

The Asynchronous NOC

Ran Ginosar

▶ **To cite this version:**

Ran Ginosar. The Asynchronous NOC. ISCA tutorial on "Multi-domain Processors: Challenges, Design Methods, and Recent Developments", Jun 2010, Saint Malo, France. inria-00493903

HAL Id: inria-00493903
https://inria.hal.science/inria-00493903

Submitted on 21 Jun 2010

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Outline

- Part I: Multi-Domain Processors Design Overview (2:00-2:45PM)
  - Multi-domain server, cell phone, and media processors
  - Power management techniques
- Part II: Router Design and Synchronization Issues (2:45-3:30PM)
  - Asynchronous router design
  - Quality of Service and virtual channels in QNoC
- Part III: Control and Power Management in Presence of Workload Variations (4:00-4:45PM)
  - VFI partitioning and voltage assignment
  - Workload modeling and dynamic control of multi-VFI designs
- Part IV: DVFS in Presence of Process Variations (4:45-5:30PM)
  - Impact of process variations on DVFS controller performance
  - Technology-driven limits on DVFS controllability

ISCA-2010 Tutorial #2

The Asynchronous NOC

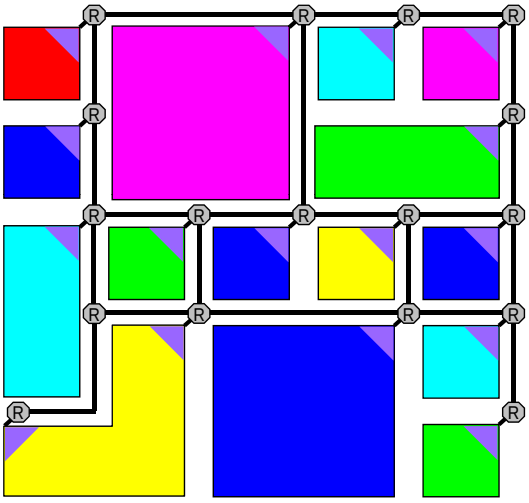
Ran Ginosar

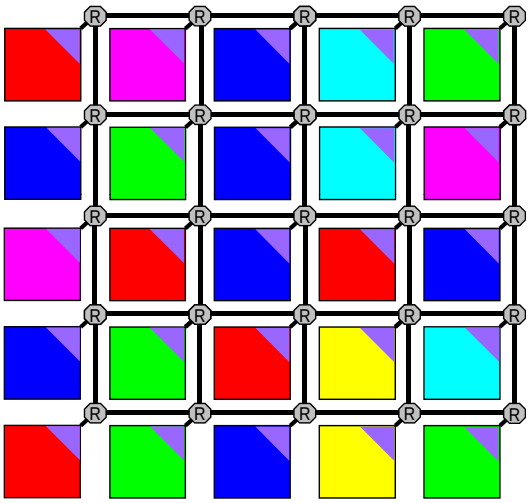
Technion

ran@ee.technion.ac.il

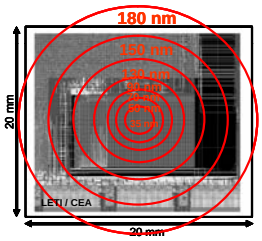
TECHNION

SoC and CMP

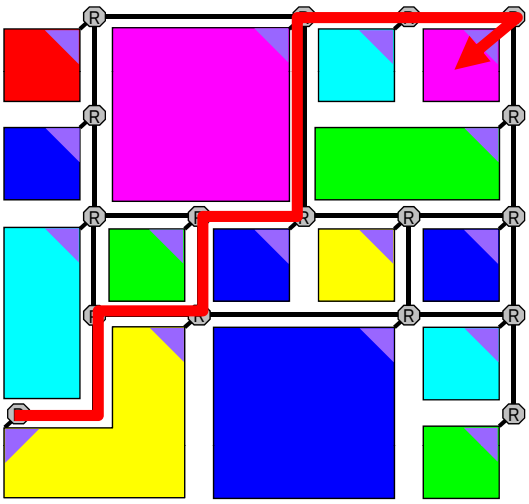


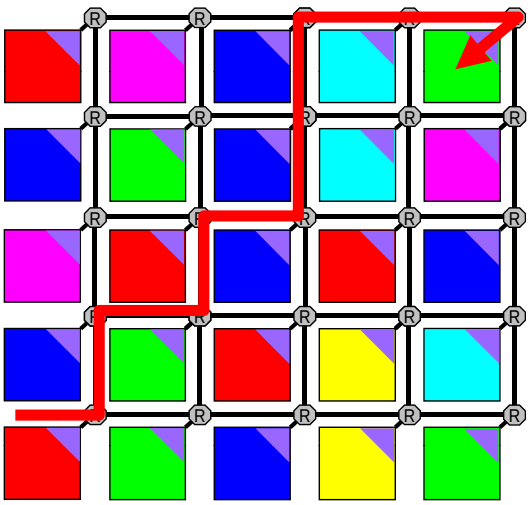


- Each block is a separate clock (+voltage) domain
  - Fast clocks cannot make it over long distances
  - They are hard to design
  - They consume too much power

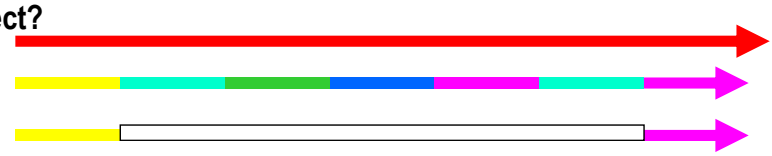


SoC and CMP





- What clock for the interconnect?
  - Fastest?
  - Opportunistic?
  - None?



The message

- NOCs are for large SOCs and CMP
- Large SOCs / CMP = multiple clock domains
  - A.k.a. GALS = Globally Asynchronous, Locally Synchronous

→ NOCs should be asynchronous
  - The link is asynchronous
  - The router is asynchronous
  - Synchronizers needed at block—NOC interfaces

What's in the remaining slides?

- NoC 101
  - Why, how
- Async NOCs
  - QNoC (Technion, Israel)
  - Faust / Alpin / Magali (LETI, France)
  - Mango (DTU, Denmark)

Why NOC?

- Scalability: Busses and p2p don't scale; NOC can scale
- Design: Simplify design of blocks, of interconnect, of timing
- Power: Lower power in global wires and in clocks

- Special discipline
  - Seen it before with power supply, clock network
  - Solutions in RTL and in "hard IP cores" and special circuits
  - Confiscate interconnect from logic design team to back-end design team

How to NoC?

- Copy ideas from networking
  - But don't over-do it!
  - Adapt to VLSI: Low area and power
- Connect routers and links
- Add Network Interface (NI) to each block
- Wrap communications as
  - Packets
  - Each packets = many flits
- Route packets (route flits)
- Plan for needed bandwidth, latency, etc

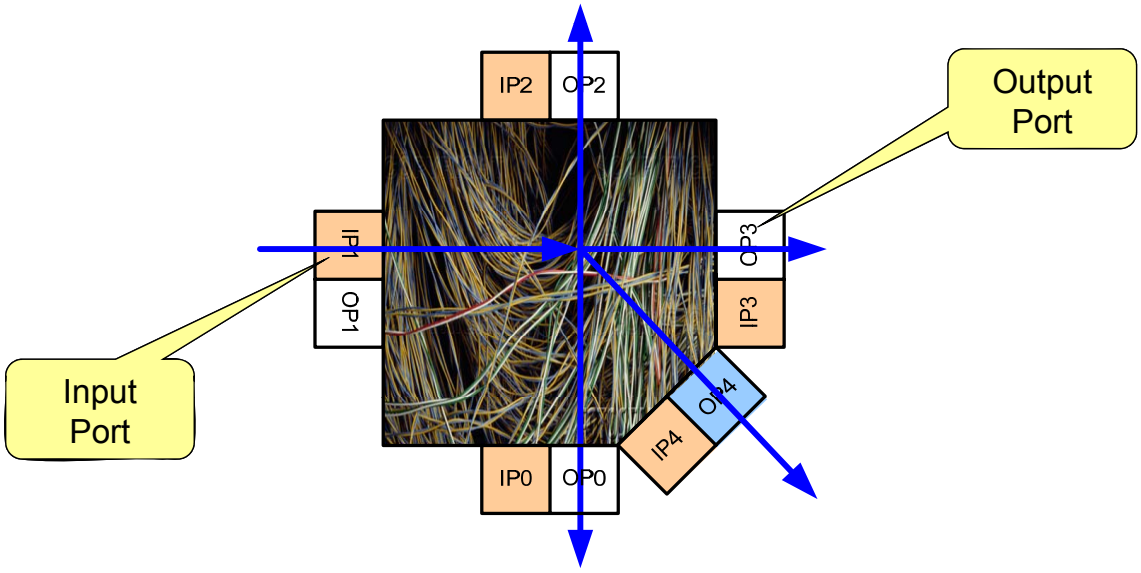
QoS in NoC

- Various types of traffic
  - Long data blocks – high bandwidth, long latency, low power, statistical delivery (“parcel post”)
  - Fast quick short msgs – low latency guarantee, maybe high power, rare (“Fedex”)
  - Others …
- QoS NoC
  - Must provide for all comm requirements
  - May use levels of priority
  - May use “virtual channels”
  - May combine “Guaranteed Service” with “Best Effort” (GS + BE)

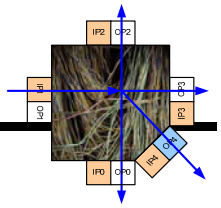
The QNoC Async Router

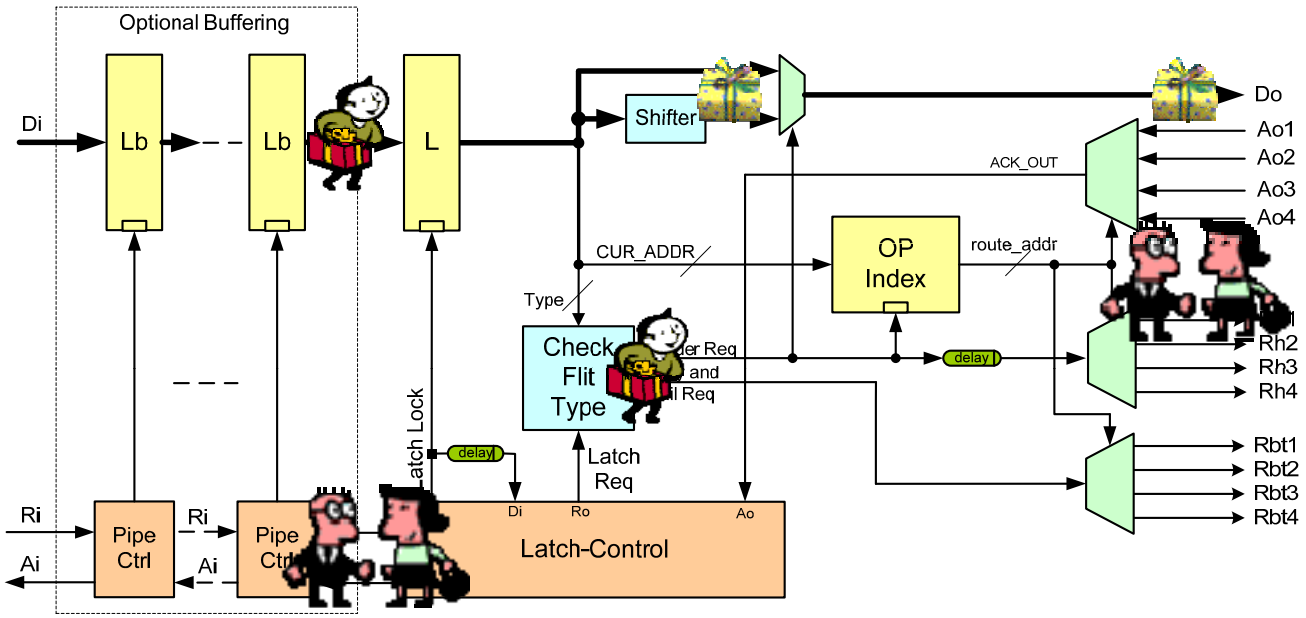
Single-Service-Level Router

Service level = priority level

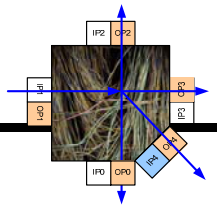


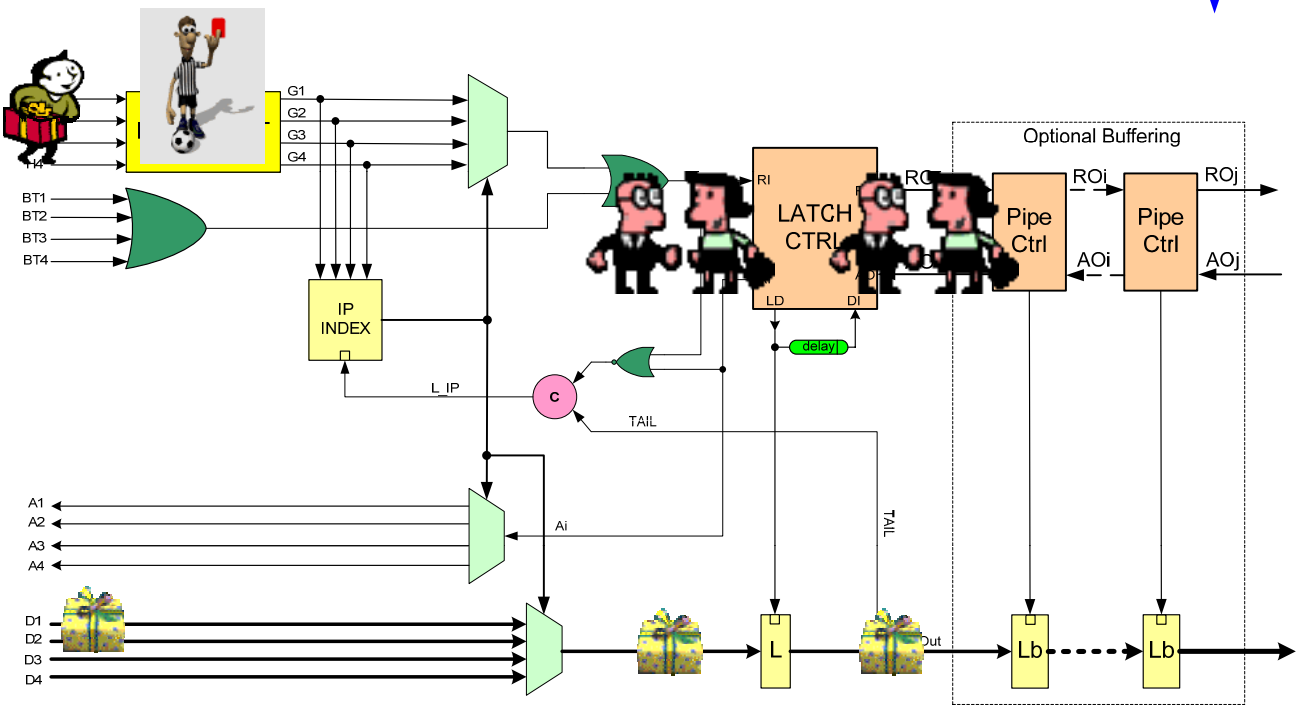
Async Single-Service-Level Input-Port





Async Single-Service-Level Output-Port

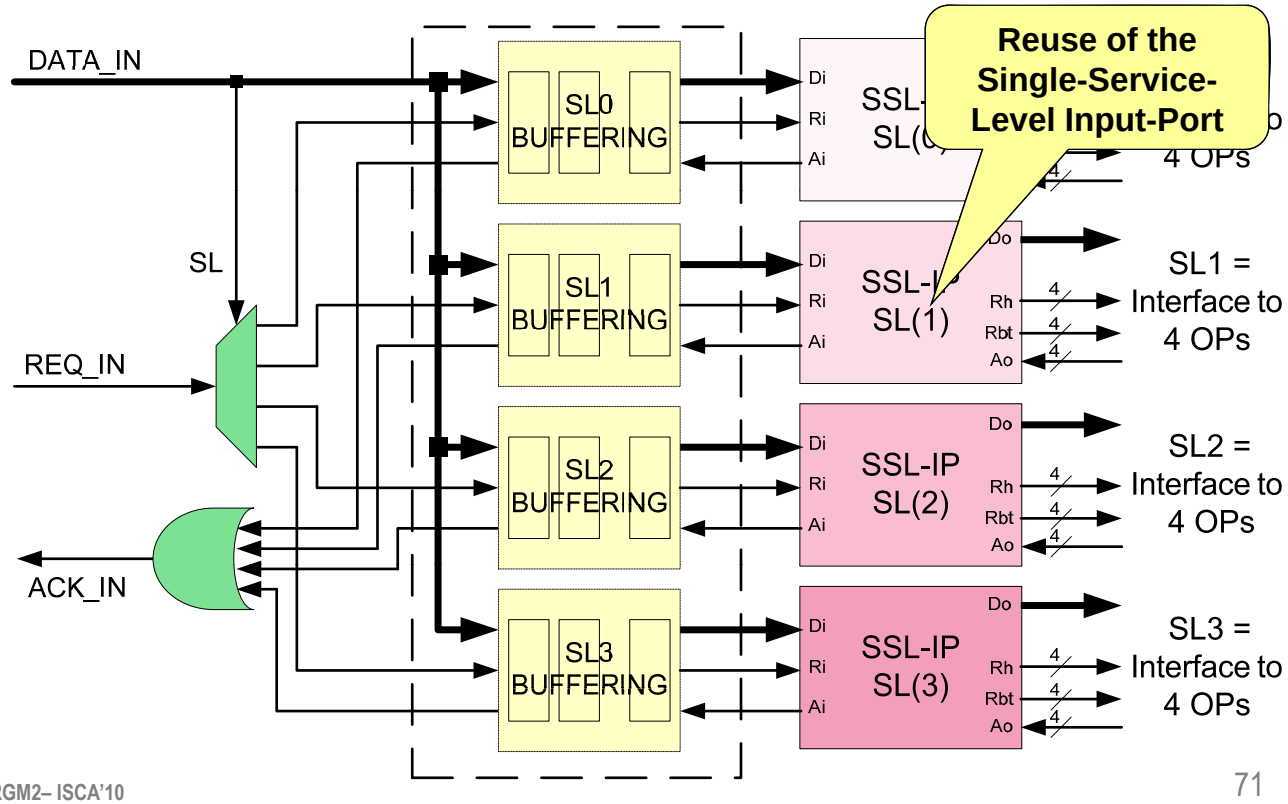




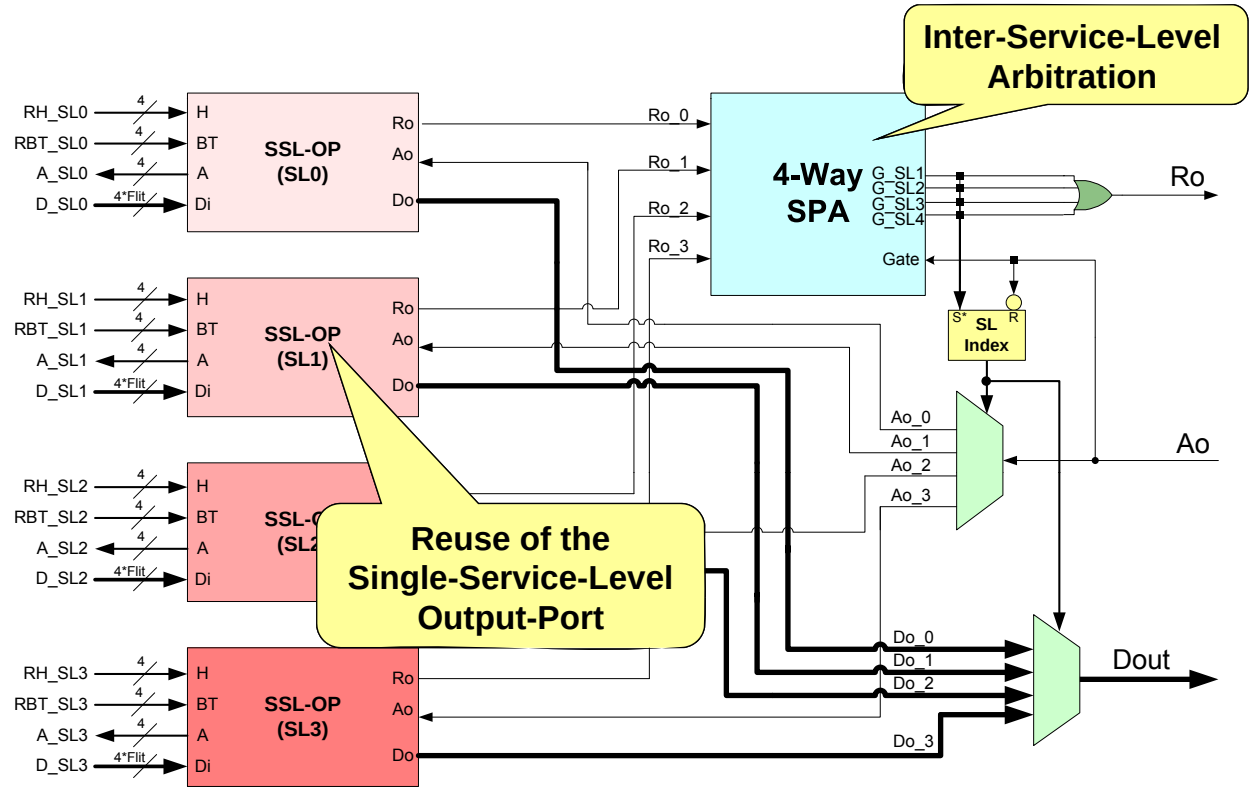
Adding multiple service levels for QoS

- Support different priorities at same time
- Arbitrate packets / flits: Who should go on the link next?
- Examples
  - Cache miss (fetch new line) is high priority
  - Prefetch is low priority

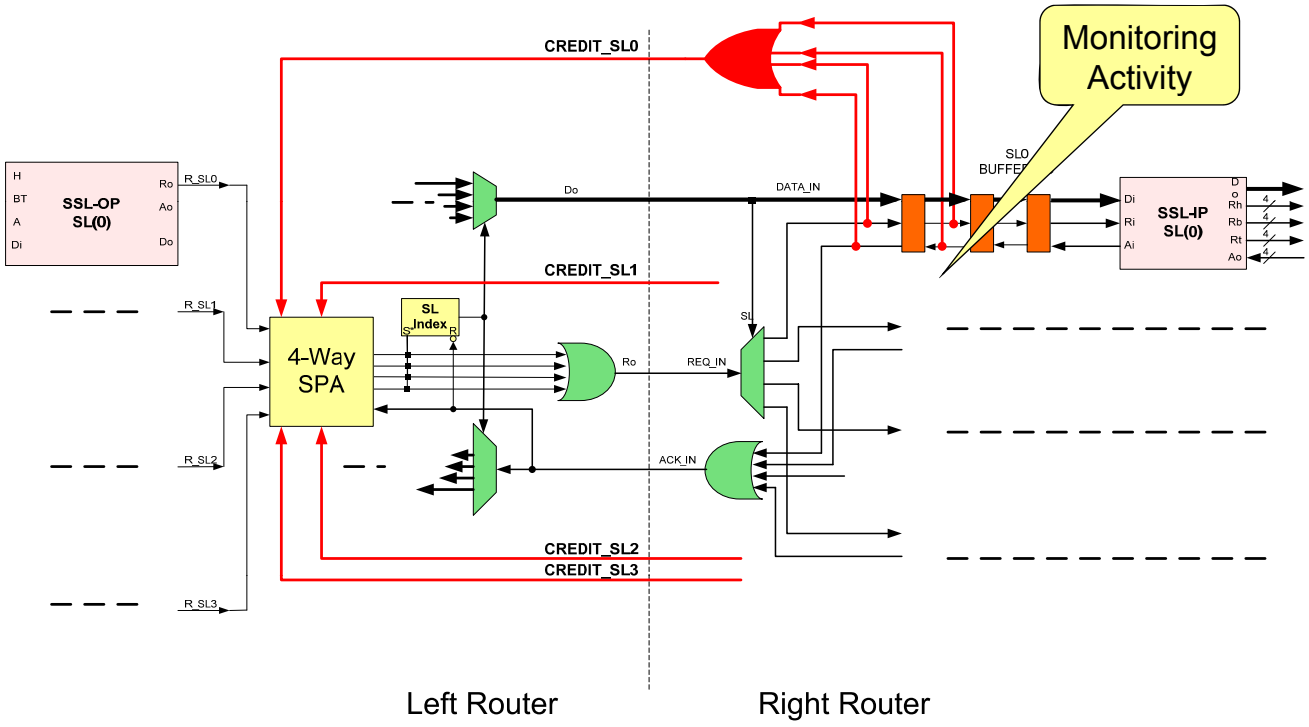
Multi-Service-Level Input-Port



Multi-Service-Level Output-Port



Buffering and Credits



Performance

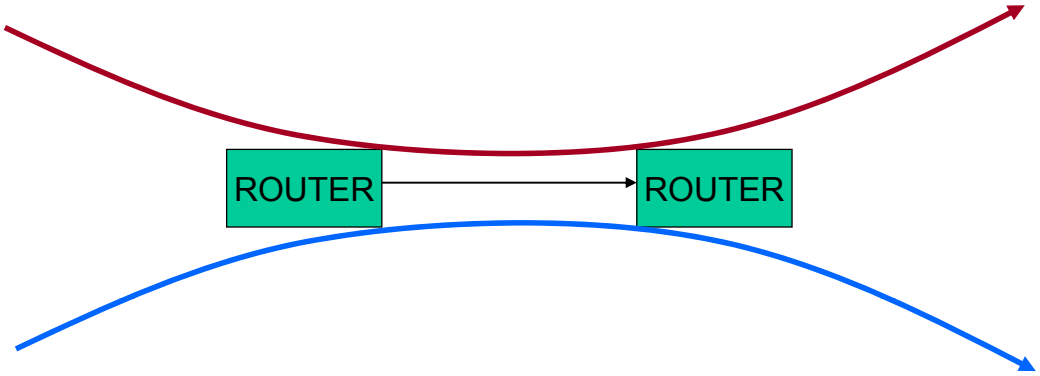
	SYNC router	ASYNC router	
Cell Area	**960,000**	**470,000**	**µm²**
Equivalent Gates (2-in NAND)	**17,500**	**8,500**	**Gates**
Number of transistors	**70,000**	**34,200**	
Number of FFs+Latches	**880**	**620**	
Min Latency (Input to Output)	**3.7 (1)**	**13.0/9.2 ***	**ns (CLK)**
Data Cycle	**14.8 (4)**	**13.3**	**ns (CLK)**
Max Data Rate	**67.6**	**75.2**	**Mflits/s**
Max Clock Frequency	**270.2 ****	**-**	**MHz**

* Latency for async router specified for header / body flits.

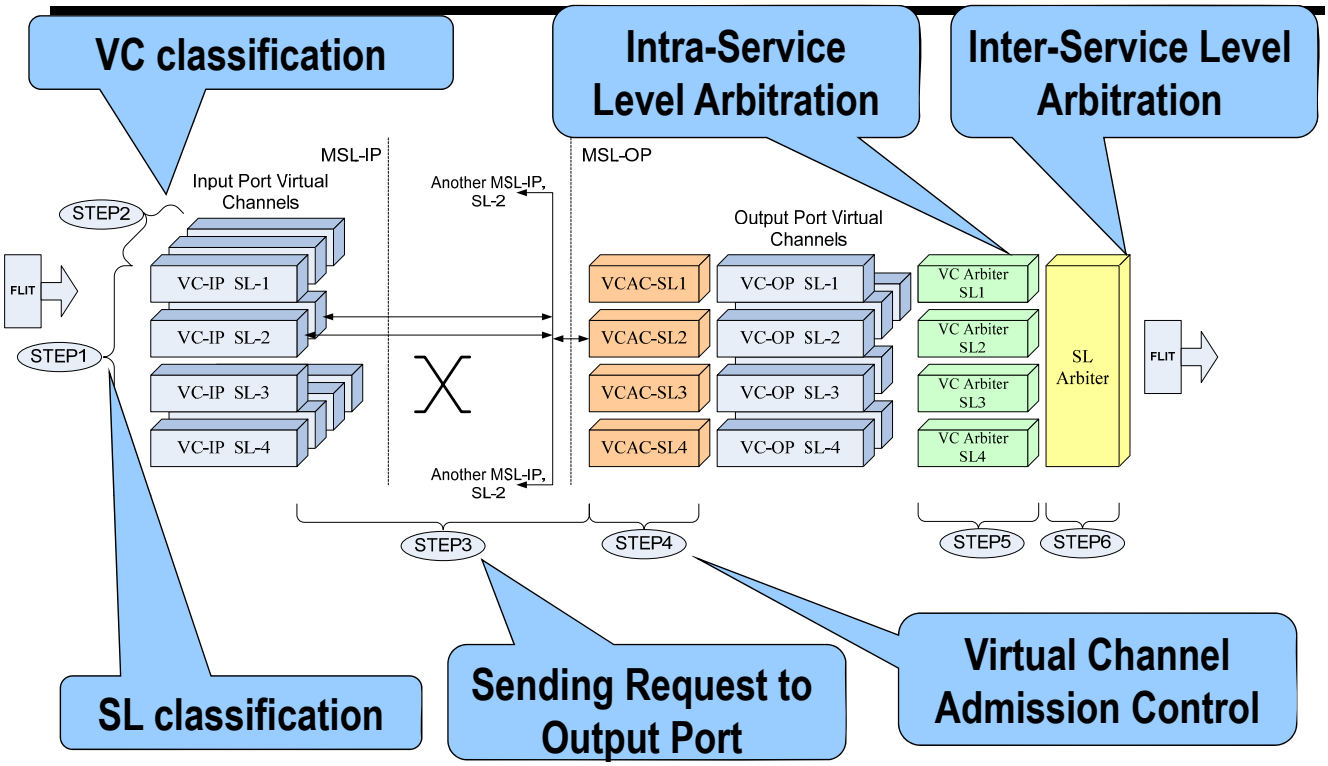
** Synchronous router has a critical path of ~20 FO4 gate delays, matching or outperforming other published results.

Adding virtual channels for even higher QoS

- Support un-related concurrent packets of same priority level
  - So that if one packet stalls, the link is not wasted
  - Flits of multiple packets interleaved on the link



Asynchronous Router 2D Structure



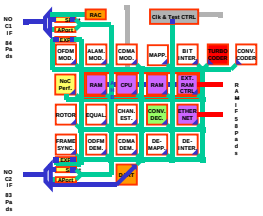
FAUST + ALPIN + MAGALI: A proof-of-concept in silicon

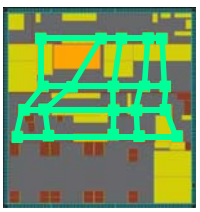
CEA-LETI

Grenoble, France

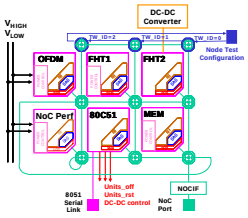
LETI's Research: 3x MPSoC with ANOC

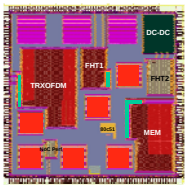
- FAUST (2005)
  - **ST 130 nm – 20 nodes**
  - **ANOC**



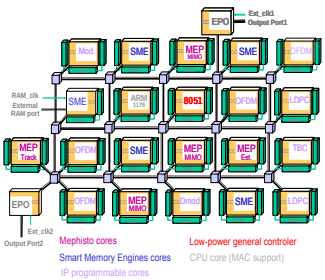


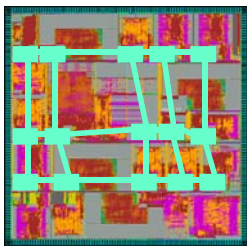
- ALPIN (2007)
  - **ST 65 nm – 9 nodes**
  - **ANOC for low-power**





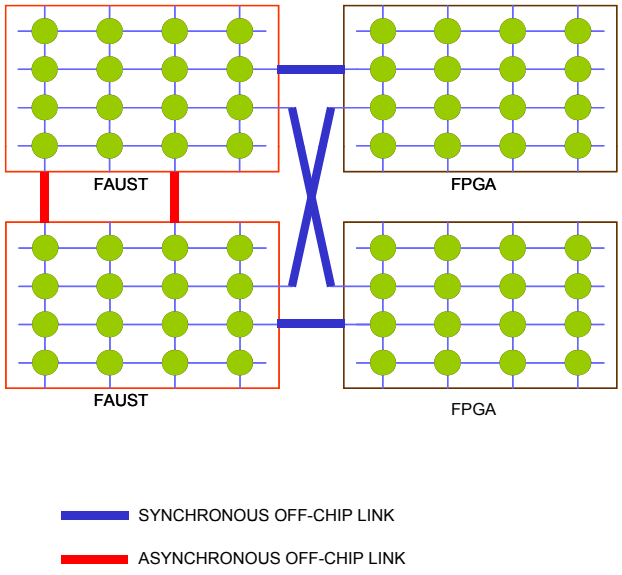
- MAGALI (2009)
  - **ST 65 nm – 15 nodes**
  - **High Performance ANOC**



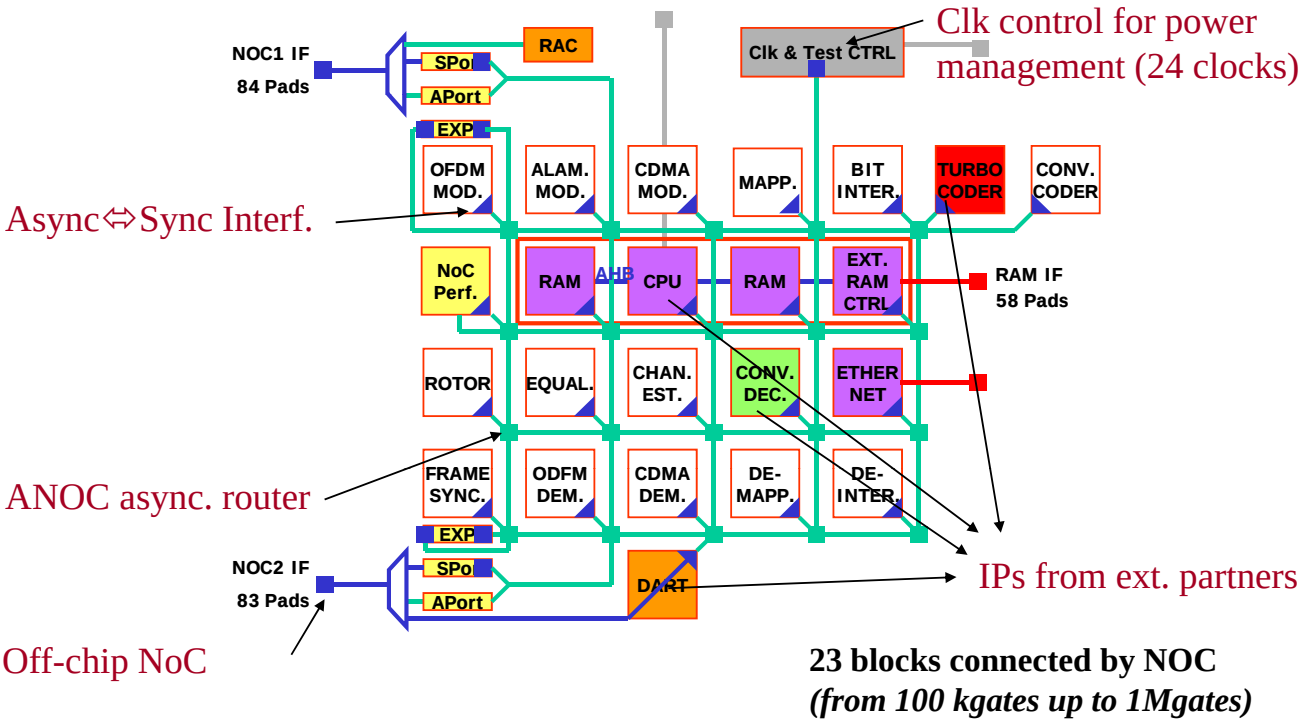


The FAUST environment: NOC off-chip, Async and Sync



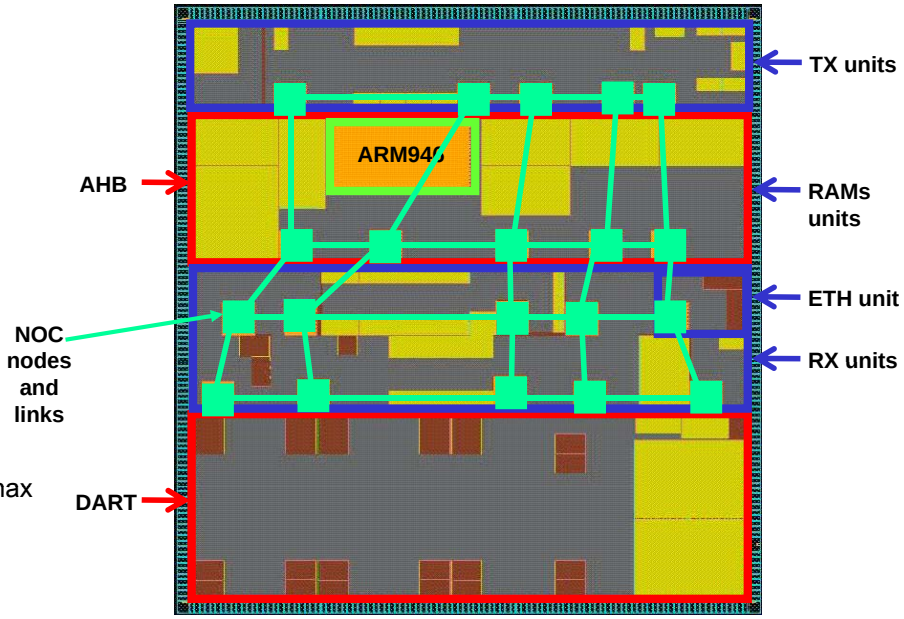


The FAUST architecture

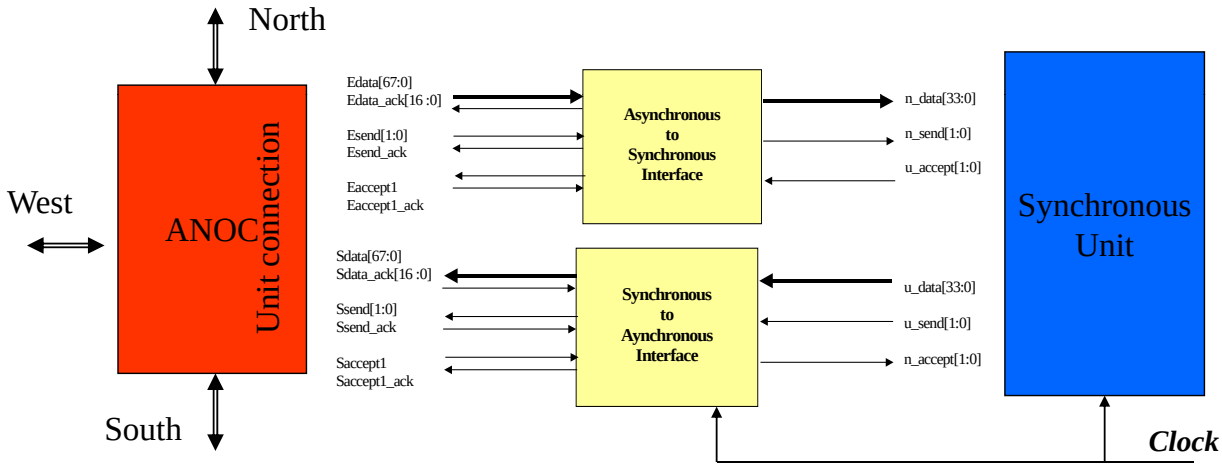


FAUST chip description & Floor-plan

- Tech.: **STM/HCMOS9GPLL** (0.13µ)
- **23 NoC units (166 MHz)**
- RAM blocks : 81
- CPU: **ARM946**
- **4.5 Mgates**
- **275 I/Os**
- Core area = 70 mm^2
- Chip area = 80 mm^2
- Package : **TBGA 420**
- Core power supply: 1.2 V
- I/O power supply: 3.3 V
- Power Consumption : 3 Watts max



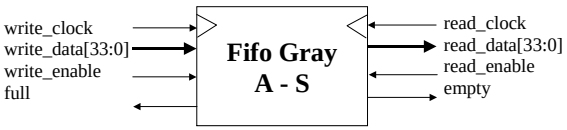
FAUST GALS interface



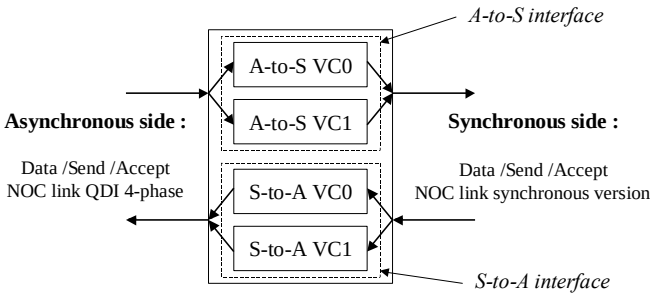
- **The GALS interface: A synchronizer**
- **Uses dual-clock fifo (2 stages)**
  - **Allows sync & async transfers every clock cycles**

FAUST GALS interface

- Design is based on Gray-code Dual-clock fifo's
  - Main objective is to provide a *full std-cell design approach*
  - Modification required for the async side

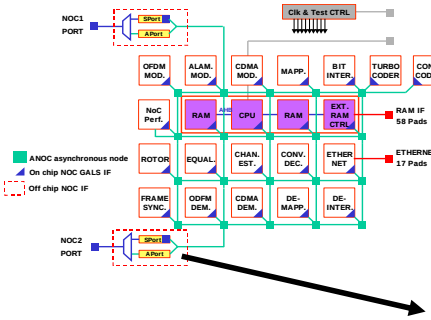


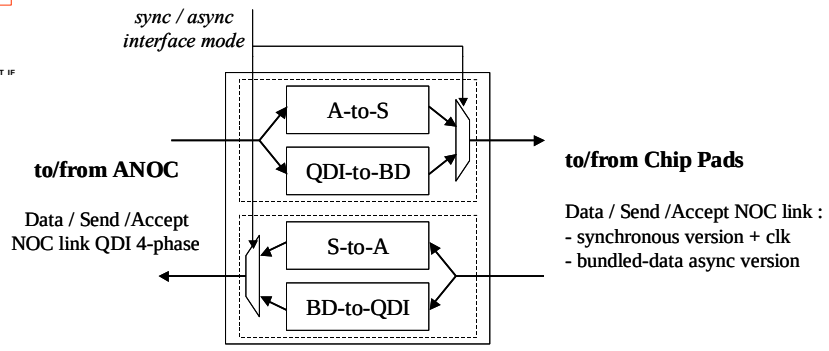
- NOC interface
  - A-to-S + S-to-A fifo's
  - One fifo per Virtual Channel



FAUST external interfaces

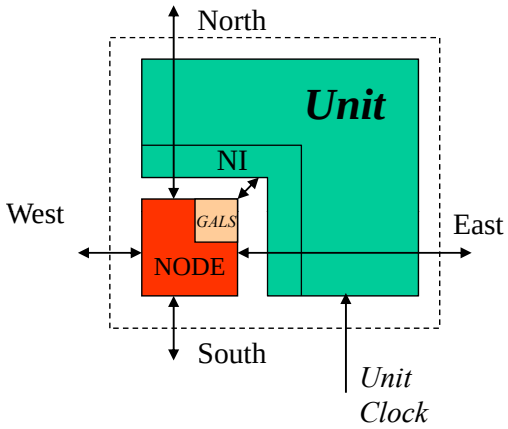
- NOC access in dual-mode : sync / async
  - Good for debug / explore asynchronous external connections
- For async mode, convert 4-rail/4-phase to bundled-data/2-phase





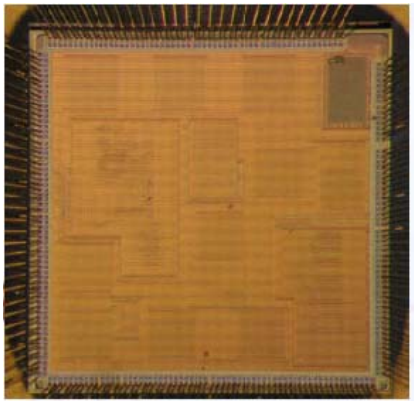
FAUST Performance Results

- ANOC performances (worse case, 5/5 nodes)
  - Per async Node : 150 Mflit/s, 5 ns Latency
  - Per GALS interf. : 120 Mflit/s, 10 ns Latency
- Area results
  - Network Interface (wo config reg.) : 10 kgates
  - ANoC node (requires specific async cells)
    - NoC node : 20 kgates
    - GALS interface : 15 kgates
  - 45 kgates totally per NoC block unit to provide :
    - communication, Quality-of-Service, configurability,
    - robustness & multi-clock domains
  - OK for units with average complexity of about 300 kgates (~15%)
- NoC communication overhead
  - NI credit mechanism + packet headers : ~10% total NoC throughput
  - Virtual channel (low latency packets) : 50 % area of NoC node + GALS IF

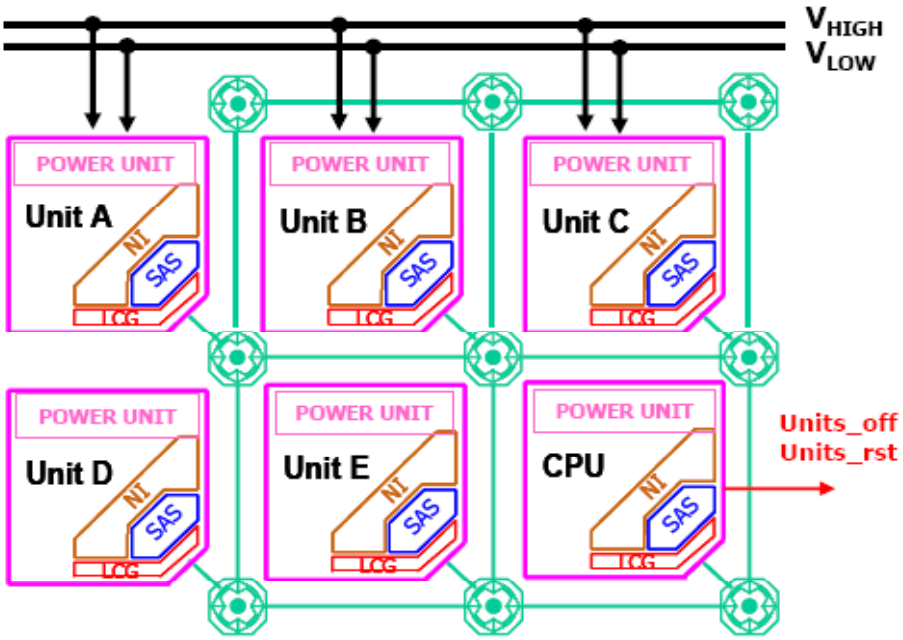


ALPIN

- Claim 1: Async NOC (=GALS SOC) easily enables dynamic voltage and frequency scaling (DVFS)
  - Lower voltage to some modules when slow
  - Power off to some modules to save leakage
  - Sync modules use "pausable clock"
    - When voltage and frequency change, local module clock is paused momentarily
- Claim 2: Routers used lightly
  - Shut off when idle
  - Easy when async
- Based on FAUST
- Another actual chip ☺
  - ST Micro 65nm

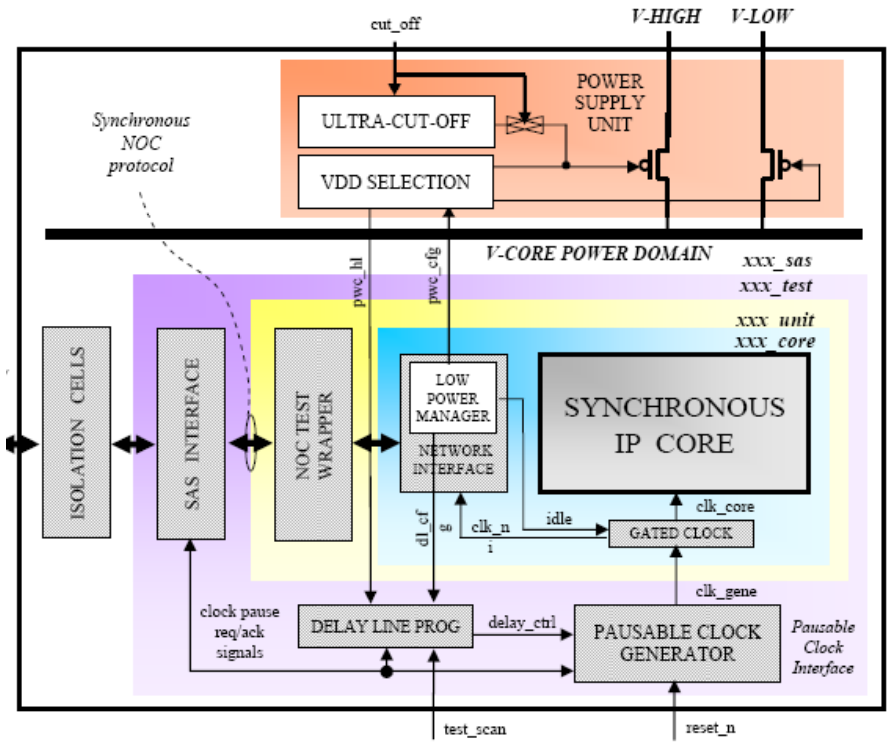


ALPIN



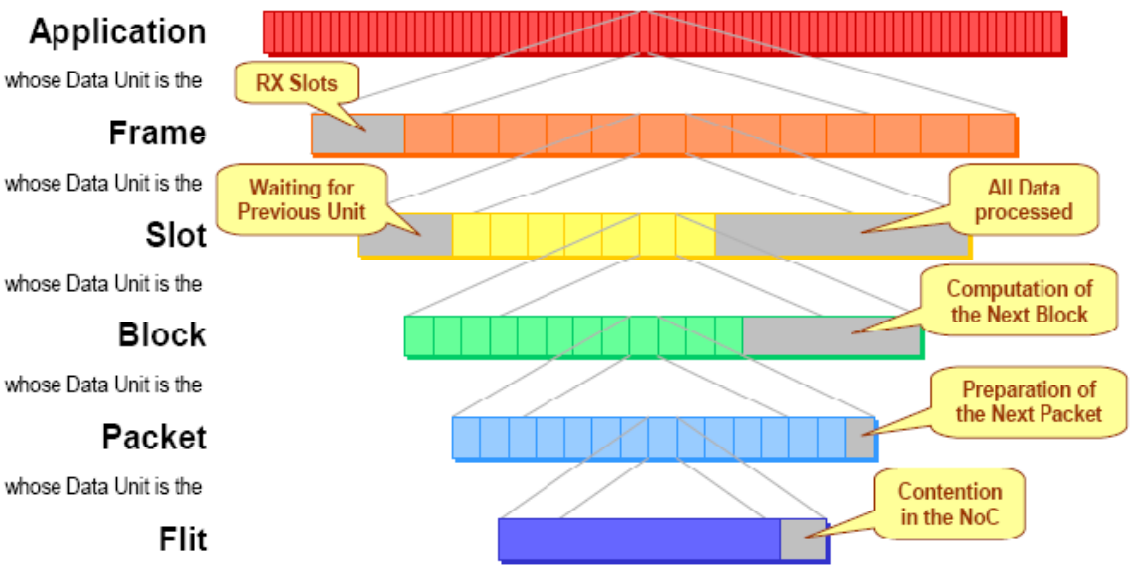
SAS=Sync/Async/Sync synchronizer LCG=Local Clock Gating

ALPIN Unit: Sync core, Async DVFS Wrapper



- **Power Modes:**
  - High
  - Low
  - Changing
  - Retention
  - Off

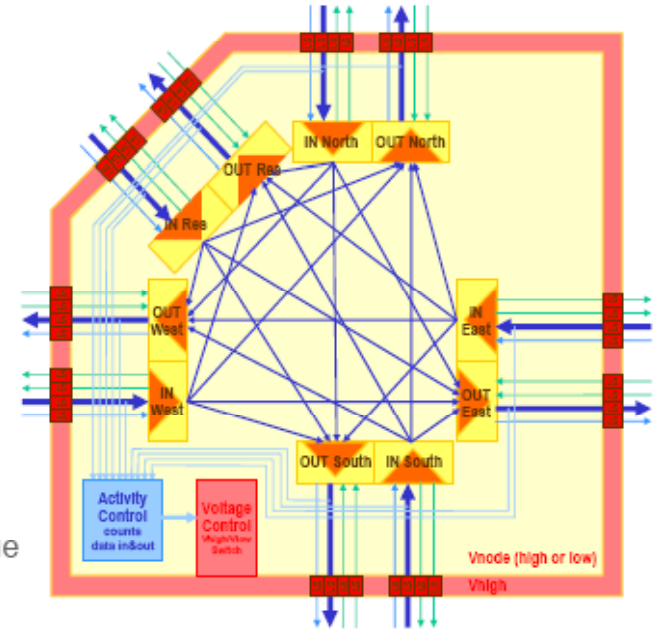
ALPIN Claim 2: Routers are used only lightly



Total idle → 90%. If shut off, save 90% of leakage in routers

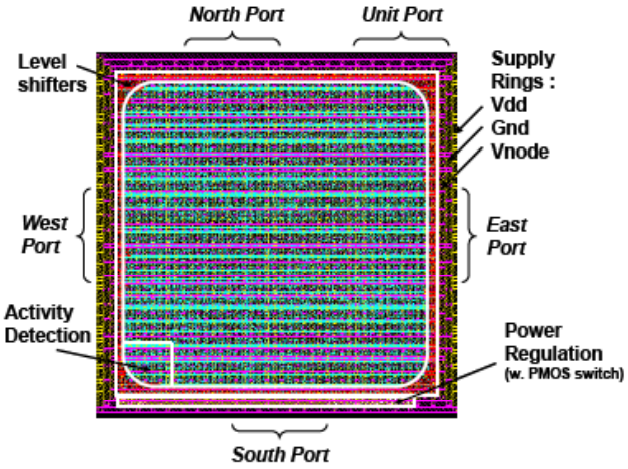
ALPIN auto-off router architecture

- Activity detection
  - QDI I/O monitoring
    - Fork on forward paths
    - C-element on ack. paths
  - Concurrent flows counter
    - BOM on an input: +1
    - EOM on an output: -1
- Power planning
  - Power switch
    - To regulate core voltage according to activity
  - Level shifters on all I/Os
    - To maintain nominal voltage on network links



ALPIN auto-off router layout

- **Area:**
  - Typical unit 0.2 mm^2
  - Core: 85%
  - Level shifters: 13%
  - Activity detection: 2%
  - Power switch: 0%



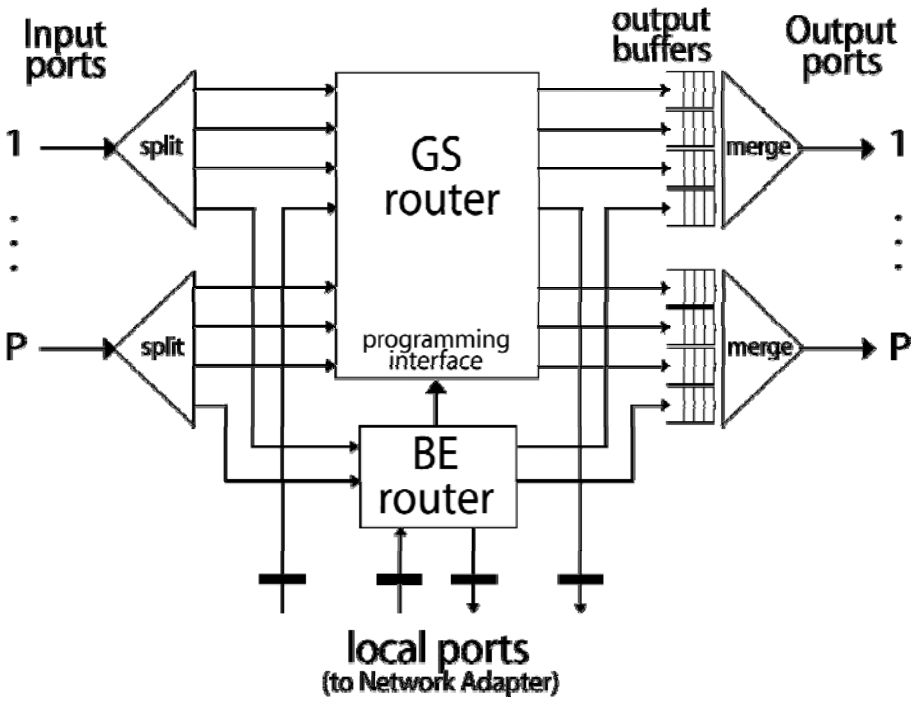
MANGO

Technical University Denmark

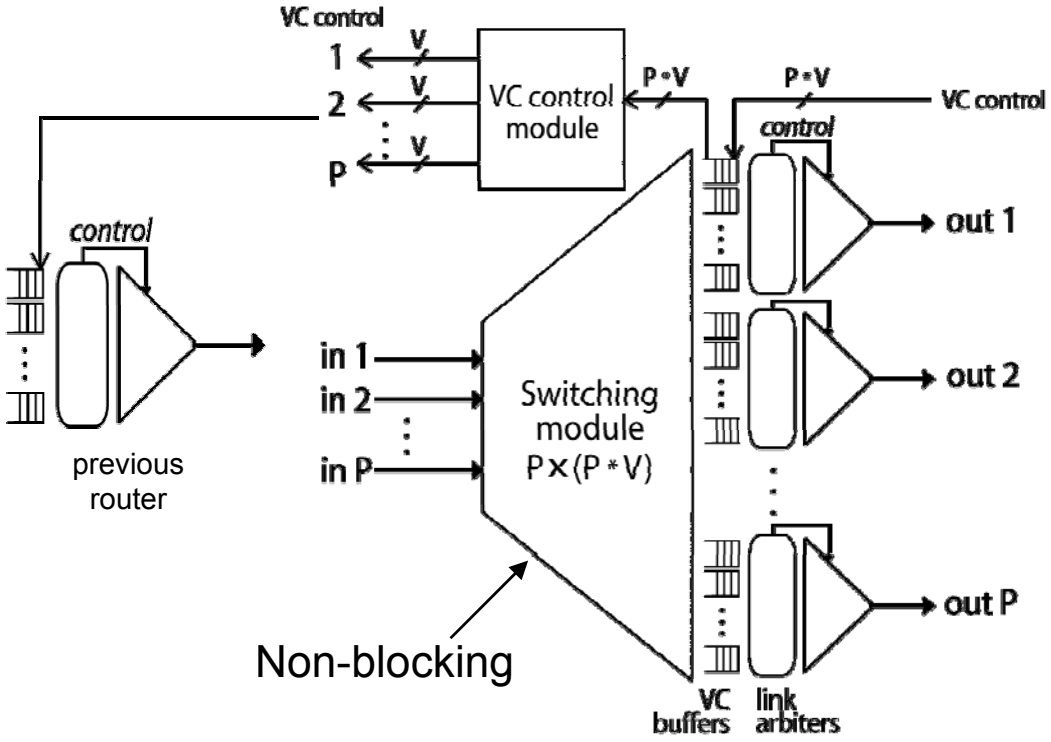
MANGO

- Guaranteed service
  - In addition to ”best effort” service
  - Not the same as 2 service levels
  - Not the same as 2 virtual channels with different priorities
- Connection-oriented service guarantees
  - Allocate resources (reserve VCs) source-to-destination
    - Service levels are applied hop-by-hop
  - Similar to circuit switching
    - Vs. packet switching
- Async / GALS
  - Cf. Sync GS/BE in Philips/NXP Aethereal

MANGO Router



MANGO GS router



MANGO simulation

- 5x5 MANGO router: 8 VCs - 32 bits
  - Connection-oriented GS (fair-share)
  - Connection-less BE
- Clockless circuits 130nm std cells
- Results:
  - 795 MHz (typical) / 515 MHz (worst case)
  - Area: 0.188 mm^2 (pre-layout)

Async Router Conclusions

- Eliminate clocks in the NoC
  - Useful for heterogeneous Multi-Clock-Domain SoCs
  - Useful for multi-voltage domain SoCs
  - Facilitates modularity
  - Helps timing closure of large SoCs
  - Facilitates DVFS
  - Can prioritize or guarantee service
- Handshake may slow traffic
  - Need careful design
- More room for improvement

Summary

- NOCs are for large SOCs
- Large SOCs = multiple clock domains

→ NOCs should be asynchronous

- ANOC enables separating clock and voltage domains
  - Key to low power

Outline

- Part I: Multi-Domain Processors Design Overview (2:00-2:45PM)
  - Multi-domain server, cell phone, and media processors
  - Power management techniques
- Part II: Router Design and Synchronization Issues (2:45-3:30PM)
  - Asynchronous router design
  - Quality of Service and virtual channels in QNoC
- **Part III: Control and Power Management in Presence of Workload Variations (4:00-4:45PM)**
  - **VFI partitioning and voltage assignment**
  - **Workload modeling and dynamic control of multi-VFI designs**
- **Part IV: DVFS in Presence of Process Variations (4:45-5:30PM)**
  - **Impact of process variations on DVFS controller performance**
  - **Technology-driven limits on DVFS controllability**

ISCA-2010 Tutorial #2

Control and Power Management in Presence of Workload Variations

Radu Marculescu

Carnegie Mellon University

radum@cmu.edu