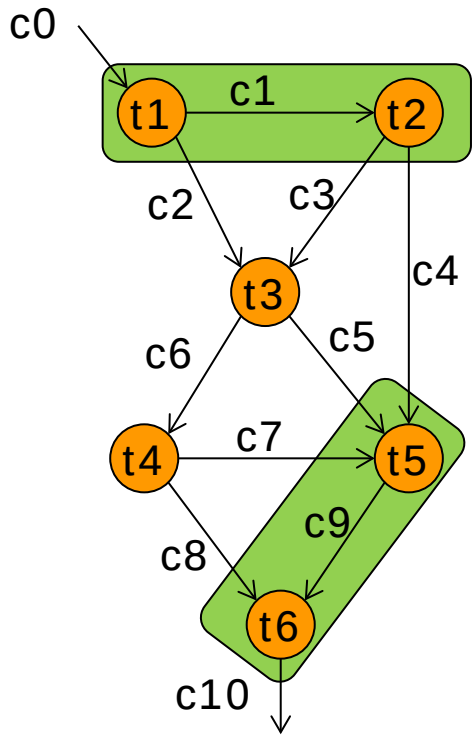
- Prevents a stalled packet from blocking the link and thereby other packets
  - Allows sharing/multiplexing of link bandwith
  - Used to break cycles and thereby avoid deadlocks
- Implemented using
  - Arbitration in merge.
  - Credit based flow control

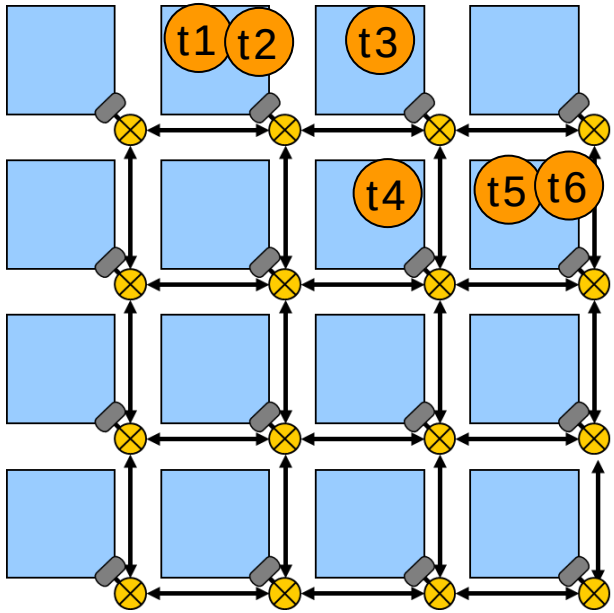
A generic VC-based router



1. Routing
2. Virtual-channel allocati
3. Switch allocation
4. Crossbar traversal
5. Link traversal

Communicating tasks, communicating IP-cores, and routing of packets in the NOC



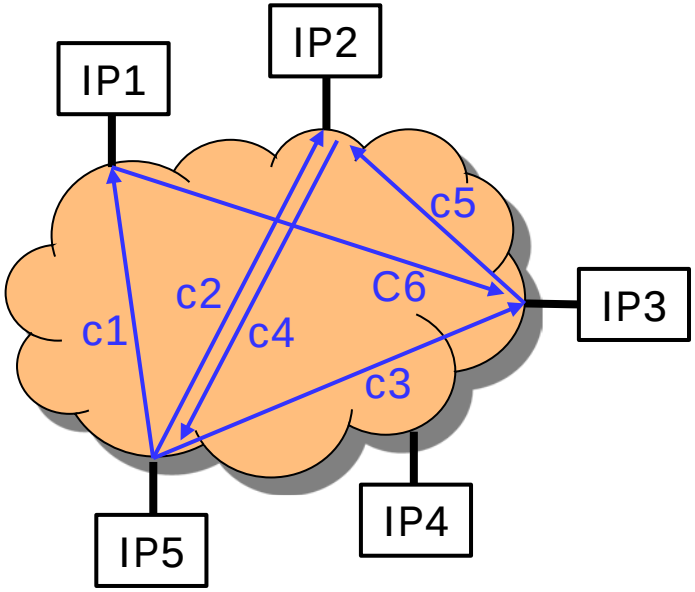


Outline

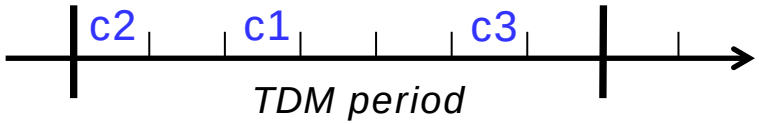
1. **Introduction**
   - Multi-core platforms and networks on chip
2. **NOC tutorial**
   - Motivation and basics
3. **NOCs for real time systems**
   - End-to-end (virtual) circuits
4. **The T-CREST NOC**
   - Scheduling of traffic
   - HW design (work in progress)
     - Router node
     - Network interface
5. **Conclusion**

Time predictable NOCs

- Note: most NOCs are not time predictable
- Time predictable
  - Avoid interference of traffic
  - End-to-end connections / circuits



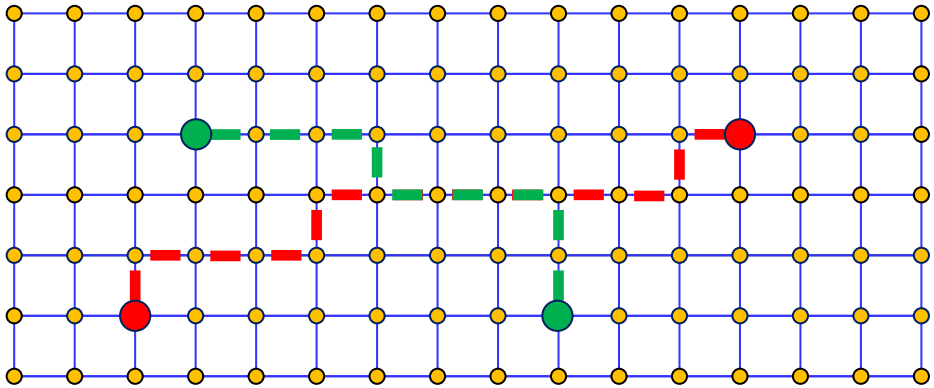
- Physical circuits (resources not shared)
  - SOCBus (D. Viklund, Linköping)
  - NOC for S4-platform (Wolkotte, Twente)
- Virtual circuits (based on time-division multiplexing)
  - Nostrum (KTH)
  - AEthereal/AElite (Goossens, TU/e)



- Virtual circuits (non-blocking routers and rate control)
  - MANGO (T. Bjerregaard, DTU)
  - Another NOC for the S4-platform (Kavaldjiev, Twente)

AEthereal / AElite (TU / e, Philips, NXP, ...)

- Time division multiplexing (TDM)
- Packet-injection into the NOC is scheduled to avoids collisions.
  - No need for arbitration.
  - Once a packet is injected into the NOC it “pipelines” and “worms” through the NOC.
- Simple and fast implementation (in principle)
- WCET analysis is simple



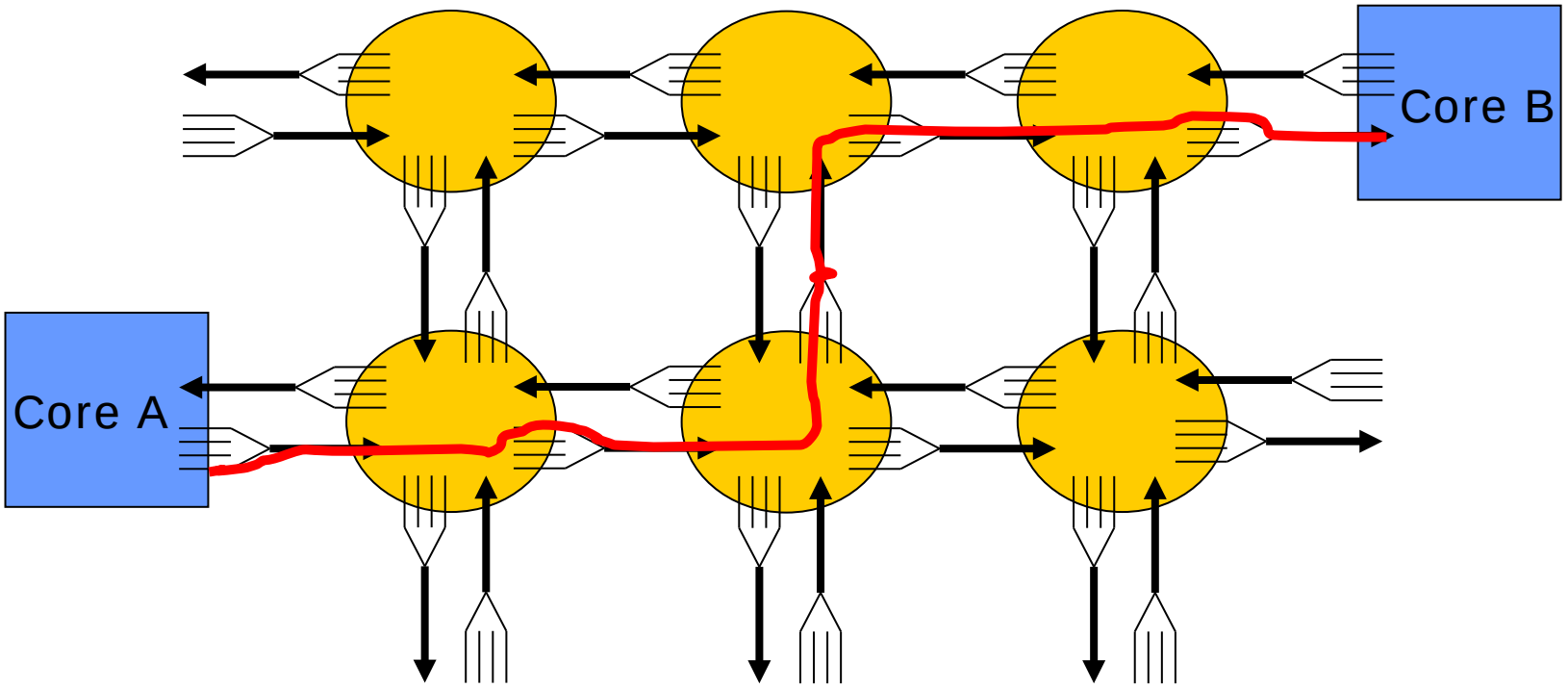
MANGO (T. Bjerregaard, DTU)

- Asynchronous NOC (Routers and links)
- Work conserving
- Virtual channels (output buffers only)
- Non-blocking routers
- Router has a number of physical VC buffers (typically 5-8) for each output port.
- End-to-end virtual circuit established as sequence of VC-buffers "connected" by statically configured crossbar switch in routers.
  - Shared links, but
  - Private VC FIFO-buffers in each router output port

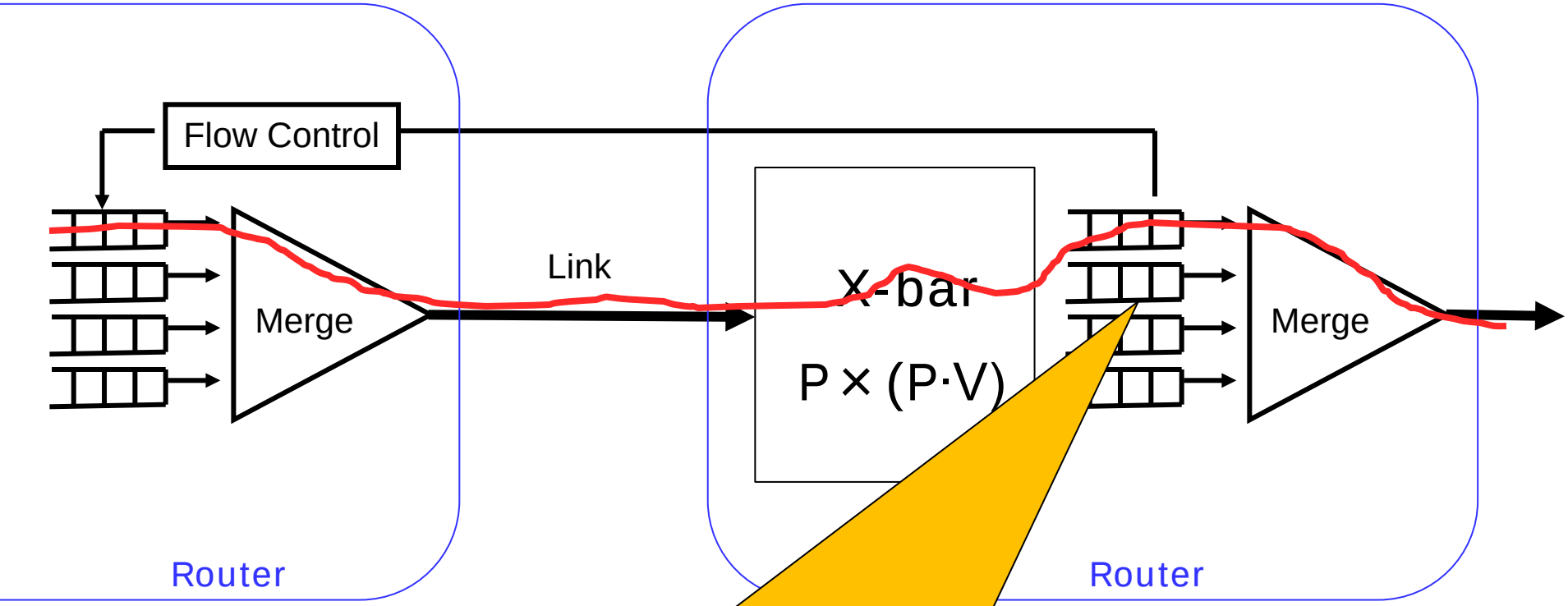
1. Bjerregaard, Tobias ; Sparsø, Jens, A Router Architecture for Connection-Oriented Service Guarantees in the MANGO Clockless Network-on-Chip. Proc. Design, Automation and Test in Europe Conference and Exhibition (DATE) pages: 1226-1231, 2005.
2. Bjerregaard, Tobias ; Sparsø, Jens, A Scheduling Discipline for Latency and Bandwidth Guarantees in Asynchronous Network-on-Chip. Proc. IEEE International Symposium on Asynchronous Circuits and Systems (ASYNC), pages: 34-43, 2005.

A Virtual (end-to-end) circuit in MANGO

- Sequence of VC-buffers connected by X-bars
- Connection owns resources in router
- Links are shared
- ALG scheduling in merge/arbiter



Mango has VC-buffers in output ports only



Routing table entries per VC-buffer:

1) "forward pointer" to VC-buffer in next router
2) "backward pointer" to VC-buffer (flow control) in previous router.

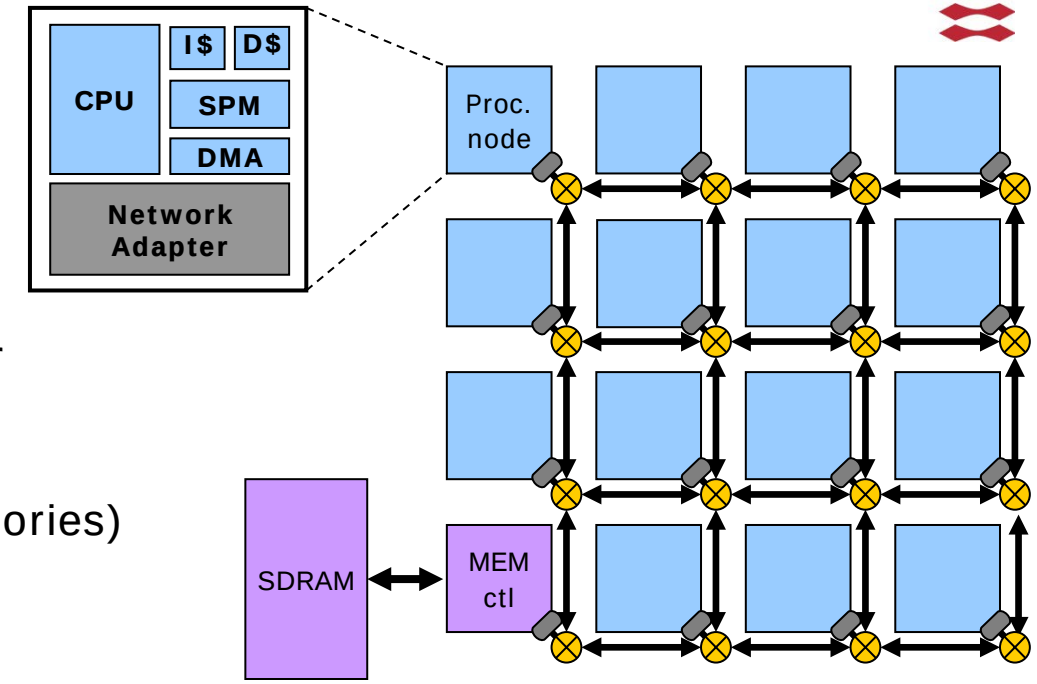
Observations / Lessons

- TDM-based Aelite router has simple/efficient implementation
- Size of MANGO router is 10x Aelite router
- Size of MANGO router is similar to simple 32-bit processor
- Mesochronous increases size of Aelite router by 2x-3x
  - Asynchronous likely to do better!
- Size of NI comparable to simple 32-bit processor
- Size of MANGO router comparable to simple 32-bit processor

Outline

1. **Introduction**
   - Multi-core platforms and networks on chip
2. **NOC tutorial**
   - Motivation and basics
3. **NOCs for real time systems**
   - End-to-end (virtual) circuits
4. **The T-CREST NOC**
   - Scheduling of traffic
   - HW design (work in progress)
     - Router node
     - Network interface
5. **Conclusion**

T-CREST platform



- Processor node:
  - Time predictable processor (Patmos)
  - Local memories (caches, scratch-pad memories)
  - DMA-controllers
- Memory controller node:
  - Time-predictable access to a shared off-chip SDRAM
- Network-on-chip:
  1. Communication between processor nodes
     Scratch-pad (SPM) to scratch-pad driven by DMA
  2. Communication between processor nodes and memory (controller)
     Scatch-pad to memory-controller

The T-CREST Network on Chip

- Baseline

  TU/e:

  – TDM-based Aelite NOC (synchronous and mesochronous)

  DTU:

  – Asynchronous design expertise

  – Mango and ReNoC networks on chip.

- Aim of T-CREST NOC work-package:

  – Asynchronous implementation of a TDM-based NOC

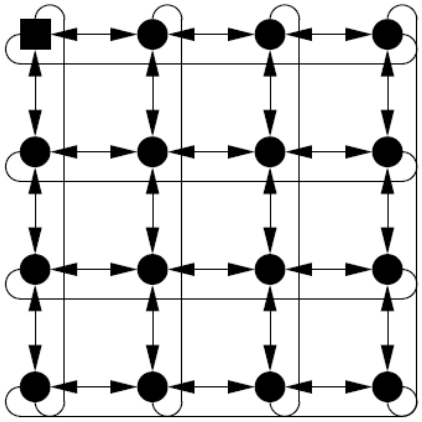
- After one year:

  – Increased focus on microarchitecture of network adapter.

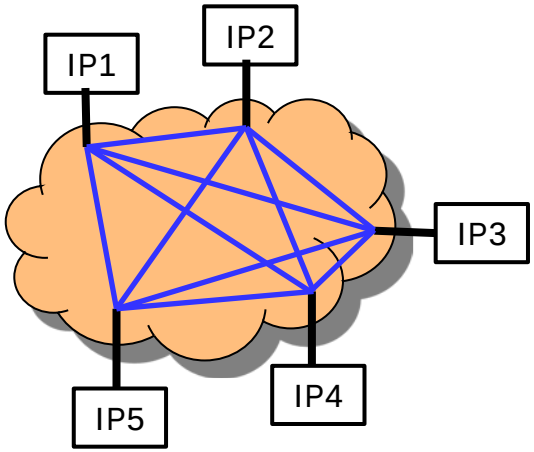
  – Increased focus on hardware cost (surprisingly little published)

A first T-CREST NOC design experiment [1)]

- Regular network topology
  - Mesh, torus, bidirectional torus



- All-to-all communication
- Single word messages

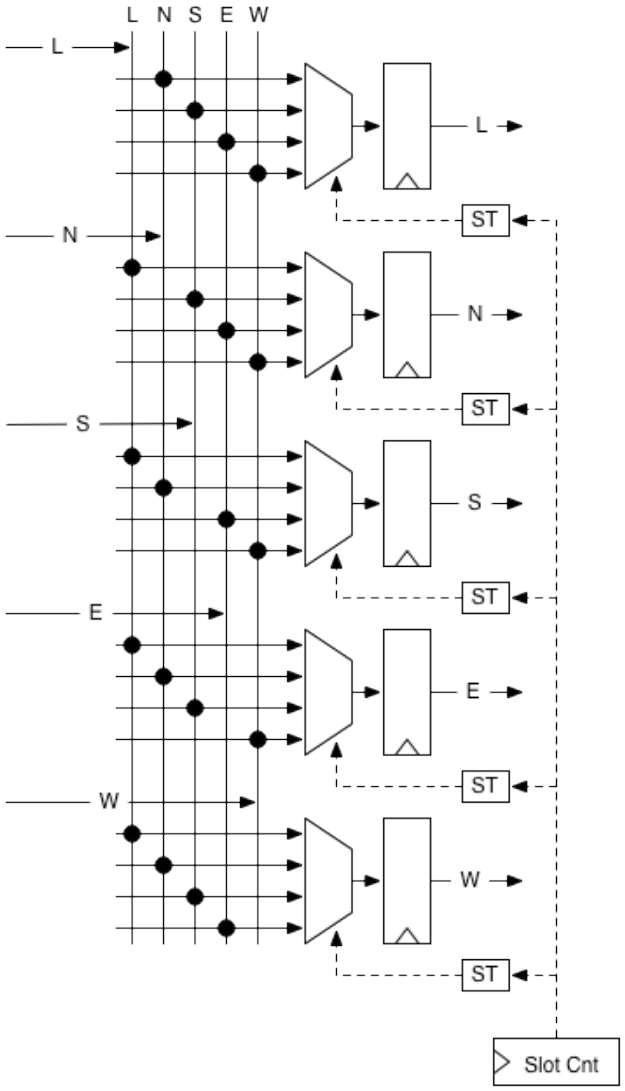


- Schedule? (heuristics/optimal)
- Length of schedule period?
  - Determines latency

1) M. Schoeberl, F. Brandner, J. Sparsø, E. Kasapaki, "A Statically Scheduled Time-Division-Multiplexed Network-on-Chip for Real-Time Systems. Proc. NOCS'12, pp. 152-160, 2012.

The Router

- Single cycle per hop
- Just multiplexer and register
- Routing information in the
  - Router
  - Network interface
- In FPGA only
  - One FF and one 6-input LUT per link-bit
  - Slot table per output port
- Static schedule
  - Generated off-line
- Slot table initialized at start-up



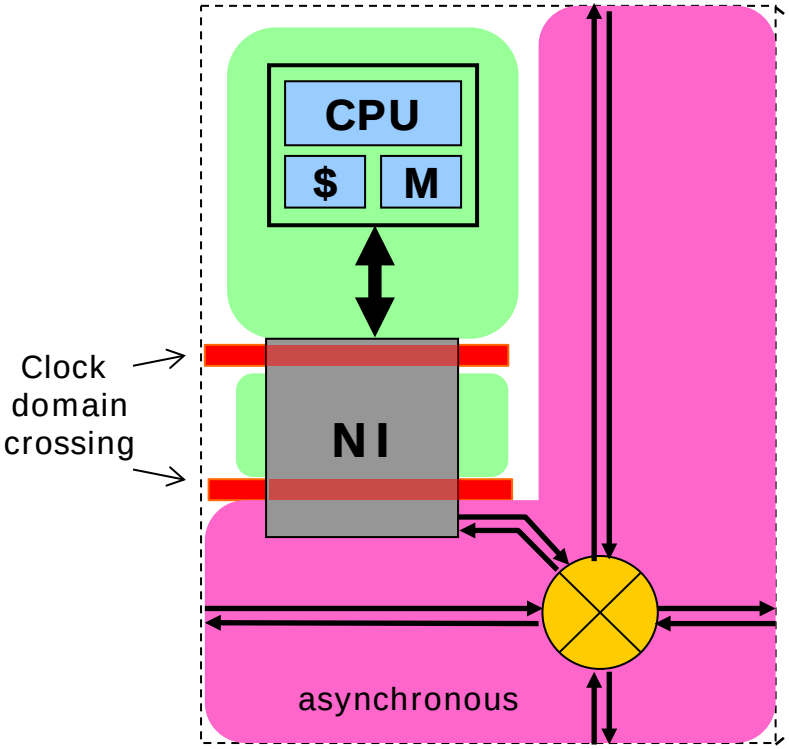
TDM schedule period

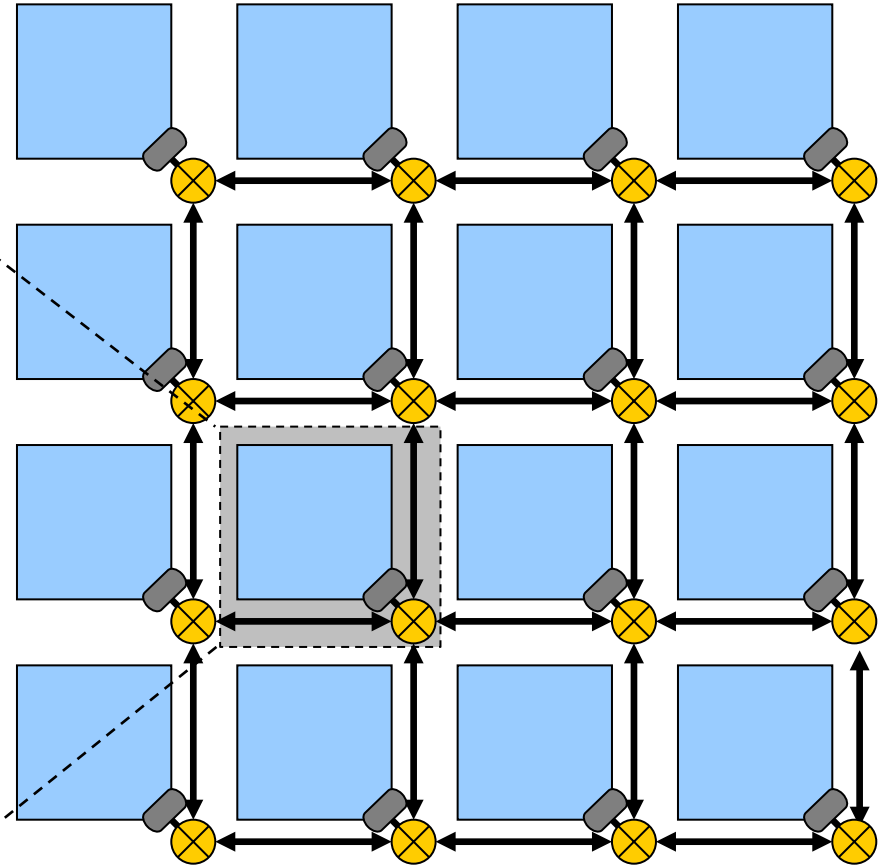
- Bounds
  - IO Bound (n-1)
  - Capacity bound (# links)
  - Bisection bound (half to half comm.)
- Actual minimum schedule periods (optimal or best found)

Size	Mesh		Torus		Bi-torus	
	bound	min	bound	min	bound	min
3x3	8	10	9	11	8	10(10)
4x4	16	18	24	26	15	18(19)
5x5	32	34	50	52	24	28(30)
6x6	54		90		35	(45)
7x7					48	(63)
8x8					64	(86)
9x9					92	(113)

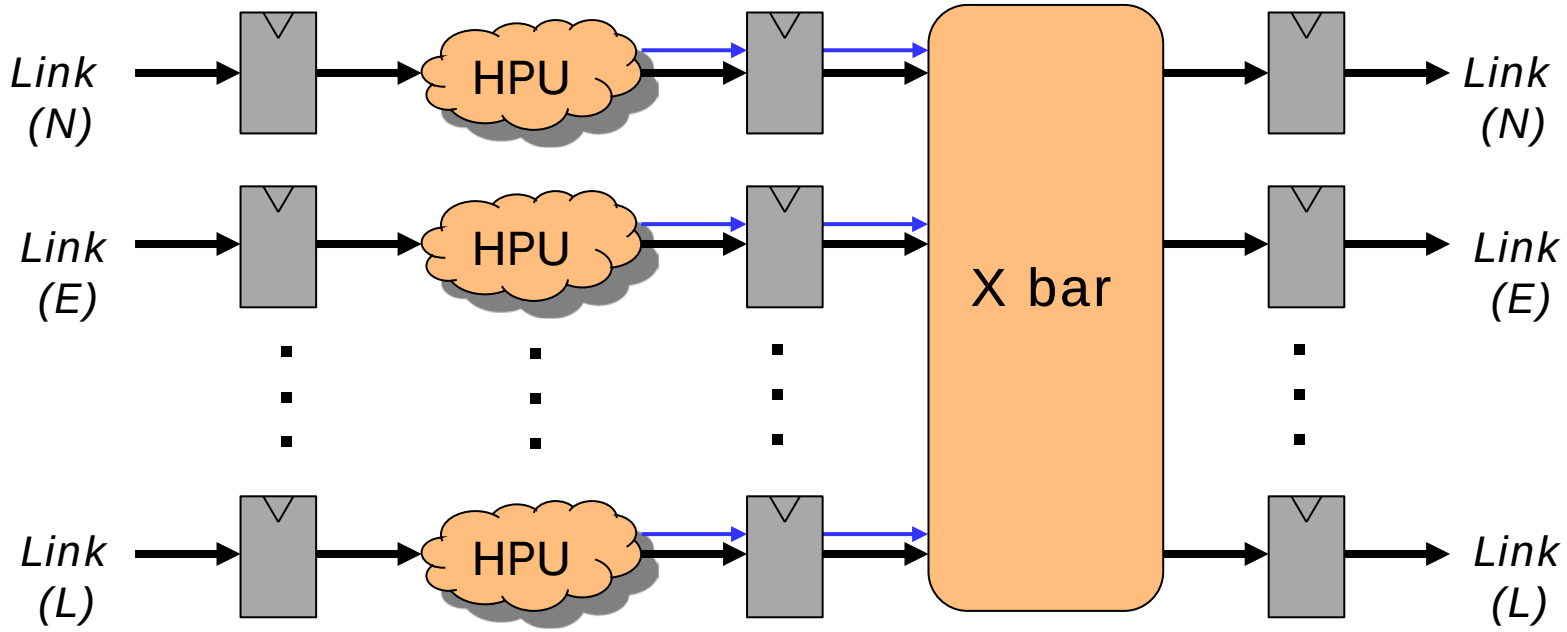
Timing organization of T-CREST NOC

- Independent IP-clocks
- Single NI-clock. Basis for TDM (Mesochronous NIs)
- Asynchronous routers and links

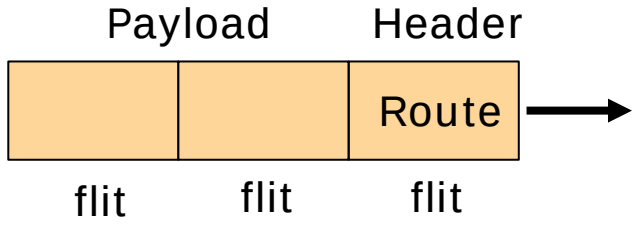




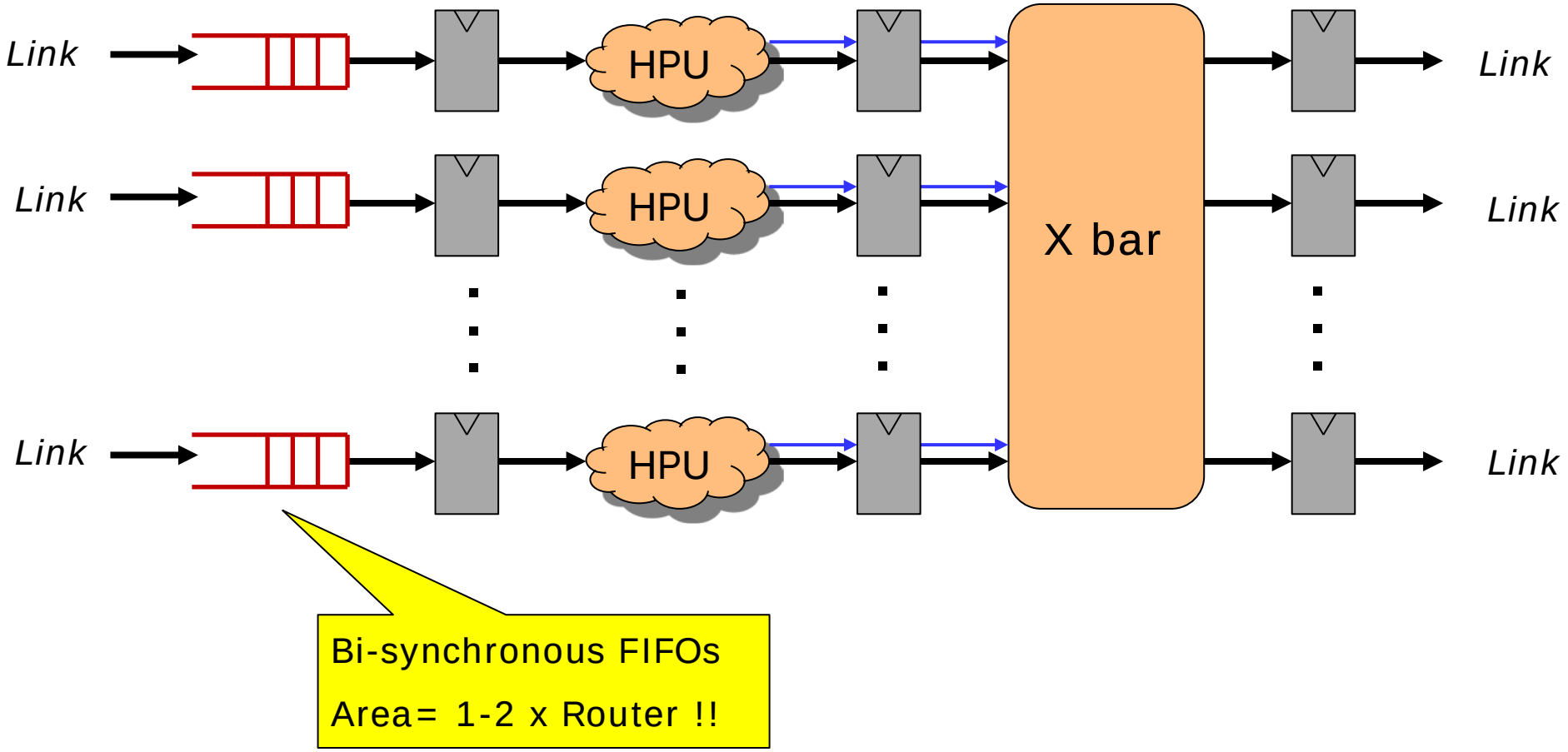
Synchronous router for source-routed TDM-based NOC



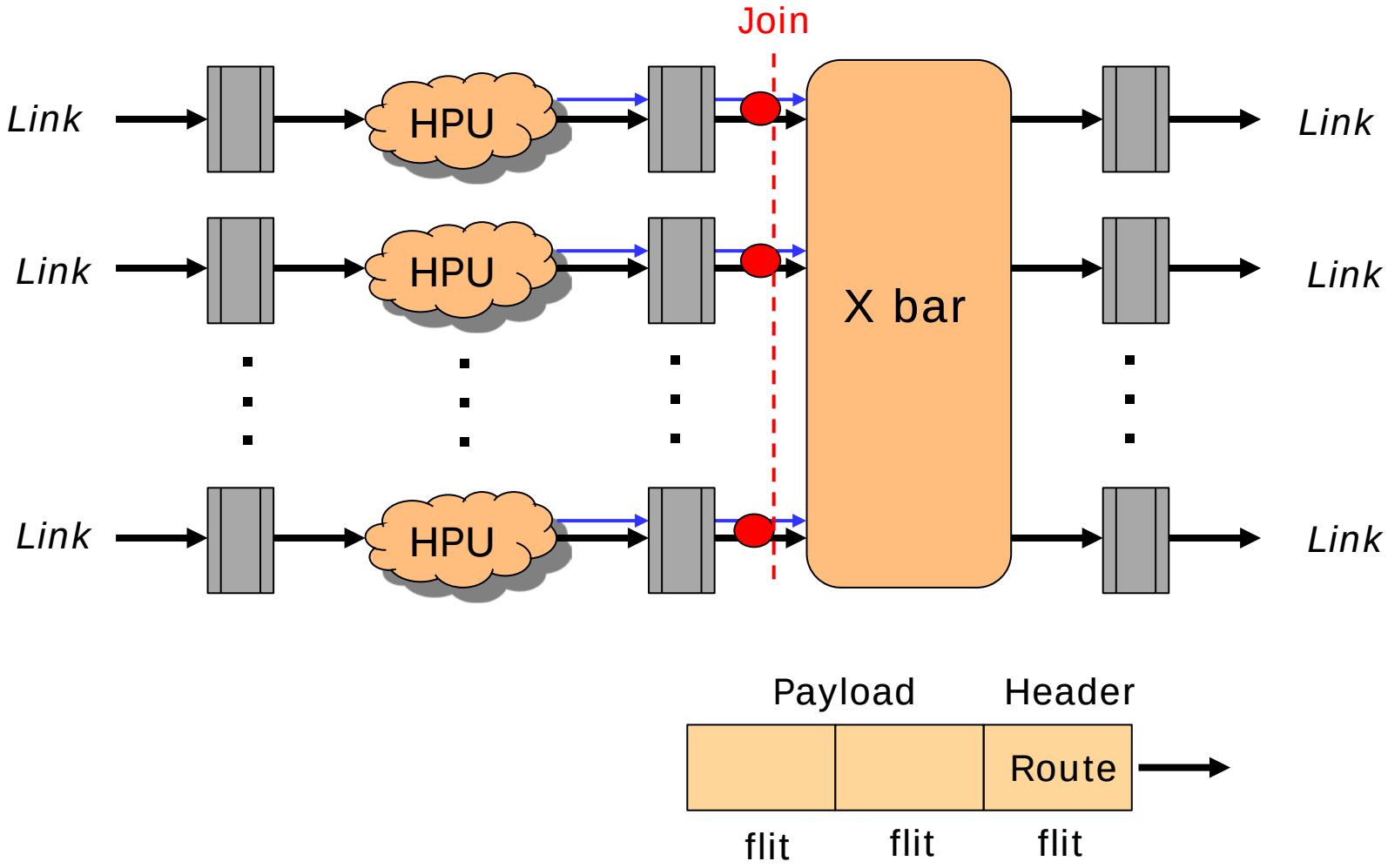
A. Hansson, M. Subburaman, K. Goossens, "Aelite: A Flit-Synchronous Network on Chip with Composable and Predictable Services," Proc. DATE 2009



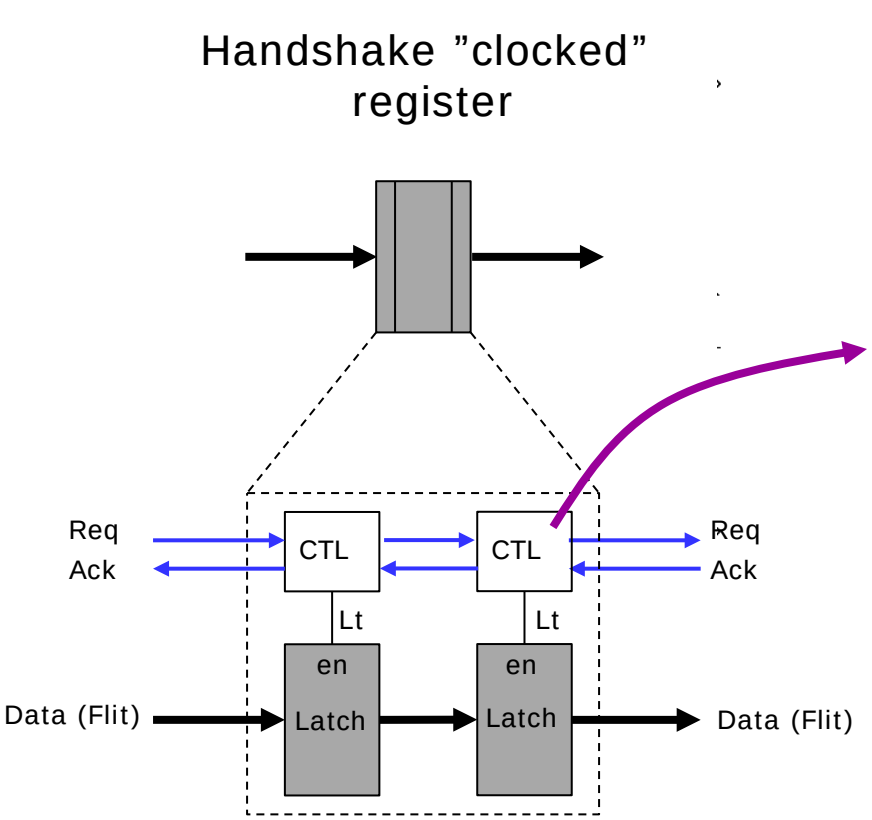
Mesochronous router for source-routed TDM-based NOC

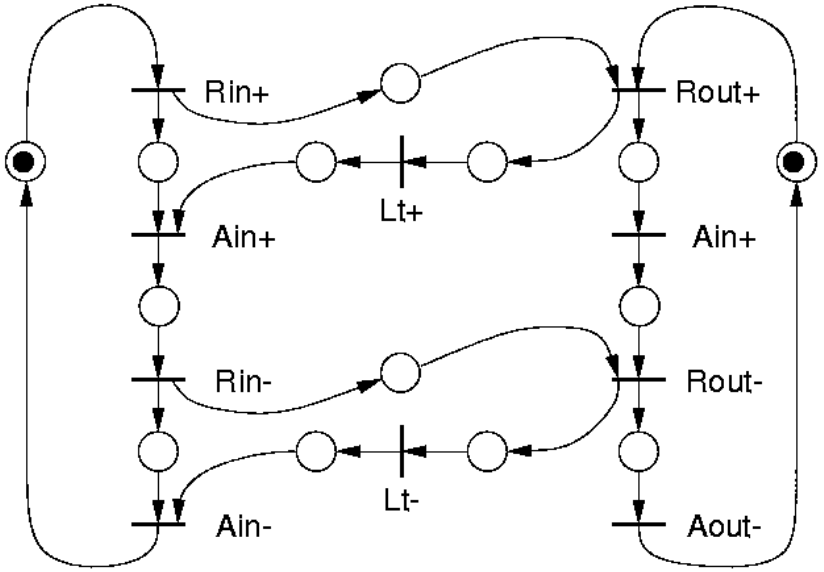


Asynchronous router for source-routed TDM-based NOC



Asynchronous handshake register



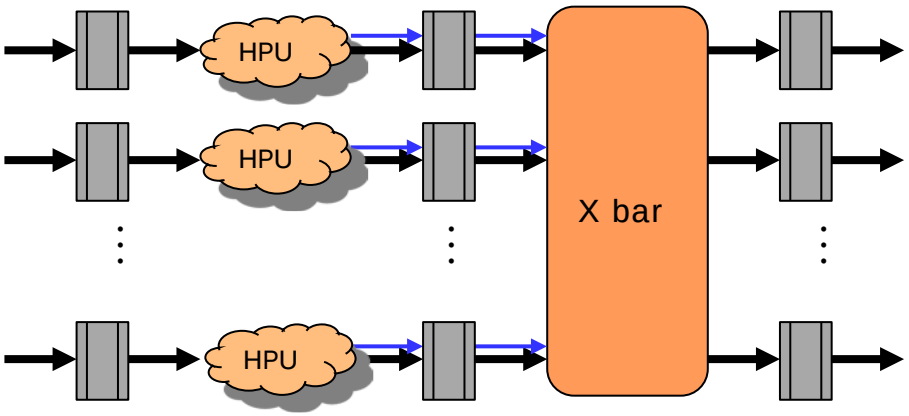


[Lt] = Rout;

[Rout] = Rin Aout' + Rout (Rin + Aout');

[Ain] = Lt;

Asynchronous Aelite router for source routed NOC

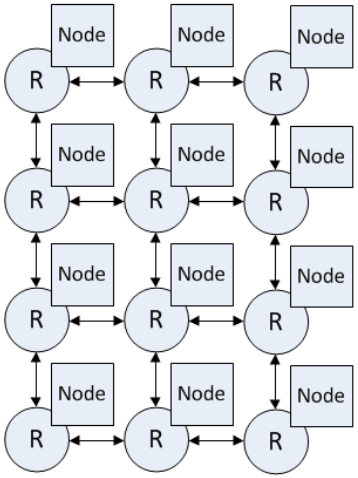


- Design is more challenging than it looks:
- Which asynchronous design style?
  - Delay insensitive / bundled data?
  - 2-phase or 4-phase signaling (events or signal levels)
- TDM requires some form of common time (ticking) but most ticks carry no data.
  - How to implement something that resembles clock gating?
  - (avoid excessive power consumption)

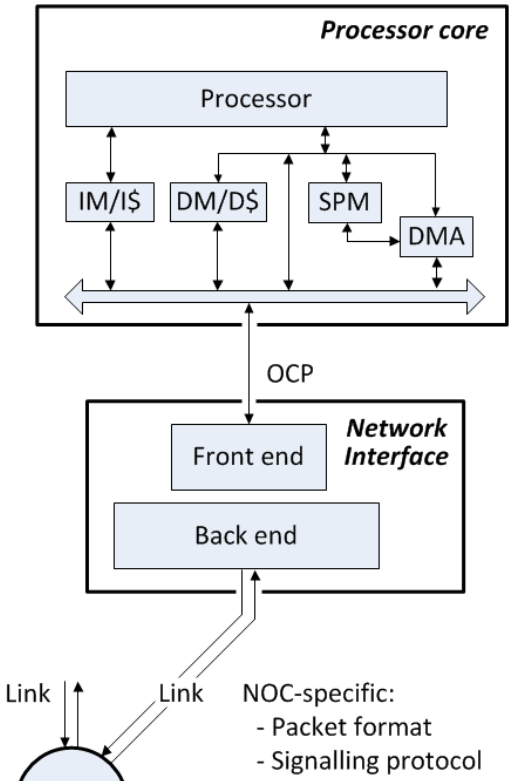
NI for T-CREST NOC

Original T-CREST plan
(Traditional NI design)

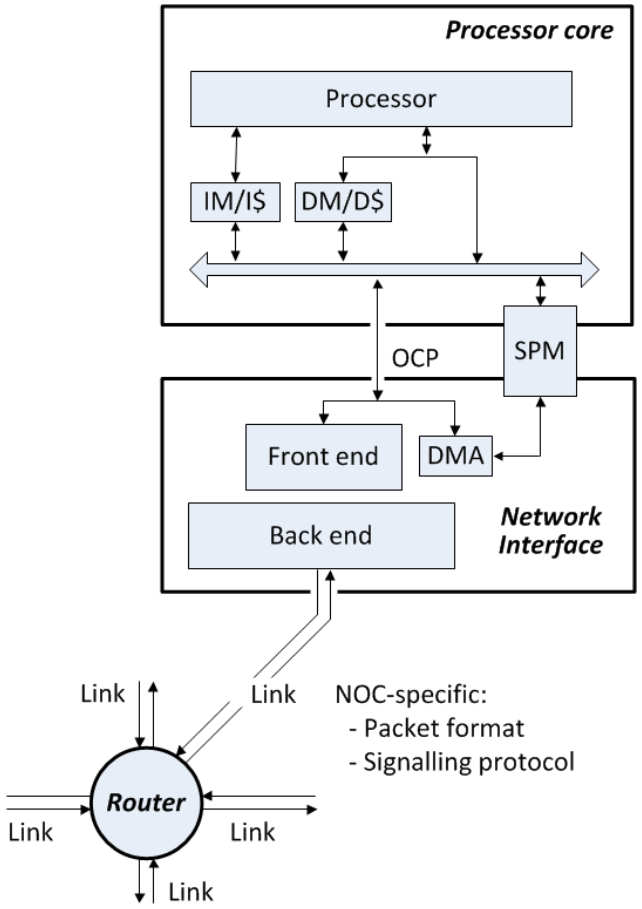
New NI micro-architecture



(a)



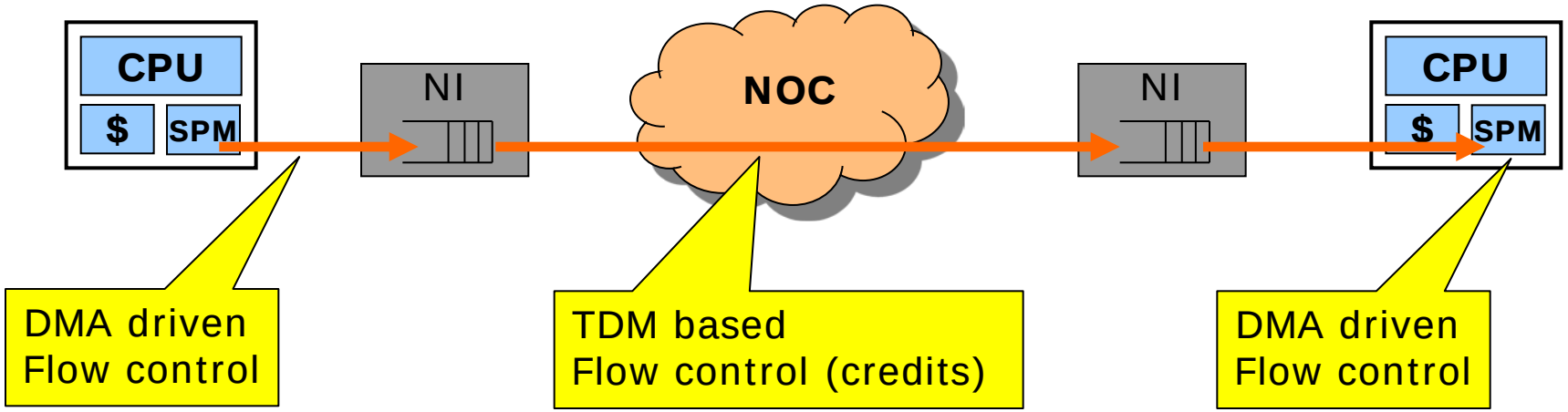
(b)



(c)

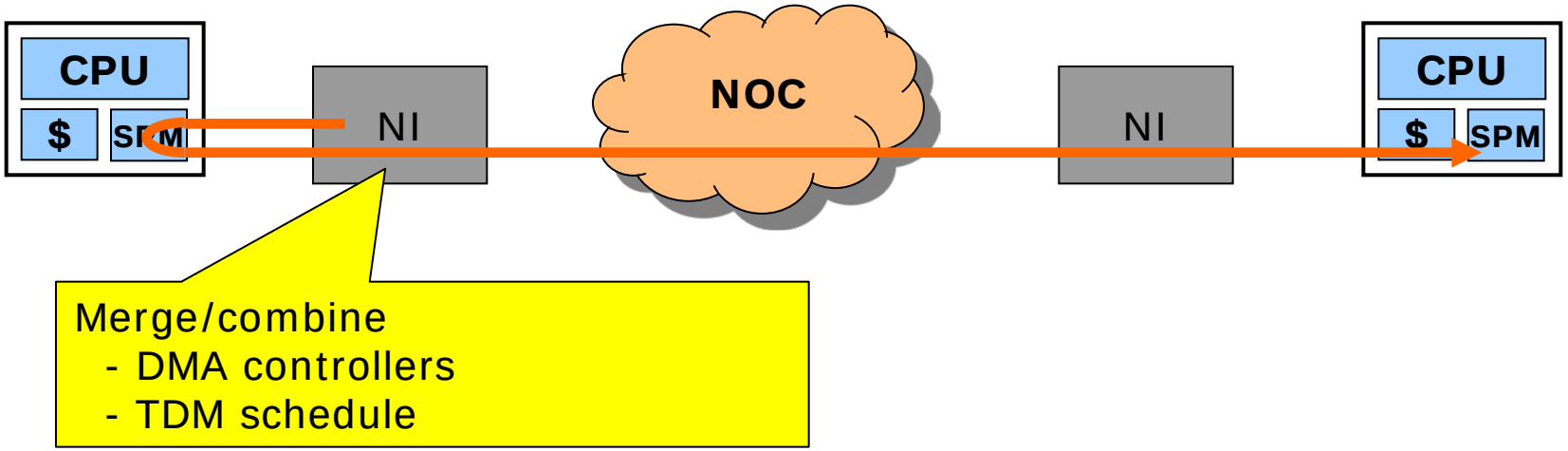
J. Sparsø, E. Kasapaki and M. Schoeberl
"An Area-efficient Network Interface for a TDM-based Network-on-Chip, Proc. DATE'13

Traditional NI design (w. buffers)

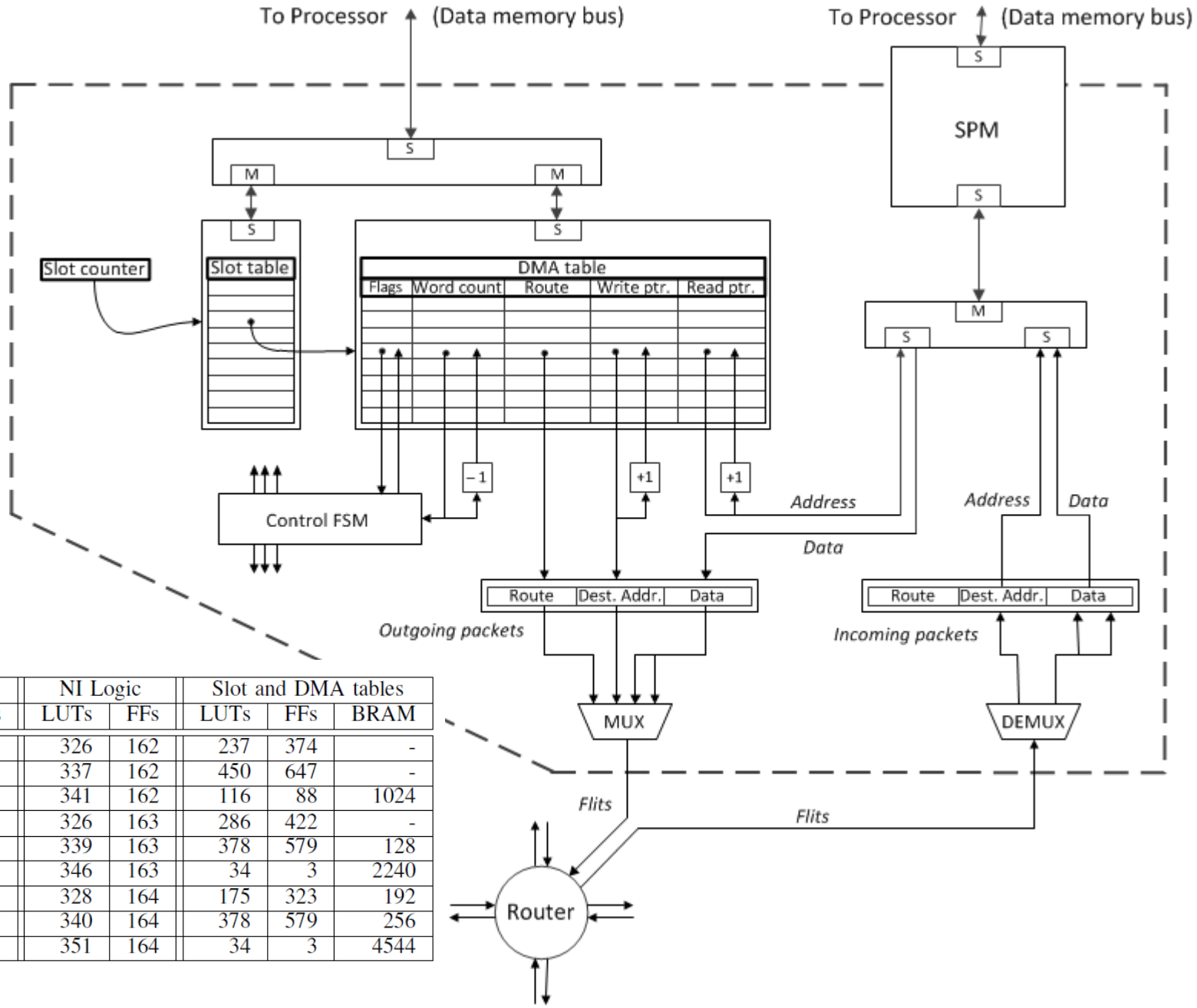


- Per-connection FIFO-buffers and flow control.
  - Accounts for 50-85% of NI hardware!

New T-CREST NI-design



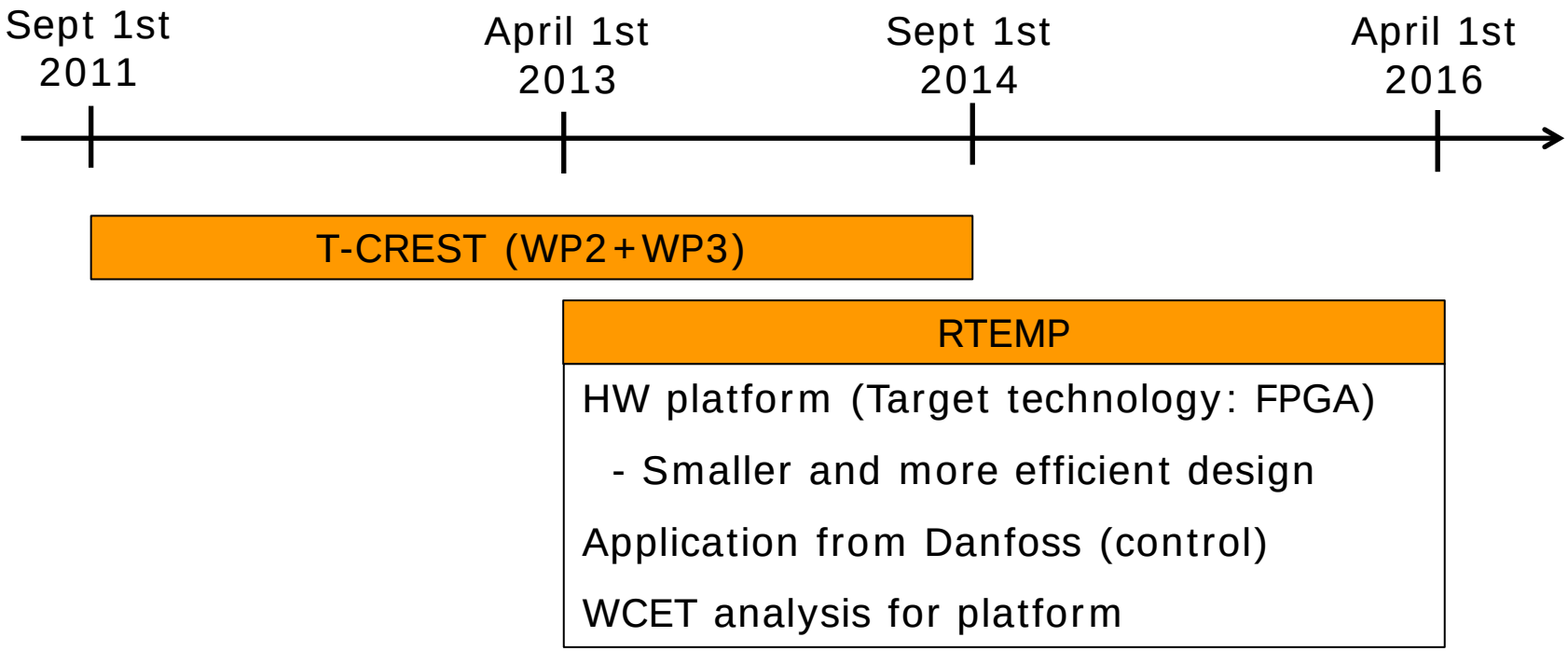
- DMA controllers in NIs and driven by TDM schedule.
  - Avoids FIFO-buffers and flow control.
  - Saves 50-85% of traditional NI hardware



NI design size		NI Logic		Slot and DMA tables		
Slot Period	DMAs	LUTs	FFs	LUTs	FFs	BRAM
16	4	326	162	237	374	-
	8	337	162	450	647	-
	16	341	162	116	88	1024
32	4	326	163	286	422	-
	8	339	163	378	579	128
	32	346	163	34	3	2240
64	4	328	164	175	323	192
	8	340	164	378	579	256
	64	351	164	34	3	4544

RTEMP http://rtemp.compute.dtu.dk

- Hard Real-Time Embedded Multiprocessor Platform - RTEMP
- The Danish Council for Independent Research | Technology and Production Sciences (FTP)



Conclusion

- NOC tutorial
- NOC's for real time systems
- Design of the T-CREST NOC
  - Globally-Synchronous Locally-Synchronous (GALS) timing architecture
  - Area-efficient network interface
  - Asynchronous router
- Brief introduction of the RTEMP project