



HAL
open science

Multi-domain Processors: Challenges, Design Methods, and Recent Developments

Radu Marculescu, Ran Ginosar, Diana Marculescu, Stefan Rusu

► **To cite this version:**

Radu Marculescu, Ran Ginosar, Diana Marculescu, Stefan Rusu. Multi-domain Processors: Challenges, Design Methods, and Recent Developments. ISCA tutorial on "Multi-domain Processors: Challenges, Design Methods, and Recent Developments", Jun 2010, Saint Malo, France. inria-00492830

HAL Id: inria-00492830

<https://inria.hal.science/inria-00492830>

Submitted on 17 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.



Multi-domain Processors: Challenges, Design Methods, and Recent Developments

Radu Marculescu
Carnegie Mellon Univ.

Ran Ginosar
Technion

Diana Marculescu
Carnegie Mellon Univ.

Stefan Rusu
Intel Corp.

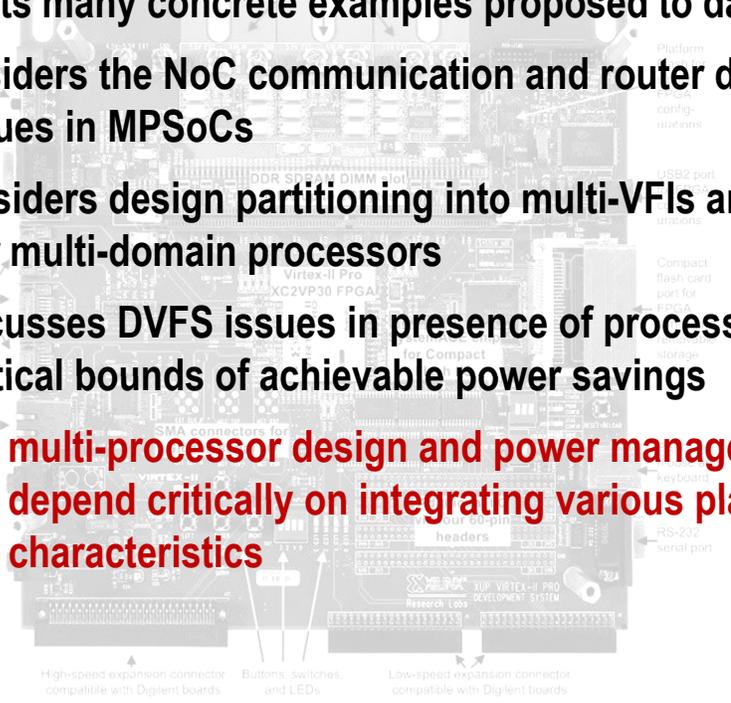
Tutorial #2 : Intl. Symp. on Computer Architecture
Saint Malo, France, June 19, 2010

Outline

- **Part I: Multi-Domain Processors Design Overview (2:00-2:45PM)**
 - ▼ Multi-domain server, cell phone, and media processors *Stefan*
 - ▼ Power management techniques
- **Part II: Router Design and Synchronization Issues (2:45-3:30PM)**
 - ▼ Asynchronous router design *Ran*
 - ▼ 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 *Radu*
 - ▼ 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 *Diana*

Big picture

- Part I discusses the main issues in multi-domain processor design and presents many concrete examples proposed to date
- Part II considers the NoC communication and router design, as well as QoS issues in MPSoCs
- Part III considers design partitioning into multi-VFIs and control policies for multi-domain processors
- Part IV discusses DVFS issues in presence of process variations and theoretical bounds of achievable power savings
- **Low-power multi-processor design and power management techniques depend critically on integrating various platform and application characteristics**



RGM2- ISCA'10

3

ISCA-2010 Tutorial #2

Multi-Domain Processors Design Overview

Stefan Rusu

Intel Corporation

stefan.rusu@intel.com



4

Why Multi-Domain Processors?

- Adapt supply voltage and clock frequency to the underlying circuit needs
 - ▼ Special supply voltage for large cache SRAM arrays
 - ▼ Adapt clock frequency to standard interface needs (e.g. DDR800)

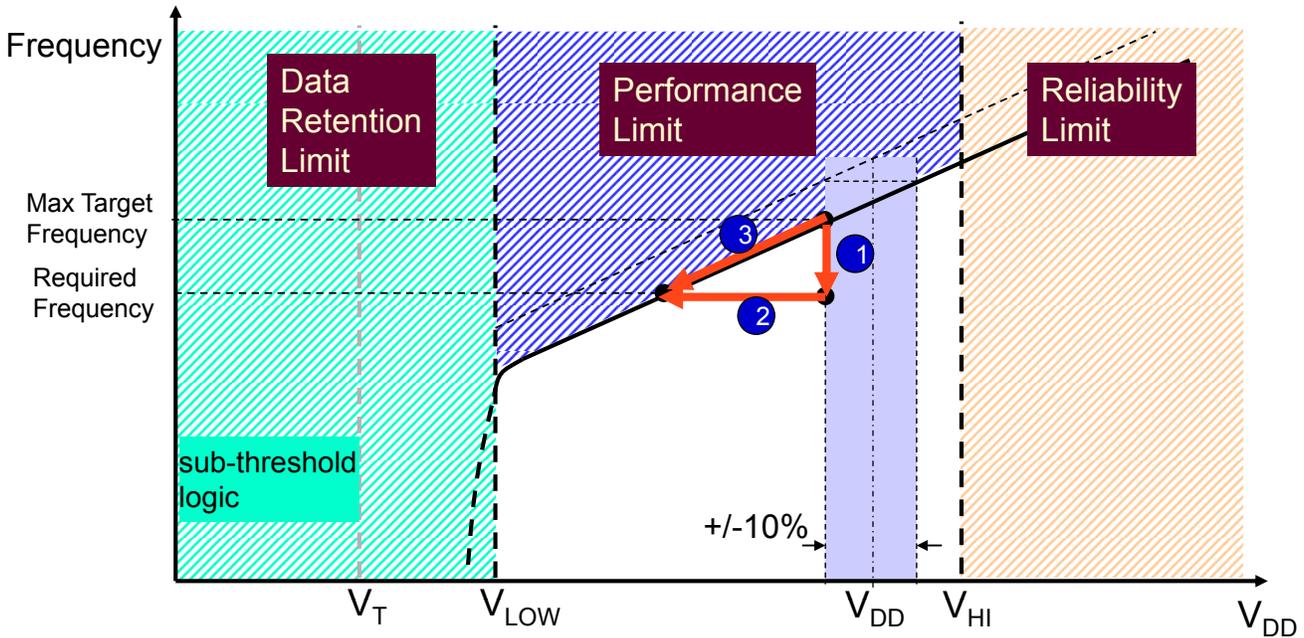
- Reduce operating voltage and frequency to save power in idle or lower performance blocks
 - ▼ Power and clock gating in idle circuit islands
 - ▼ Run last level cache at half frequency

- Clock and voltage knobs are usually used together to enable a cubed power reduction factor
 - ▼ A 10% reduction in both voltage and frequency provides a 30% reduction in power consumption

Multi-Domain Processors Design Overview

- Voltage / frequency scaling basics
- Multi-domain server processors
 - ▼ Power gating
 - ▼ Core and cache recovery
 - ▼ Split vs. connected supplies
 - ▼ Globally Asynchronous, Locally Synchronous (GALS)
- Cell phone processors
- Media processors
- Dual voltage supply at the cell level
- Future directions
- Summary

Voltage and Frequency Scaling



- Case1 - Fixed V_{DD} , Frequency Scaling: Linear Power Reduction
- Case2 - Fixed MHz, V_{DD} Scaling: Square Power Reduction
- Case3 - Voltage and Frequency scaling: Cubic Power Reduction

RGM2- ISCA'10

[J. Rosal, ISSCC 2006] 7

Core Power Management States

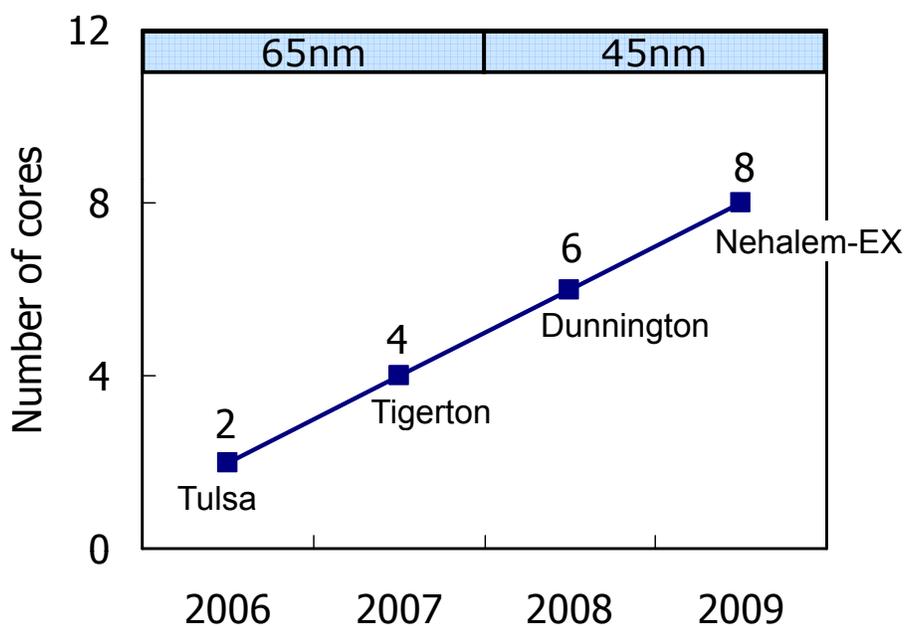
	C0 HFM	C0 LFM	C1/C2	C4	C6
Core voltage					
Core clock			OFF	OFF	OFF
PLL				OFF	OFF
L1 caches			flushed	flushed	off
L2 caches				Partial flush	off
Wakeup time	active	active	<1us	<30us	<100us
Power					

Modulating the processor core voltage and frequency enables lower power states

RGM2- ISCA'10

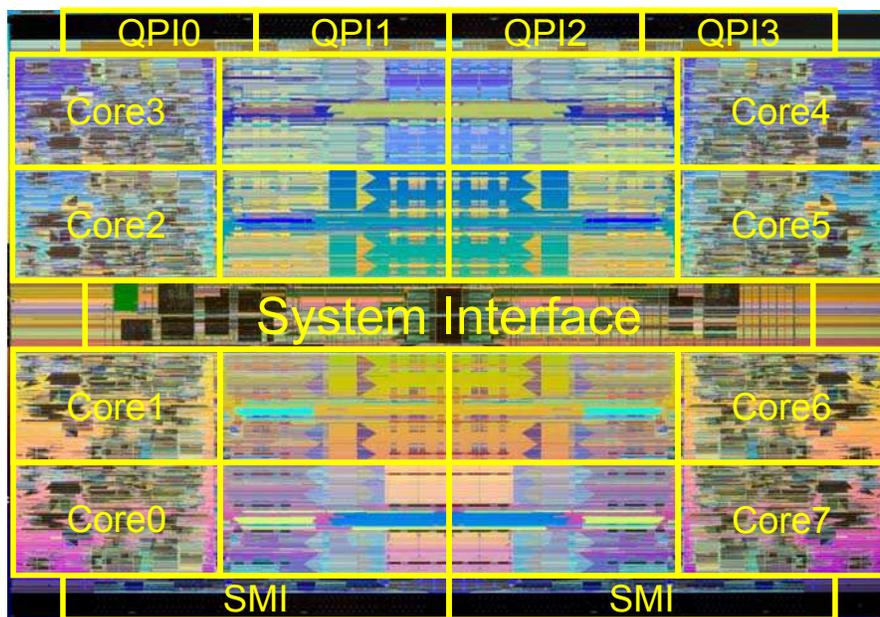
[Gerosa, A-SSCC 2008] 8

Xeon® EX Multi-Core Scaling Trend



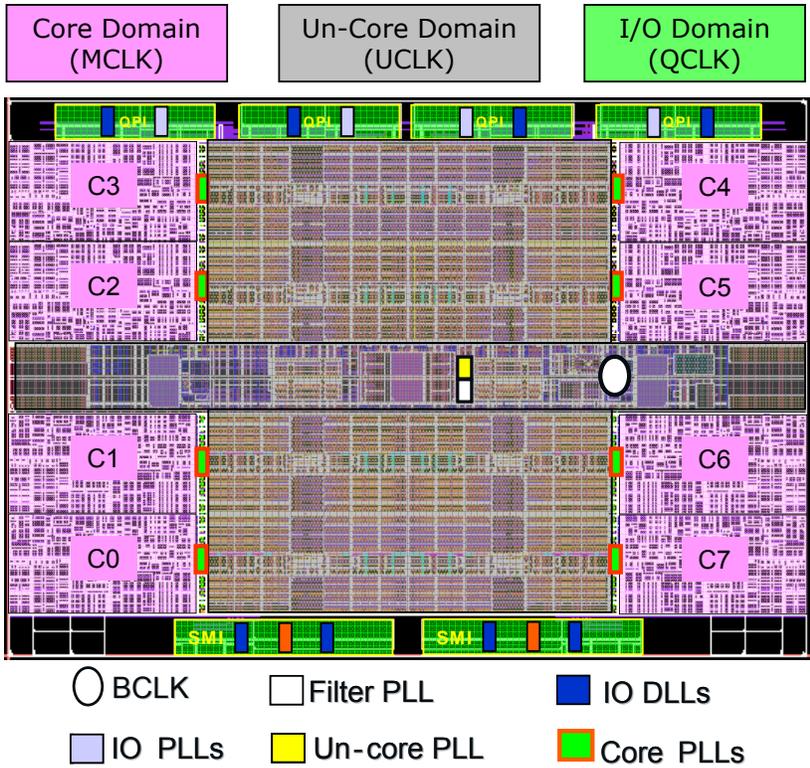
Two additional cores every year

8-Core Xeon® Processor



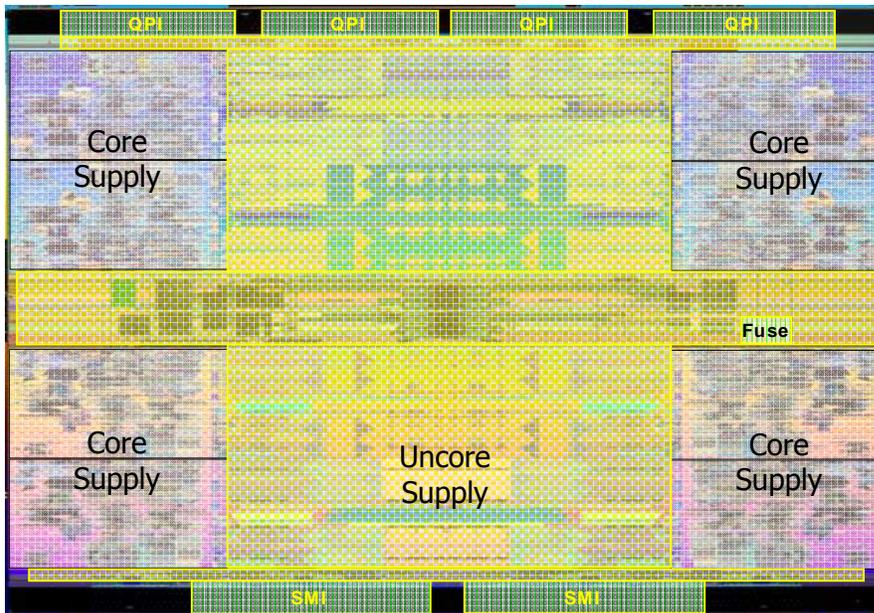
The largest device count reported for a microprocessor
2.3B transistors

8-Core Xeon® Processor Clock Domains



- 3 primary clock domains:
 - ▾ Core
 - ▾ Un-core
 - ▾ I/O
- 16 PLLs & 8 DLLs
 - ▾ Single system clock input (BCLK)

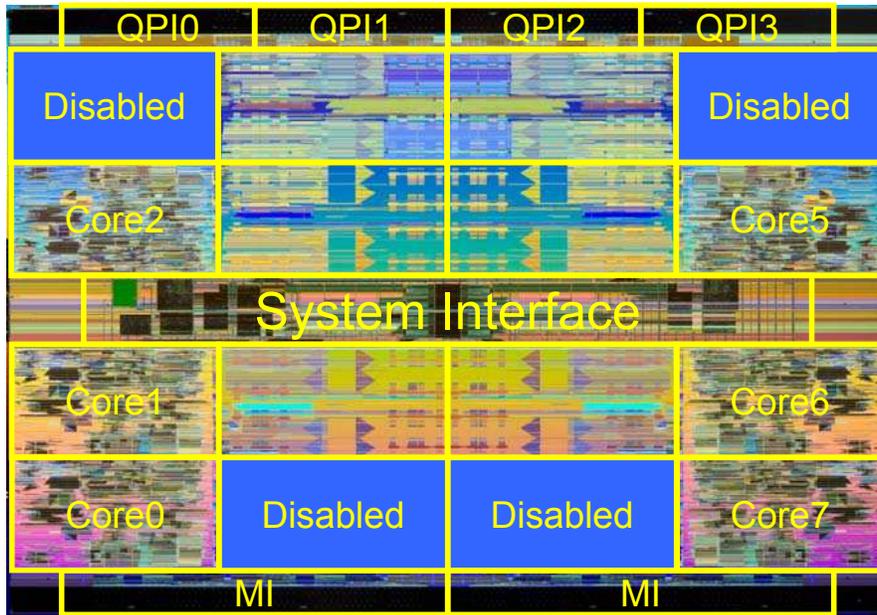
8-Core Xeon® Processor Voltage Domains



- I/O Domain
1.1V
fixed
- Un-Core Domain
0.9-1.1V
fixed
- Core Domain
0.85-1.1V
variable

Multiple voltage domains minimize power consumption across the core and uncore areas

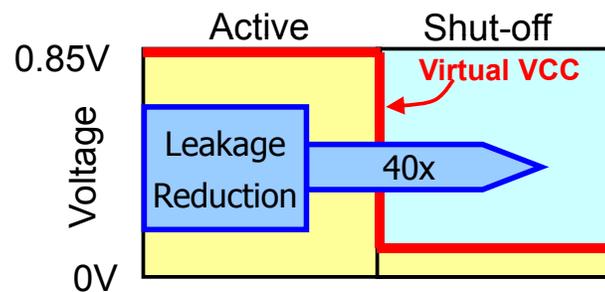
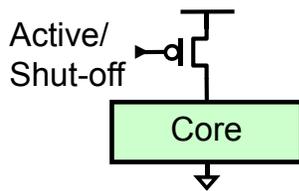
Core and Cache Recovery Example



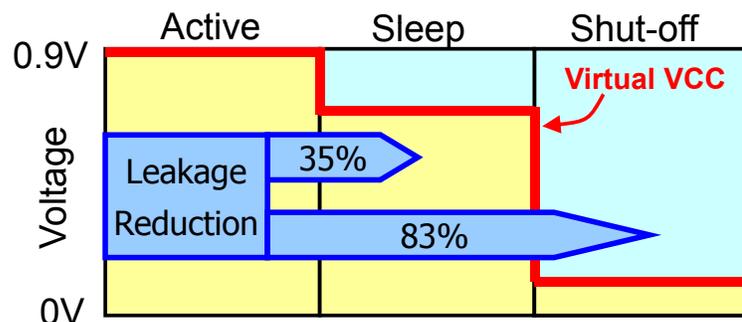
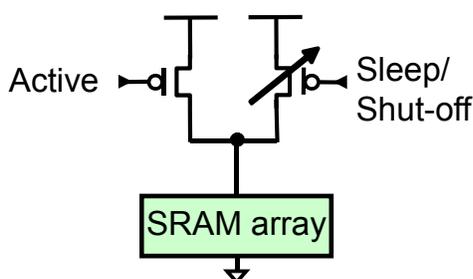
Disabled 2 cores and 2 cache slices

Minimize Power in Disabled Blocks

- Disabled cores ► Power gated

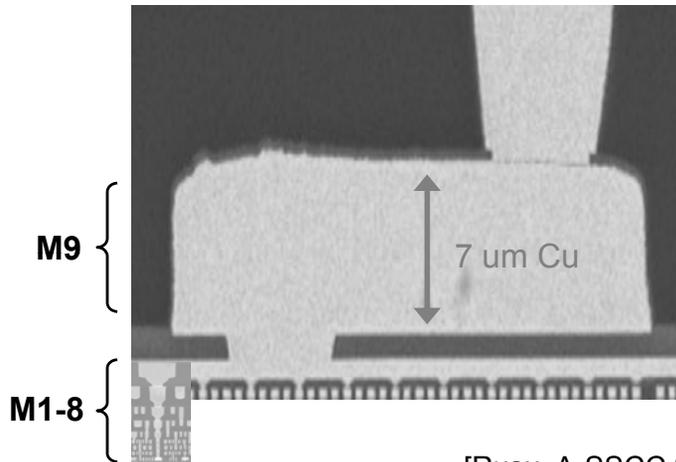
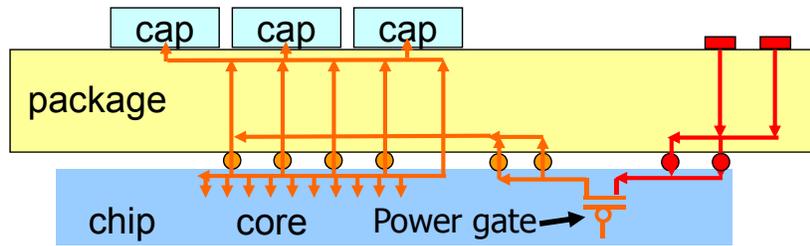


- Disabled cache slices ► All major arrays in shut-off



Core Power Gating

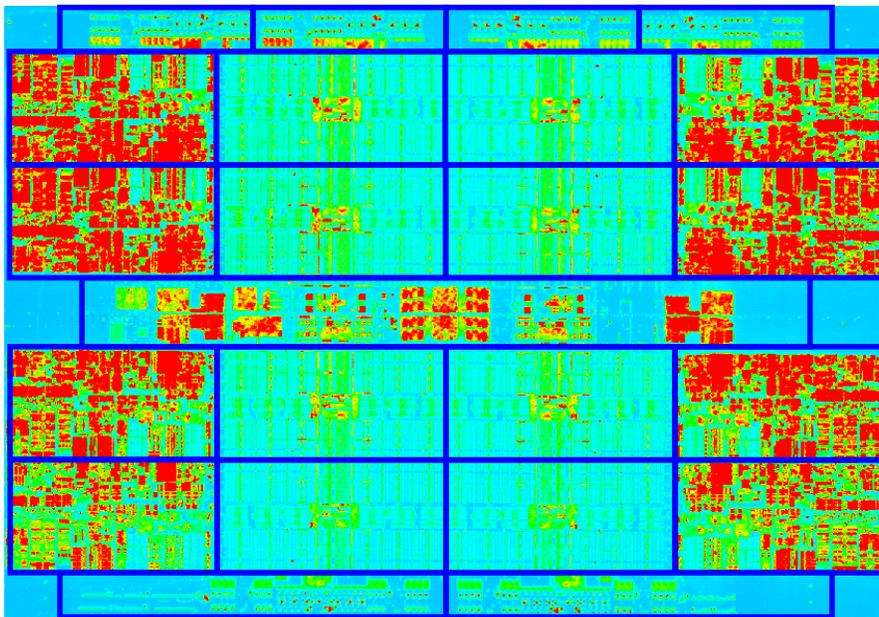
- Resistance target: less than 1% performance loss
- New package-like metal layer on silicon was developed
- M9 has ~10X lower resistance than M8



[Rusu, A-SSCC 2009] 15

RGM2- ISCA'10

Core and Cache Recovery – Infrared Image

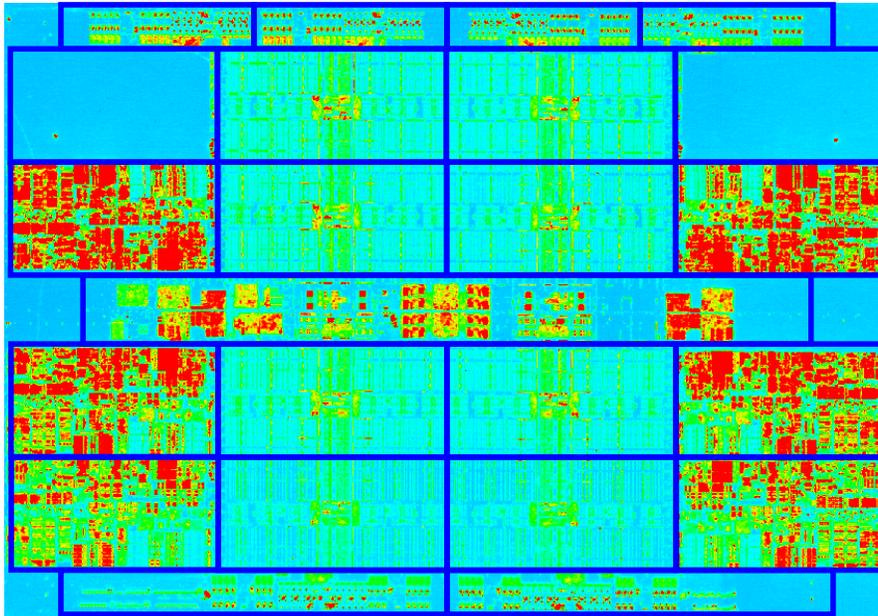


All cores and cache slices are enabled

[Rusu, ISSCC 2009] 16

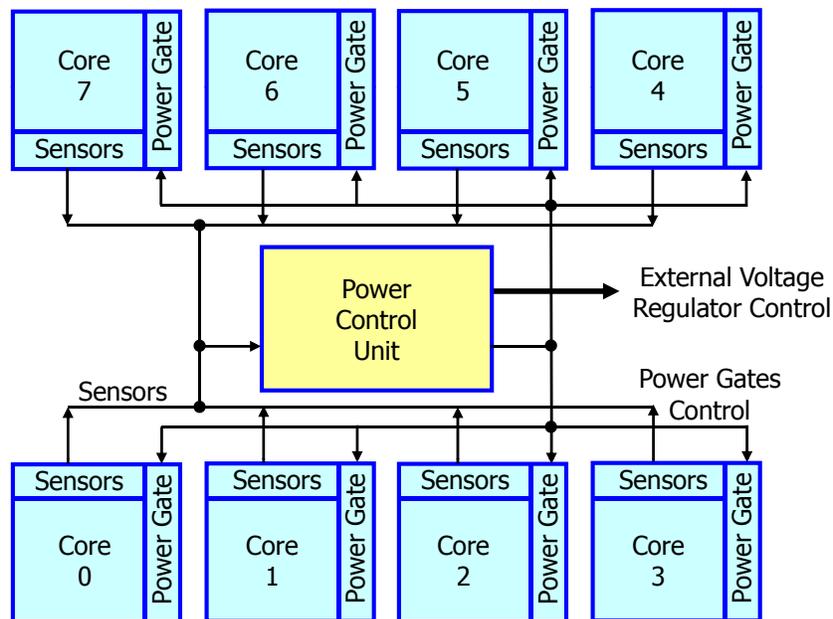
RGM2- ISCA'10

Core and Cache Recovery – Infrared Image



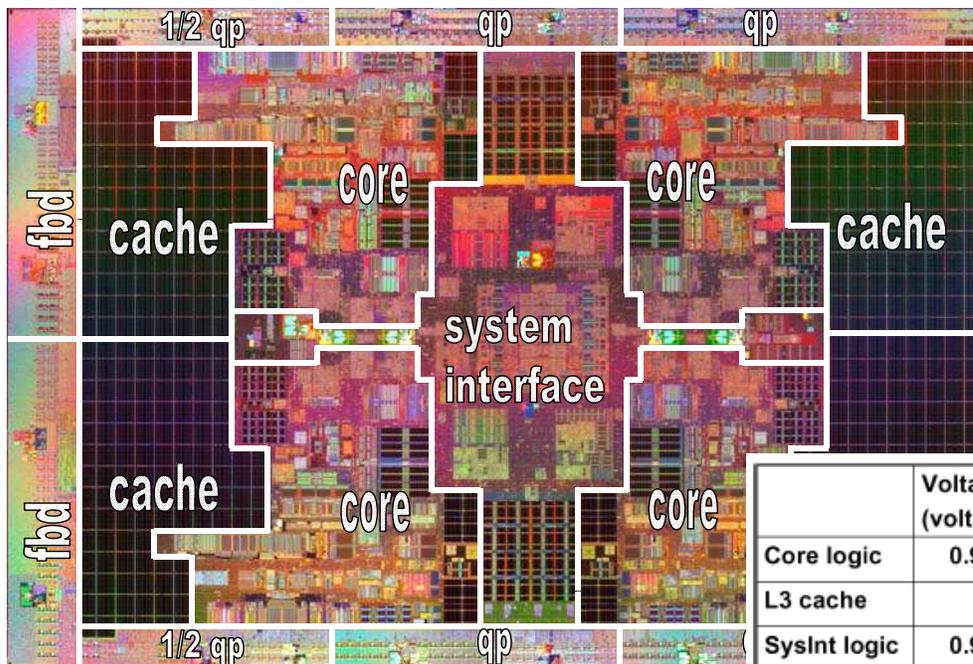
Shut-off 2 cores (top row) and 2 cache slices (bottom row)
Disabled blocks are clock and power gated

Power Management Unit



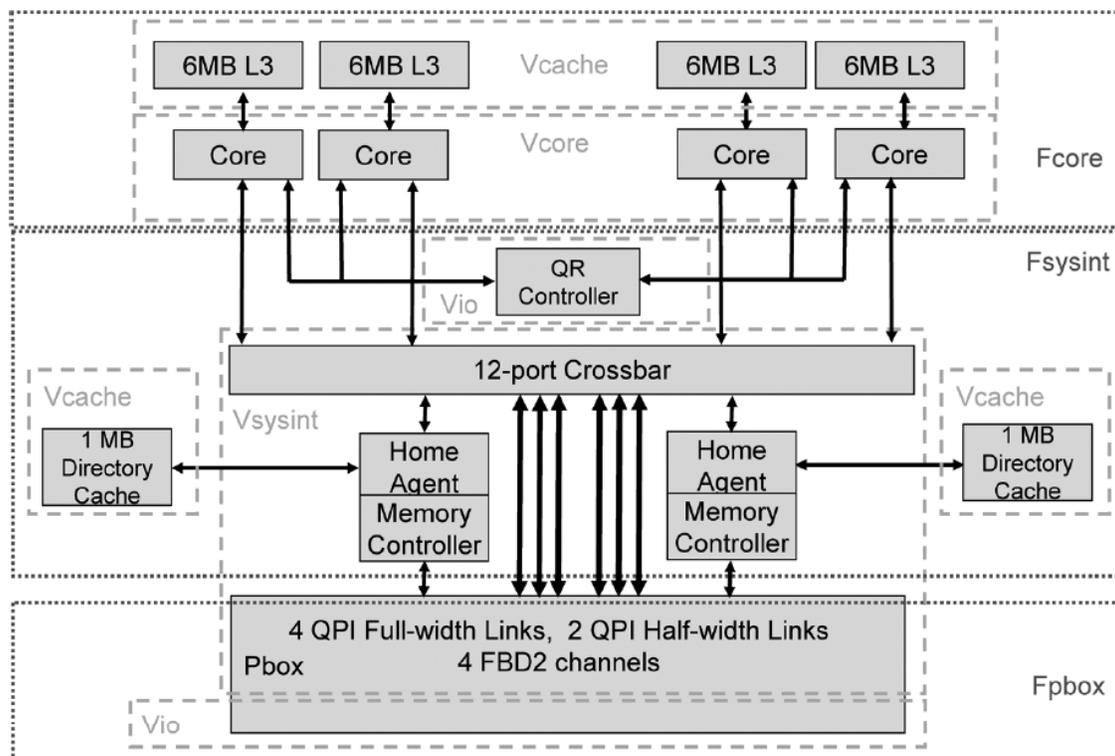
PMU controls processor voltage and frequency based on compute loading and thermal data

65nm Itanium® Processor Domains

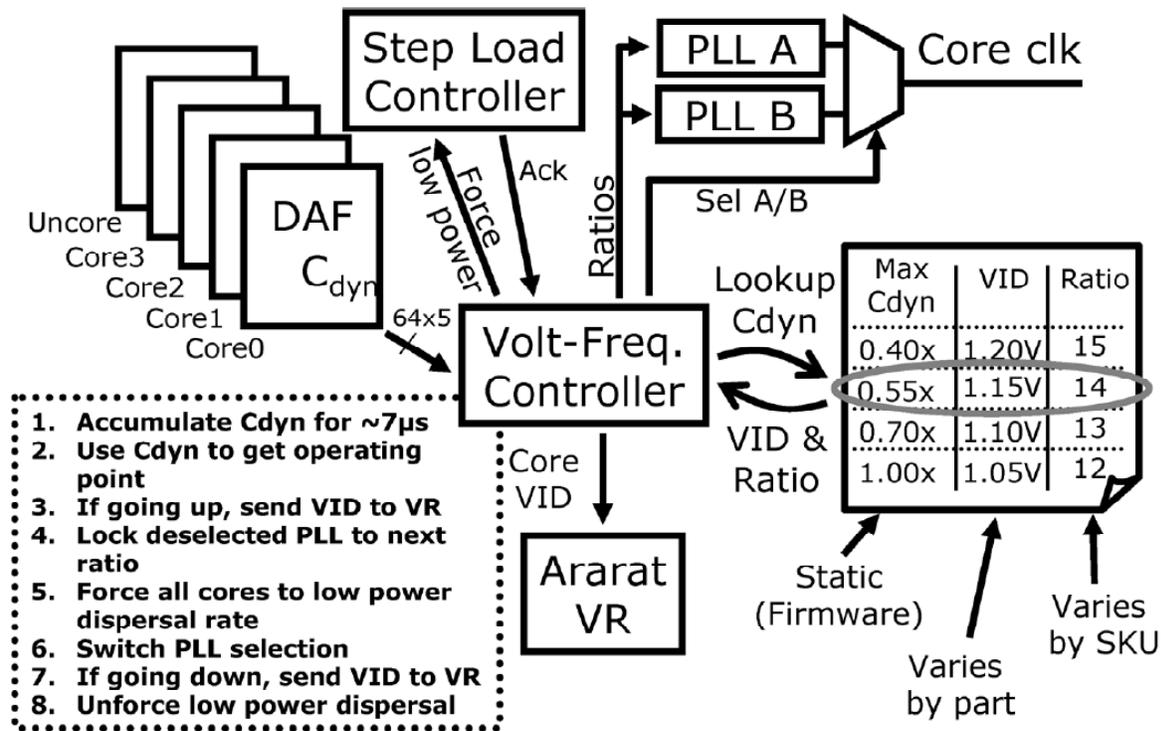


	Voltage (volts)	Frequency (GHz)
Core logic	0.9-1.2	Variable
L3 cache	1.1	Variable
SysInt logic	0.9-1.2	2.4
IO logic	1.1	2.4
QR logic	1.1	0.8

65nm Itanium® Processor Domains



Adaptive Voltage and Frequency Control



RGM2- ISCA'10

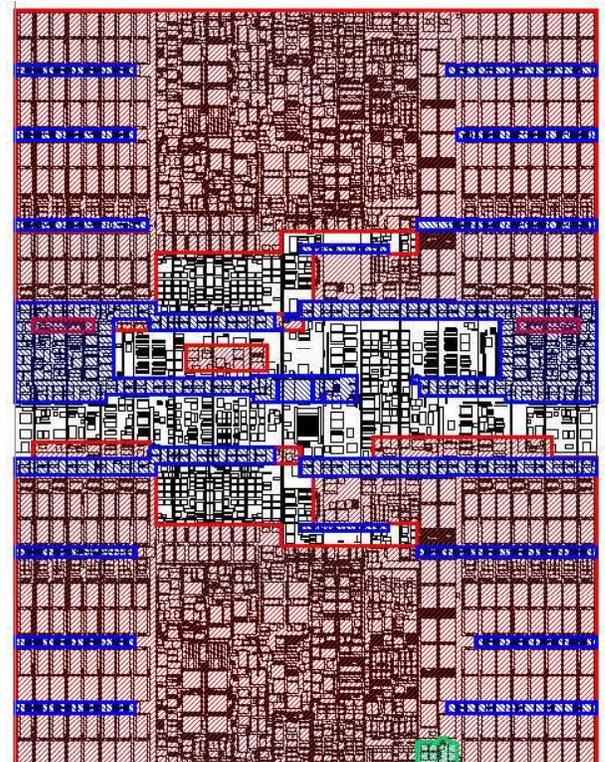
[Stackhouse, JSSC 2009] 21

IBM POWER6 Multi-Rail Design

- POWER6 infrastructure contains 4 voltage domains

Rail	Purpose	Plot Color
VDD	Logic	All
VCS	Array	Red
VIO	IO, PLL, MC	Blue
VSB	Power-up	Green

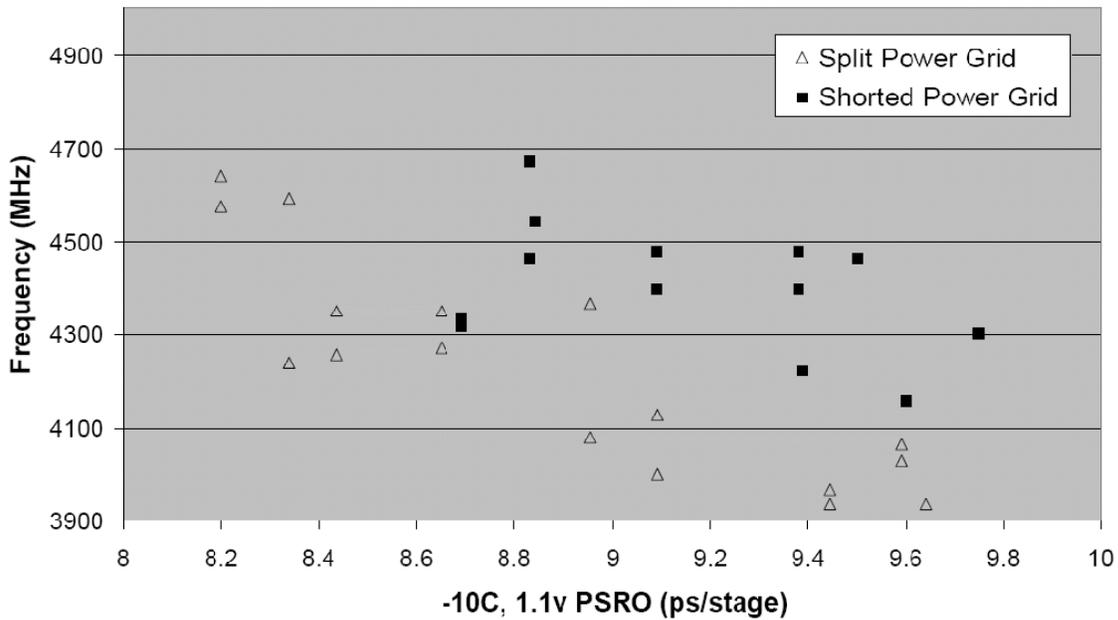
- Multi-rail power grid defined based on macro current requirements and iterative IR analysis of each rail.
- Voltage domain of macros and global signals explicitly specified in RTL and validated by checking tools.



RGM2- ISCA'10

[J. Friedrich, ISSCC 2007] 22

Split vs. Connected Core Supplies

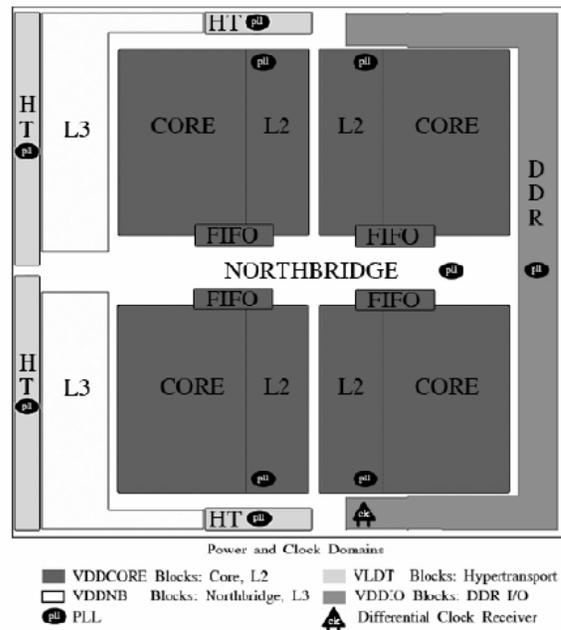


Performance sort ring oscillators (PSRO) show that the connected power chips exhibit a 3-5% fmax improvement

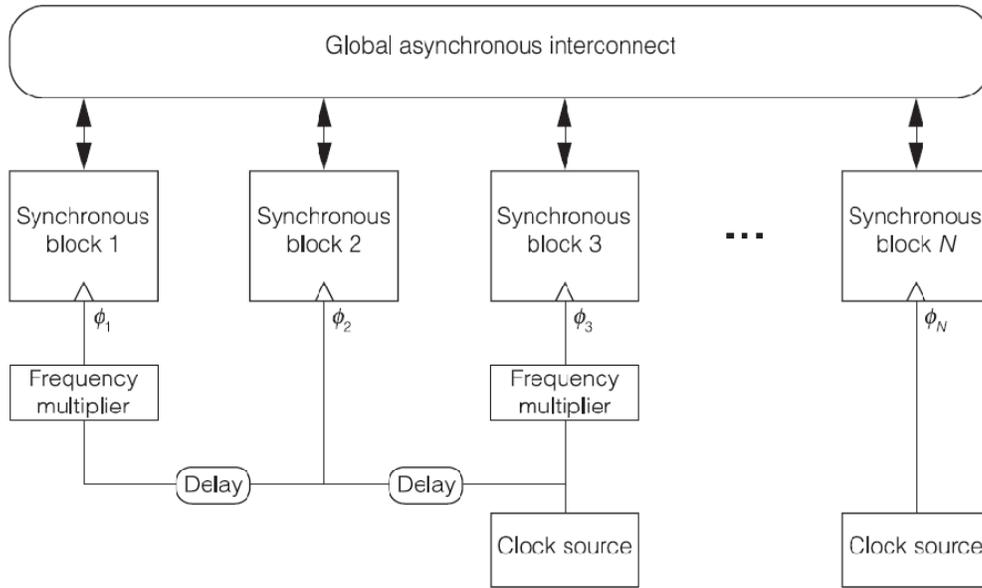
AMD Barcelona Processor Voltage Domains

Multiple supplies for power optimization and isolation

- VDDCORE: 0.8V-1.4V
 - Core and L2: 2.0GHz and up
- VDDNB: 0.8V-1.4V
 - Northbridge and L3: 75% of core
- VLDT: 1.2V
 - HyperTransport links
- VDDIO: 1.8V (VTT:0.9V)
 - DDR I/O
- VDDA: 2.5V
 - PLLs (10 across the die) + Thermal

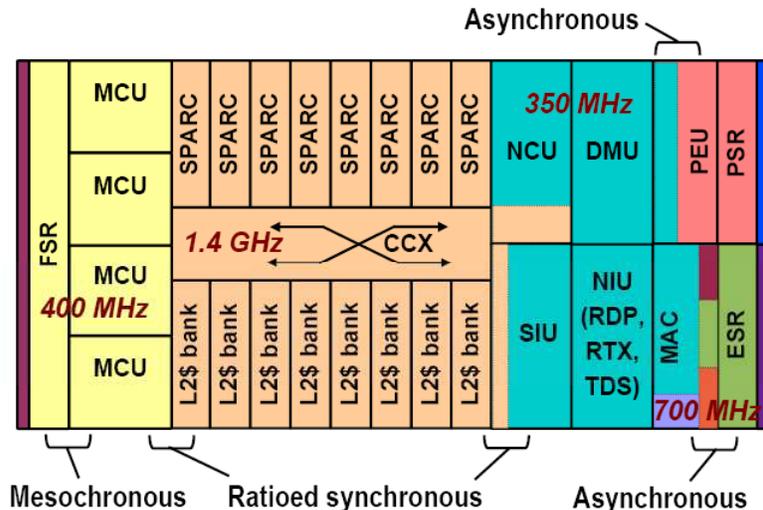


Globally Asynchronous, Locally Synchronous (GALS)



GALS methodology is a natural approach for SoC design, allowing the integration of independently designed blocks operating at different frequencies

Sun/Oracle Niagara2 Clocking

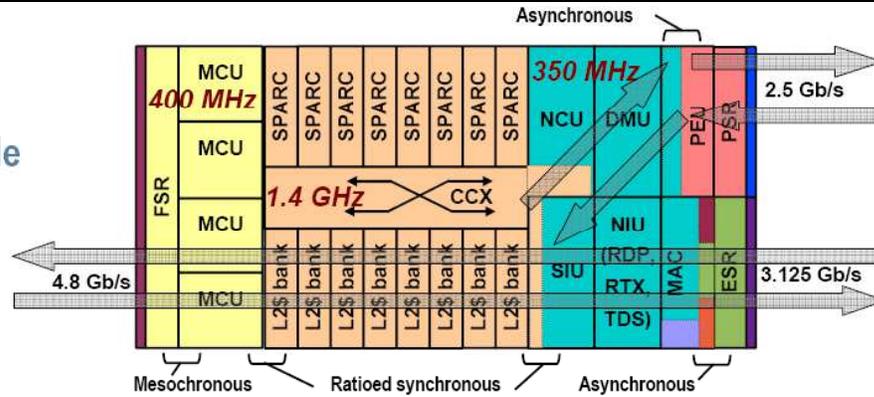


REF	133/167/200 MHz
CMP	1.4 GHz
IO	350 MHz
IO2X	700 MHz
FSR.refclk	133/167/200 MHz
FSR.bitclk	1.6/2.0/2.4 GHz
FSR.byteclk	267/333/400 MHz
DR	267/333/400 MHz
PSR.refclk	100/125/250 MHz
PSR.bitclk	1.25 GHz
PSR.byteclk	250 MHz
PCI-Ex	250 MHz
ESR.refclk	156 MHz
ESR.bitclk	1.56 GHz
ESR.byteclk	312.5 MHz
MAC.1	312.5 MHz
MAC.2	156 MHz

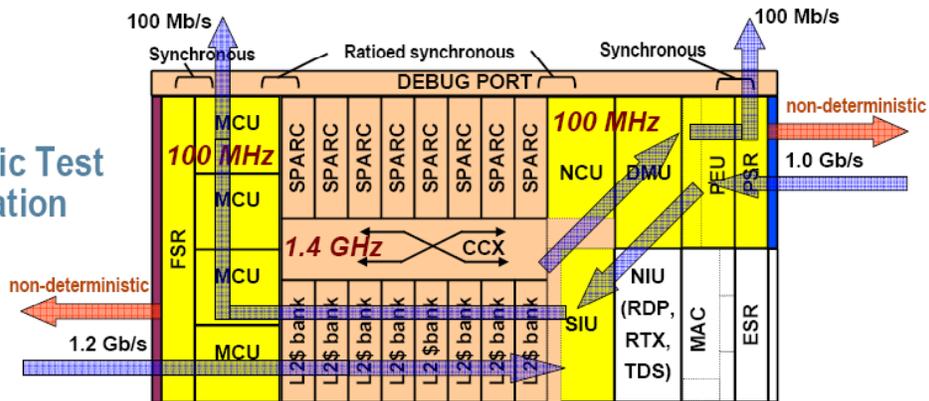
SoC processors must handle multiple clock domains for the various external interfaces

Deterministic Test Mode

Mission Mode Operation



Deterministic Test Mode Operation



RC.....

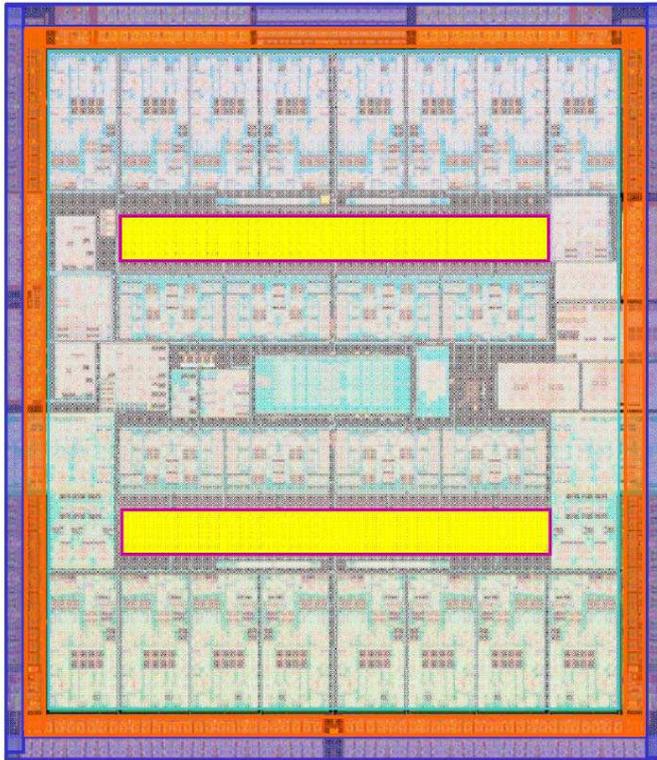
27

Sun/Oracle Rainbow Falls SoC Processor



- TSMC 40nm process, 11 ML
- 16 8-threaded SPARC[®] cores
- 16KB 8-way Icache
- 64-entry ITLB
- 8KB 4-way Dcache
- 128-entry DTLB
- Enhanced multiply/add FGU and crypto per core
- Unified 6MB 16-bank 24-way L2 cache
- Hierarchical crossbar
- 4 DDR3 channels at 6.4Gbps
- 6 coherency links at 9.6Gbps
- 2x8 PCIe 2.0 at 5GTS
- 2x10G XAUI Ethernet

Sun/Oracle Rainbow Falls Voltage Domains



Vdd_core (VDDC) 0.8V ~ 1.1V	All Logic
Vdd_memory 0.9V ~ 1.1V	L2D Memory Cells
Vdd_analog 1.0V	PLL/SerDes
VDDT/VDDR 1.5V	I/O
VNW (VDDC +/- 0.3V)	PMOS Bias
VNWM	L2D Memory Cells PMOS Bias
VSB (VSS +/- 0.3V)	NMOS Bias

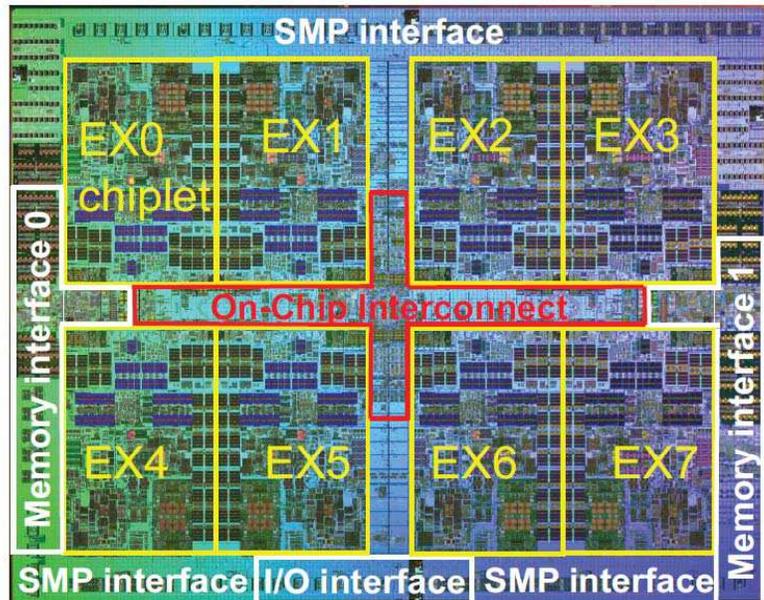
RGM2- ISCA'10

[Shin, ISSCC 2010]

29

IBM POWER7™ Processor

- 3.0 to 4.14 GHz
- 45nm CMOS SOI
- 567mm²
- 1.2B transistors
- Eight processor cores
 - 12 execution units per core
 - 4 Way SMT per core
 - 32 Threads per chip
 - 256KB L2 per core



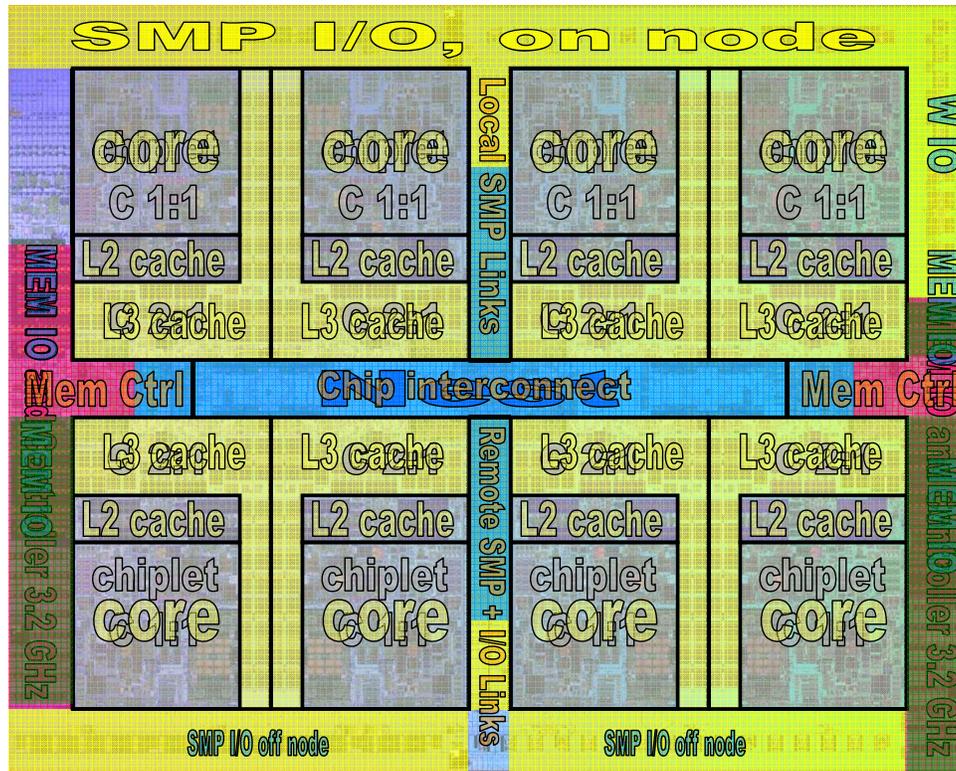
- 32MB on chip eDRAM shared L3
- Dual DDR3 Memory Controllers
 - 100GB/s Memory bandwidth per chip sustained

RGM2- ISCA'10

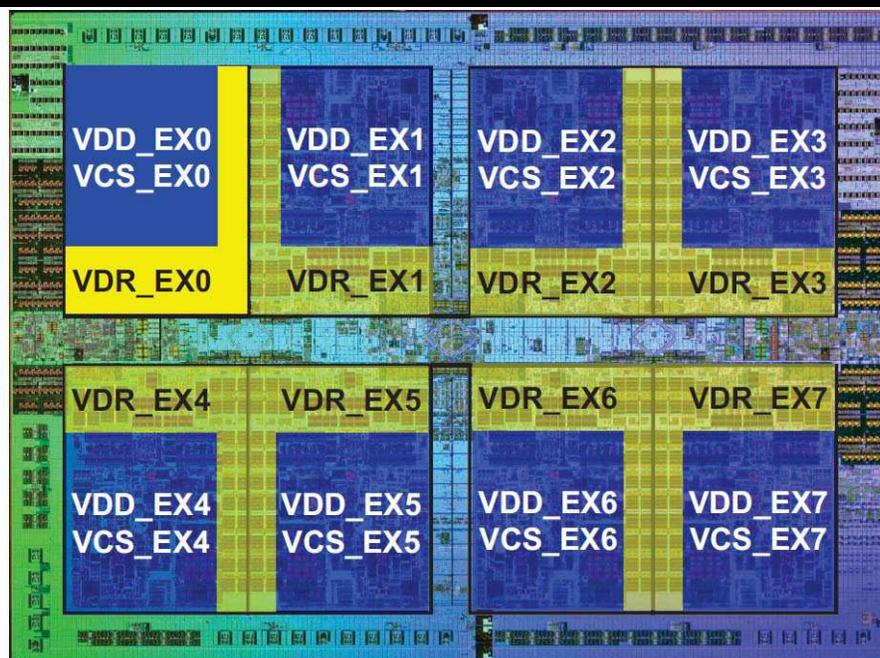
[Wendel, ISSCC 2010]

30

P7 Clock Domains

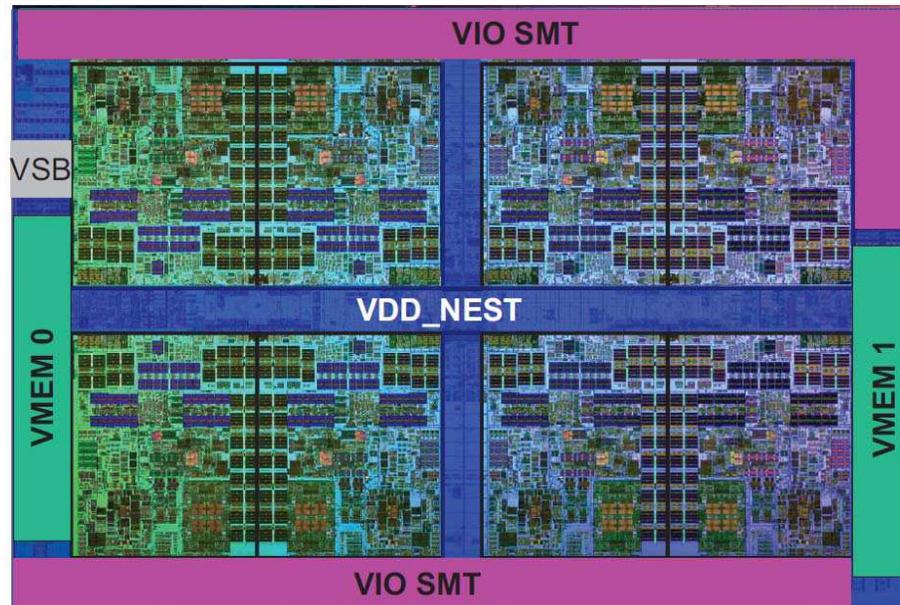


P7 Voltage Domains (1)



Name: # = 0 to 7	Meaning	Type of Voltage	Min.	Max.
VDD_EX#	Logic supply for Chiplet #	Adaptive, Dynamic	0.7V	1.3V
VCS_EX#, VDR_EX#	Array supply for chiplet#, DRAM supply for chiplet #	Adaptive, Dynamic	0.8V	1.3V

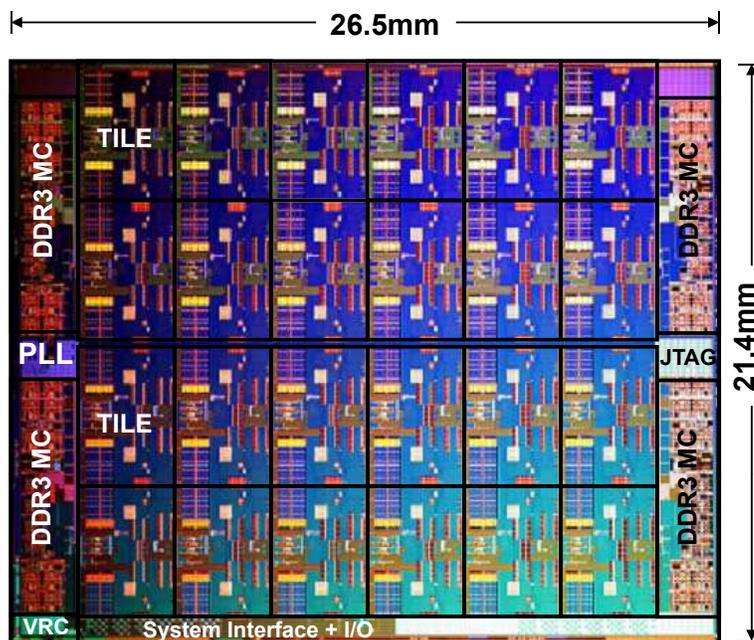
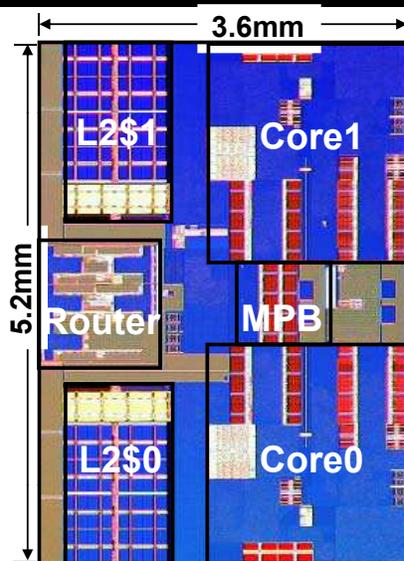
P7 Voltage Domains (2)



Name: # = 0 to 7	Meaning	Type of Voltage	Min.	Max.
VDD_NEST	Logic supply for nest & pervasive logic	Fixed, Static	1.0V	1.2V
VSB	Supply for pervasive	Fixed, Static	1.1V	1.3V
VMEM 0/1	Supply for differential IO	Fixed, Static	0.95V	1.15V
VIO SMT	Supply for IO receiver and transmitter	Fixed, Static	1.0V	1.2V

RGM2-100710

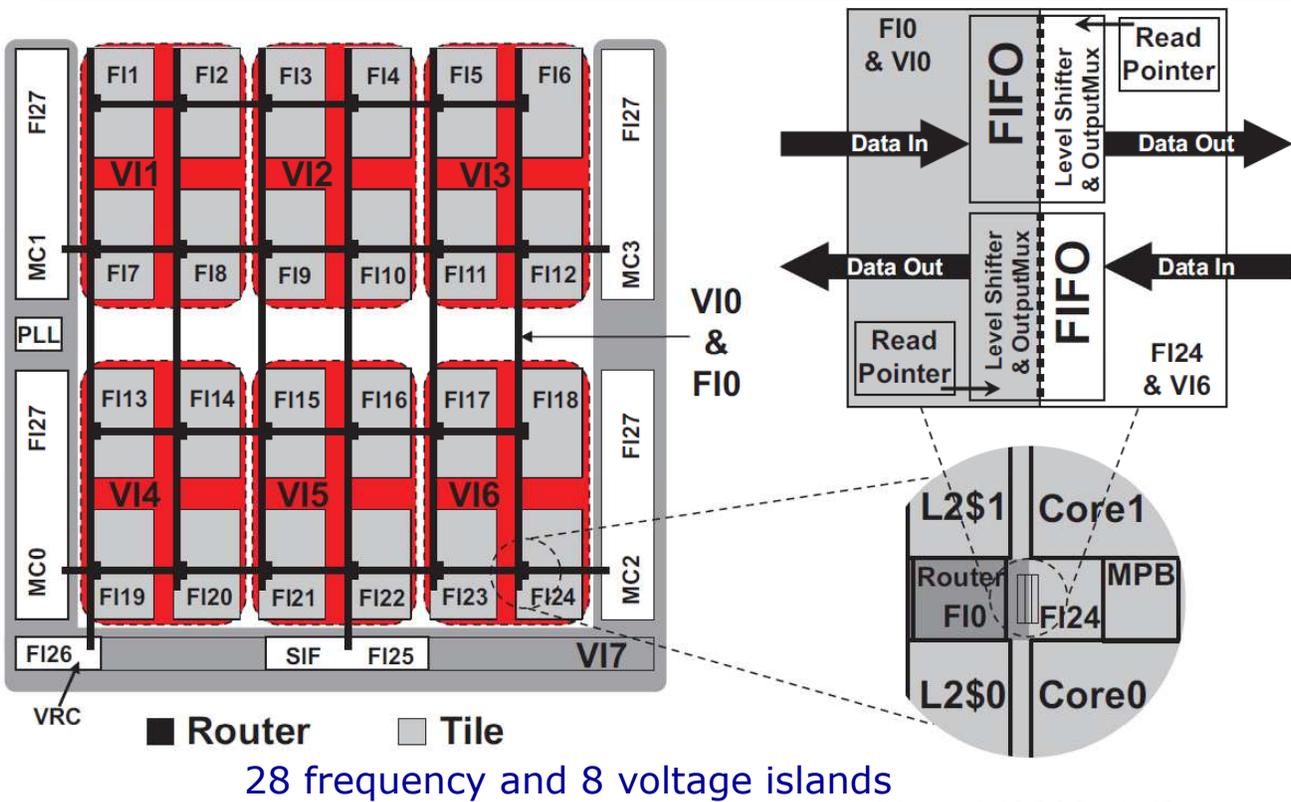
Intel 48-core SoC Processor



Technology	45nm Hi-K CMOS
Interconnect	9 Metal (Cu)
Transistors	Die: 1.3B, Tile: 48M
Tile Area	18.7mm ²
Die Area	567.1mm ²

RGM2- ISCA'10

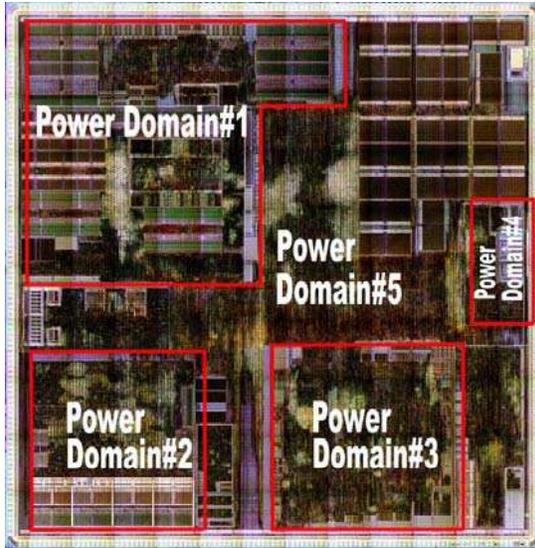
Voltage and Frequency Islands



Multi-Domain Processors Design Overview

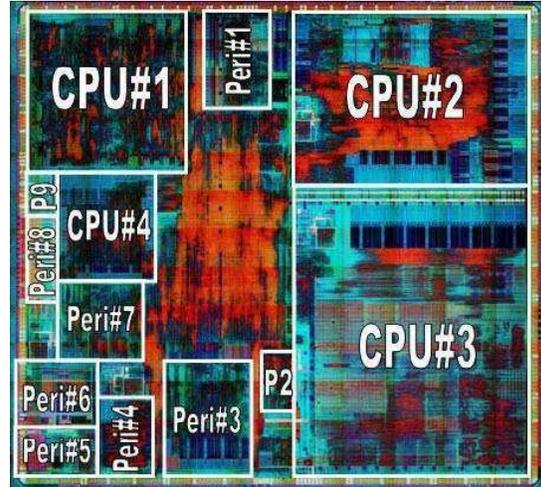
- Voltage / frequency scaling basics
- Multi-domain server processors
- Cell phone processors
- Media processors
- Dual voltage supply at the cell level
- Future directions
- Summary

TI Application Processors



- 90nm OMAP2
- 1 voltage domain
- 5 power domains

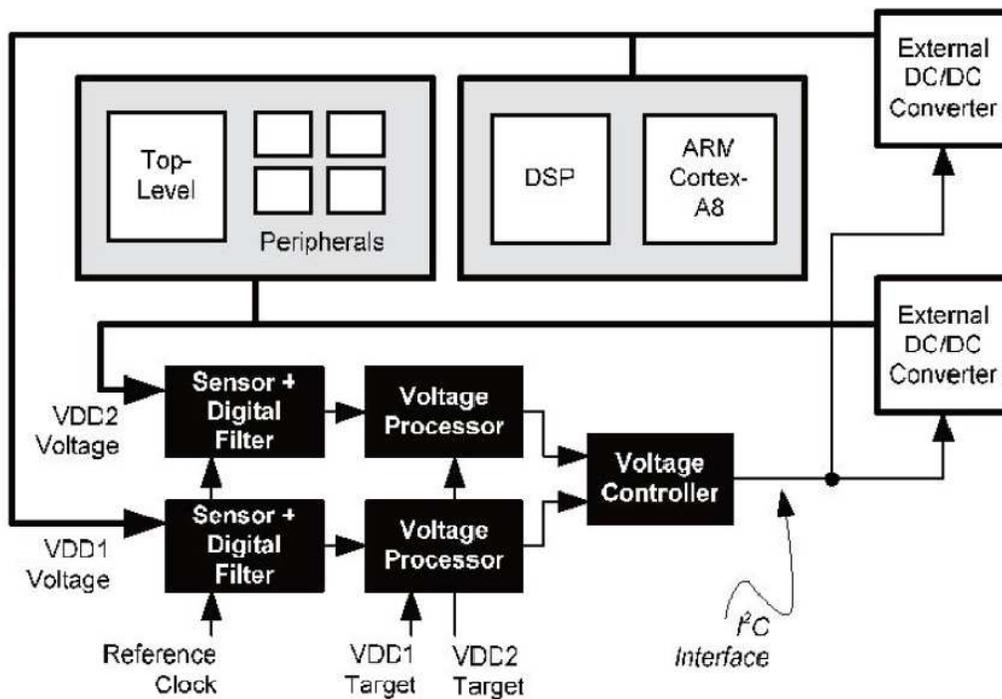
[Royannez, ISSCC 2005]



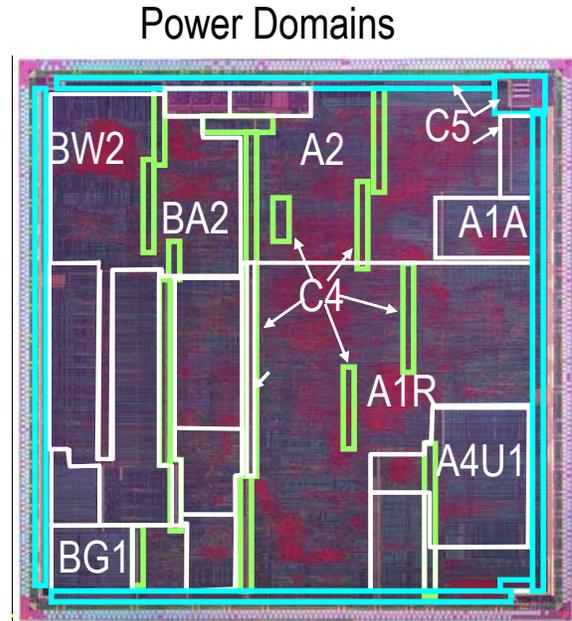
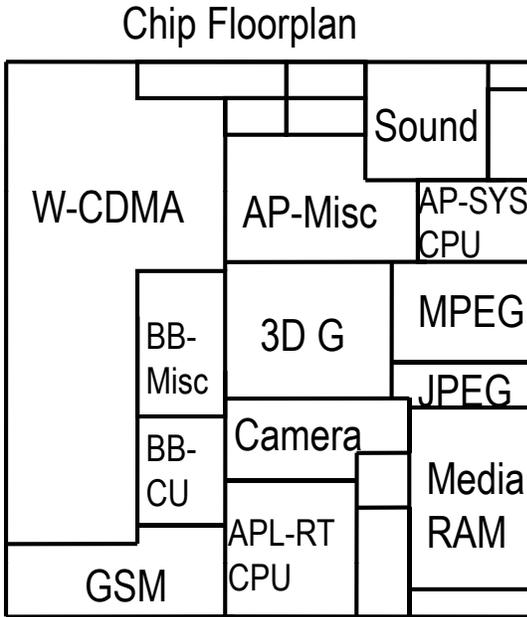
- 65nm OMAP3
- 2 voltage domains
- 11 major power domains

[Mair, VLSI Symp. 2007]

Adaptive Voltage Scaling Control Loop



Renesas 90nm Cell Phone Processor

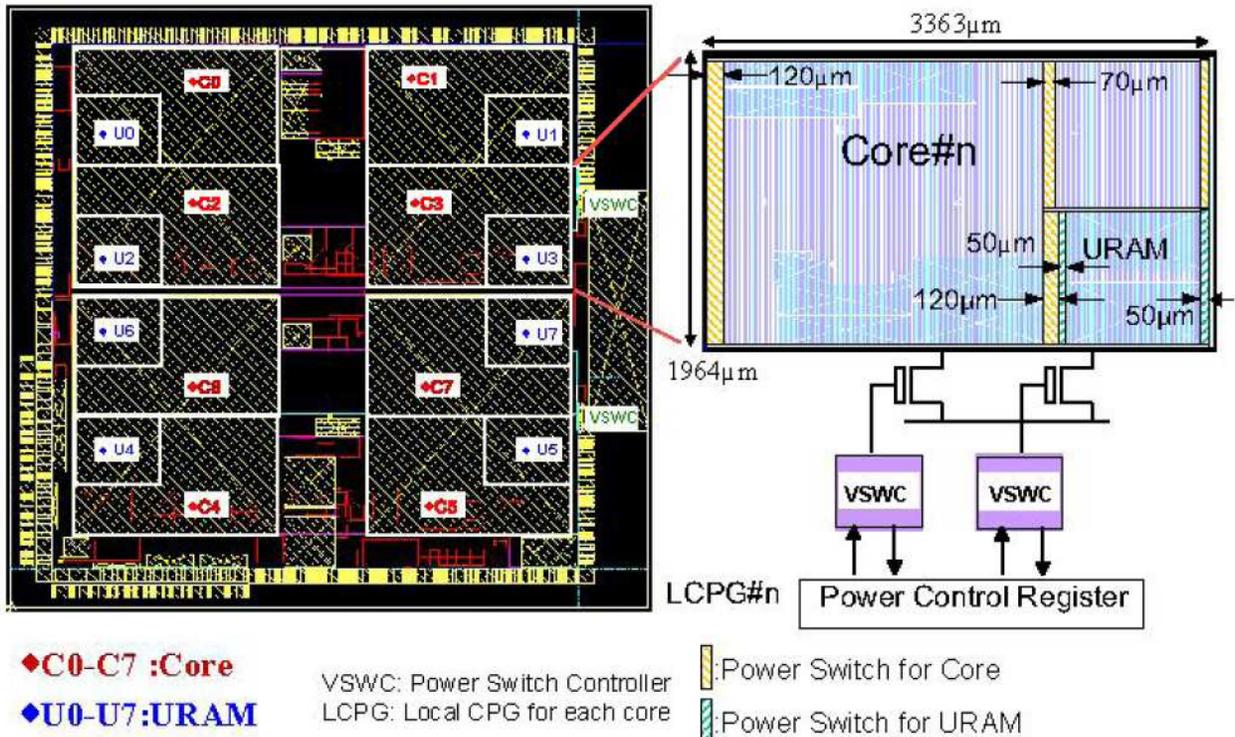


20 power domains for partial power-off,
C4 (common) domain for repeaters

Implementation Details

# of Power domains	20 domains
# of Islands for C4 (Repeaters, CK buffers, BKUP FFs)	19 islands
# of Repeaters in C4 domain	3100 cells
# of Clock buffers in C4 domain	1600 cells
# of Backup FFs in C4 domain	2300 cells
# of μ IOs (isolation cell)	20000 cells
Total area of power switch	4.2 mm ²
Power switch area ratio in the chip	3.4 %
Power-off -> power-on time (one-by-one on)	<100 μ Sec

Renesas 8-Core 90nm Processor

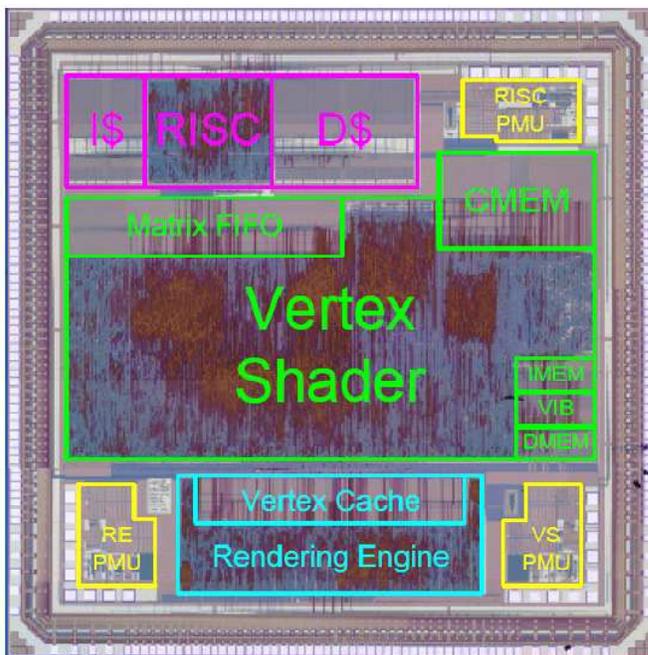


RGM2- ISCA'10

[Ito, ISSCC 2008]

41

Samsung 3D Graphics Processor



[RAMP-VI]

Chip Characteristics

Technology	0.18µm 1-P 6-M CMOS
Area	Core: 17.2mm ² Die: 25mm ²
Power supply	Core: 1.0V - 1.8V IO: 3.3V
Frequency	RISC: Max. 200MHz GP: Max. 200MHz RE: Max. 50MHz
Transistors	1.6M logic, 29KB SRAM
Processing speed	RISC: 200MIPS VS: 141Mvertices/s RE: 50Mpixels/s
Power consumption	52.4mW @ 60fps 153mW @ full speed

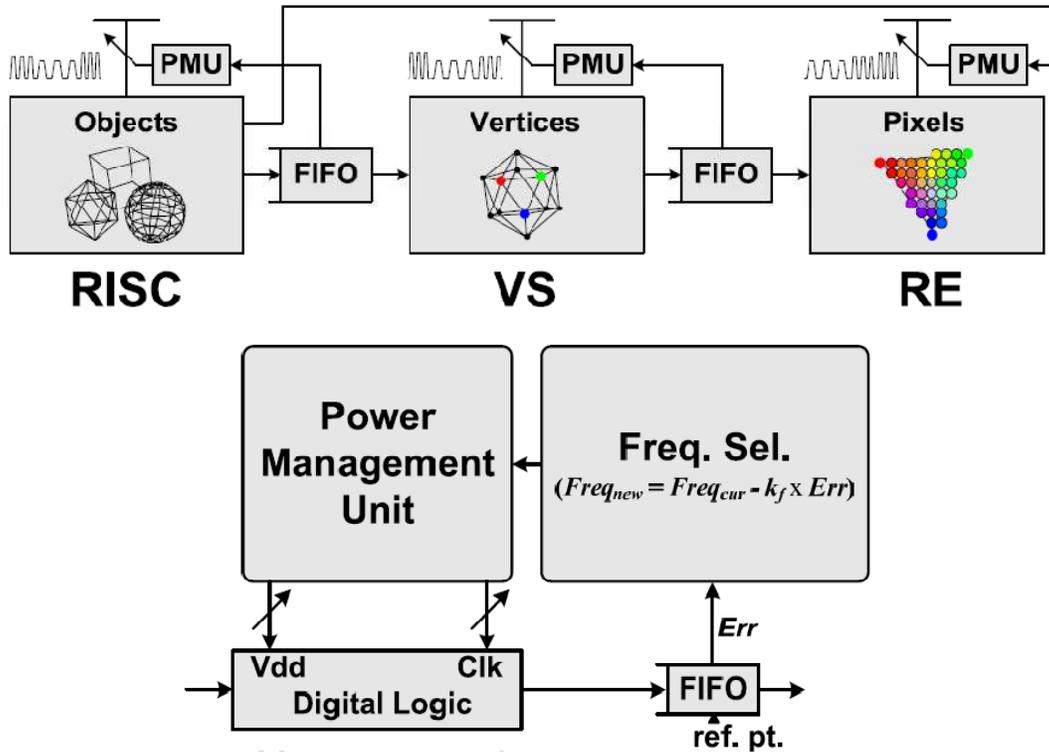
Three power domains with independent dynamic voltage-frequency scaling

RGM2- ISCA'10

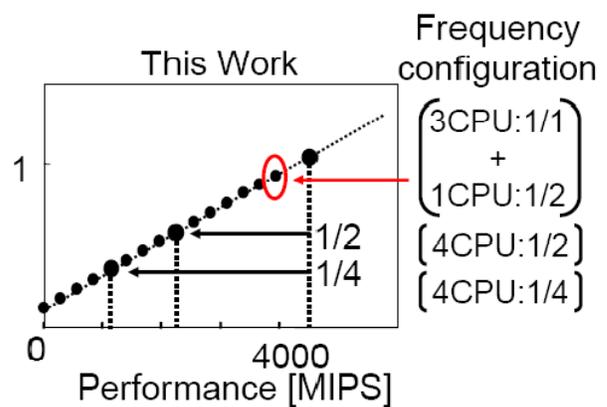
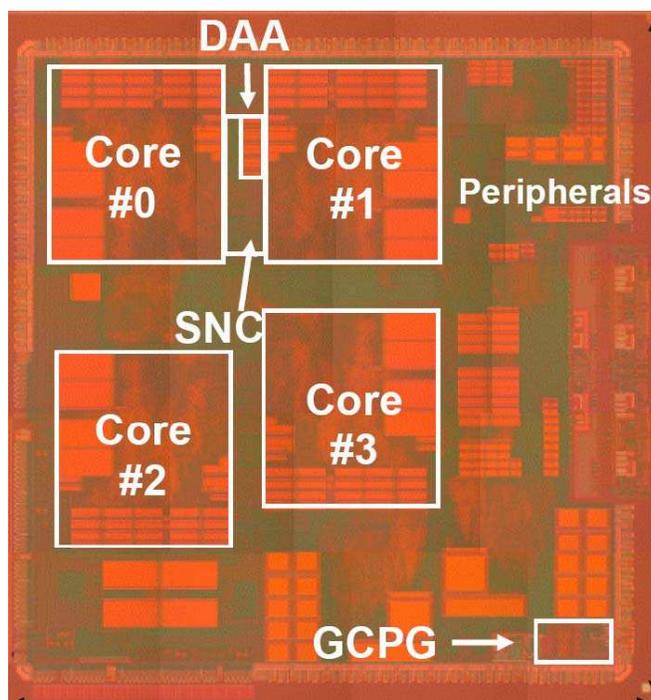
[Nam, ISSCC 2007]

42

Multiple-Domain Power Management

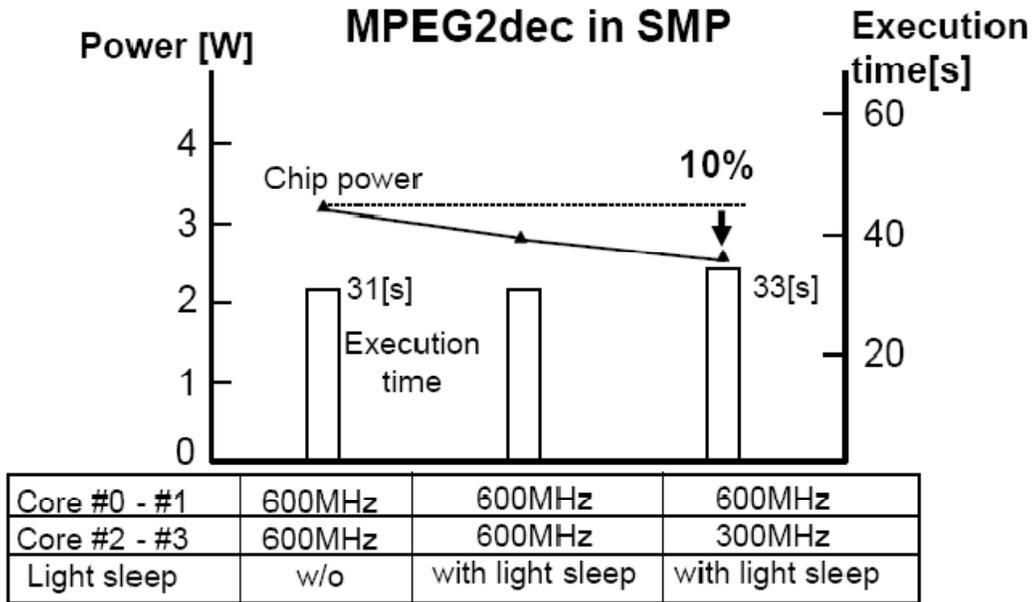


Individually Managed Core Clock Frequency



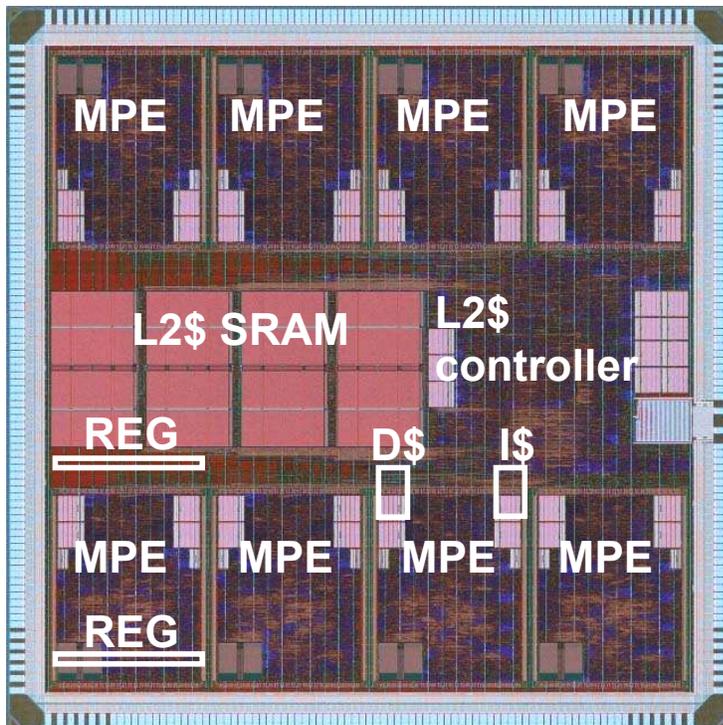
Individual core-level frequency control enables multiple performance points

Power Reduction Benefits



Individual core clock distribution has modest power reduction benefits
 Much higher benefit achievable with separate voltage domains

Toshiba's 8-Core Media Processor

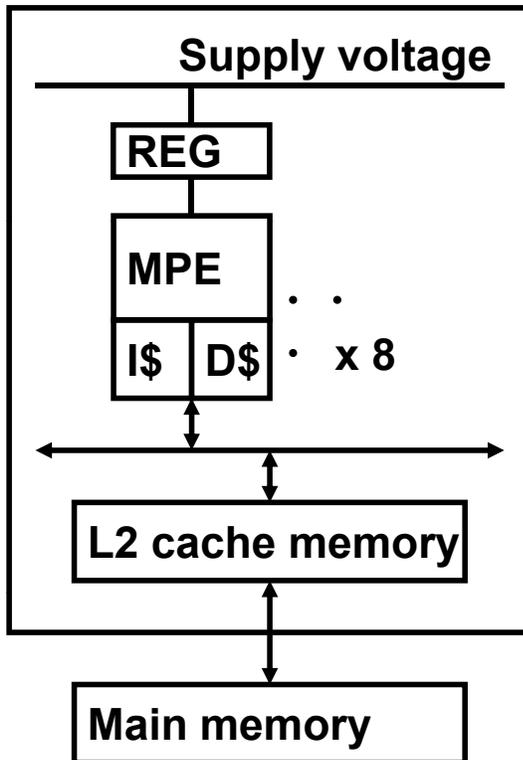


65nm CMOS
 8-layer-metal

Supply voltage
 2.5V (I/O)
 1.2V (core)
 1.2V / 0.95V / 0V
 (REG output)

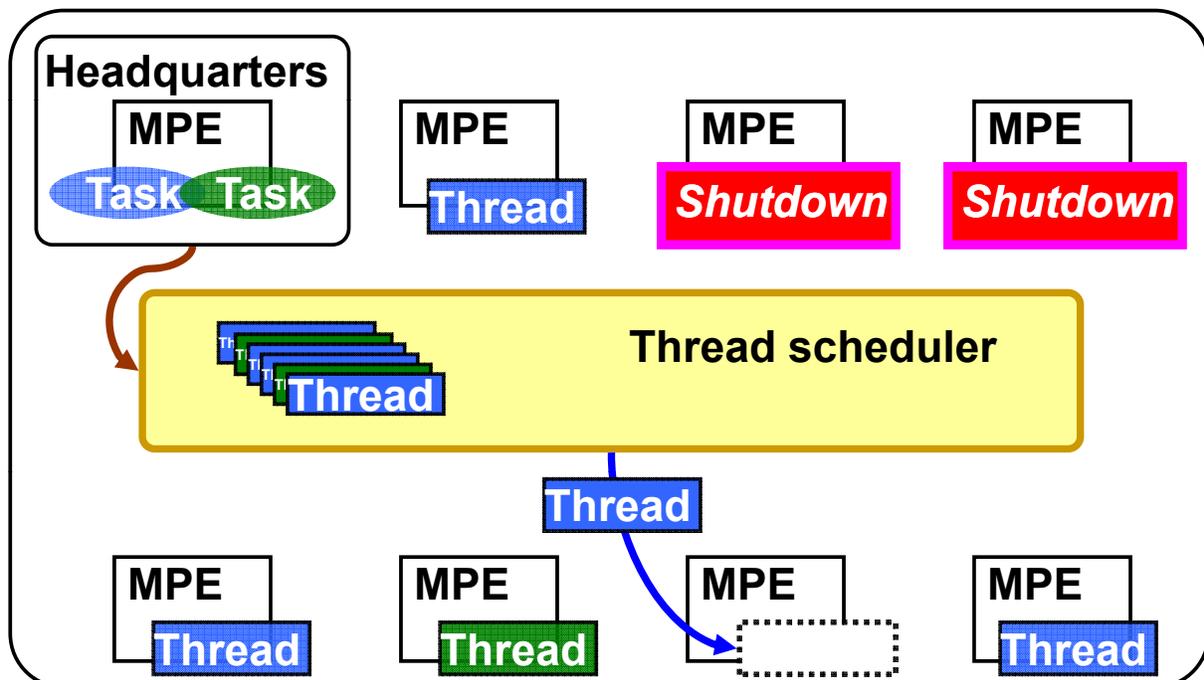
Frequency
 333MHz
 (MPE / L2\$ logic)
 166MHz
 (L2\$ SRAM / bus I/F)

Processor Architecture



- MPE (Media Processing Engine)
 - 5-stage 32b RISC with 64b SIMD 2-way VLIW co-processor
- L1 cache (8KB I\$ + 8KB D\$)
- L2 cache
- Voltage regulator (REG)
 - Control of supplied voltage
 - 1.2V / 0.95V / 0V

Software Power Management



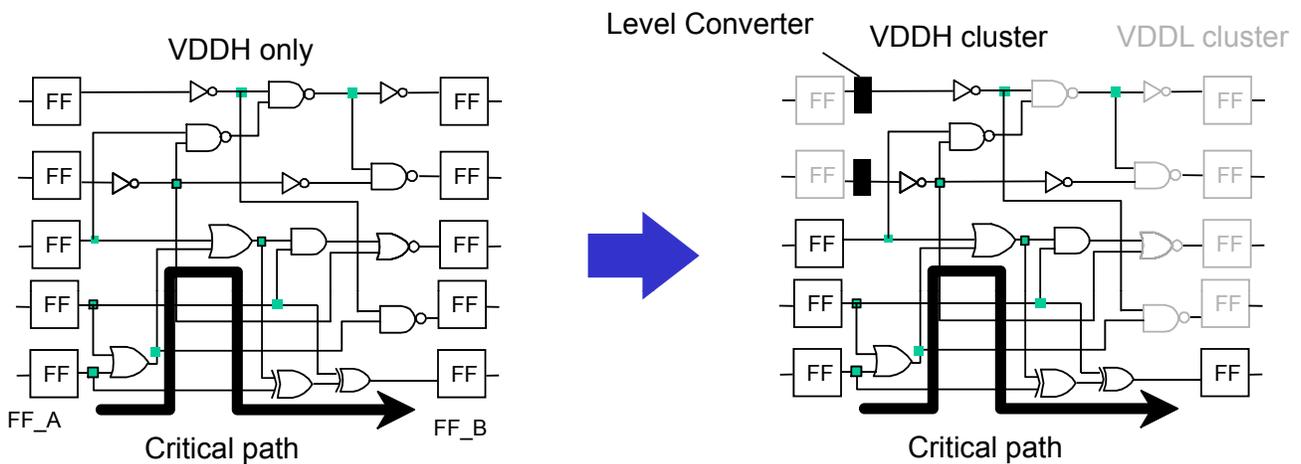
Shutdown MPEs at low work load

Multi-Domain Processors Design Overview

- Voltage / frequency scaling basics
- Multi-domain server processors
- Cell phone processors
- Media processors
- Dual voltage supply at the cell level
- Future directions
- Summary

Cell-Level Dual-VDD Approach

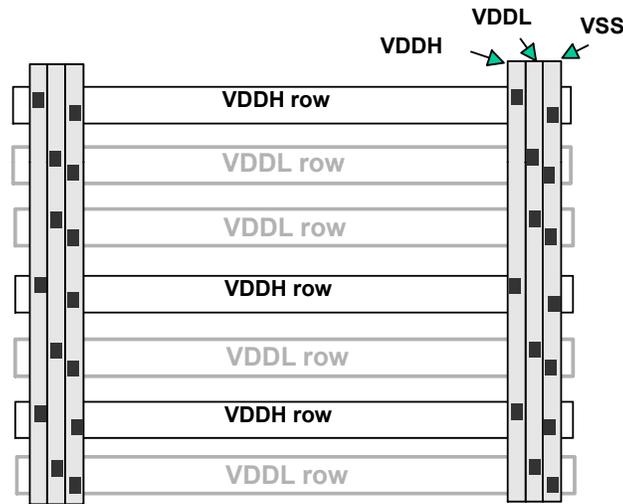
- Use reduced voltage *VDDL* in non-critical paths
- Apply original voltage *VDDH* to timing critical paths



Challenge: minimize number of level converters by clustering

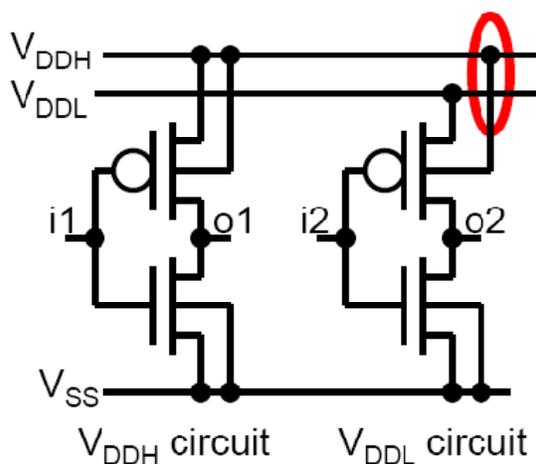
Cell-Level Dual-VDD (cont)

Row-by-Row layout architecture with Dual-V_{DD}

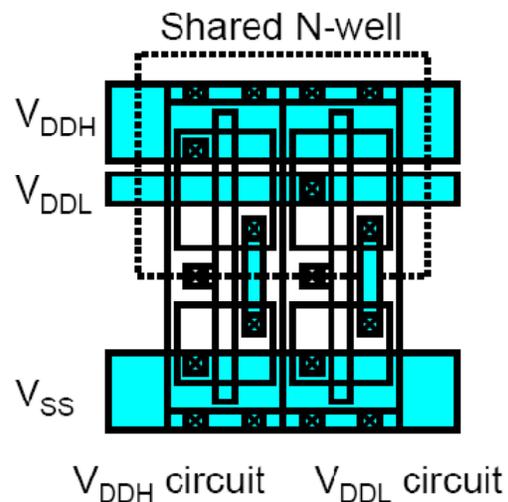


- P&R tool determines which rows should be *VDDL*
- Clock tree synthesis using *VDDL* clock buffers
- 25% power reduction on MPEG4 video codec core

Shared Well Dual Supply



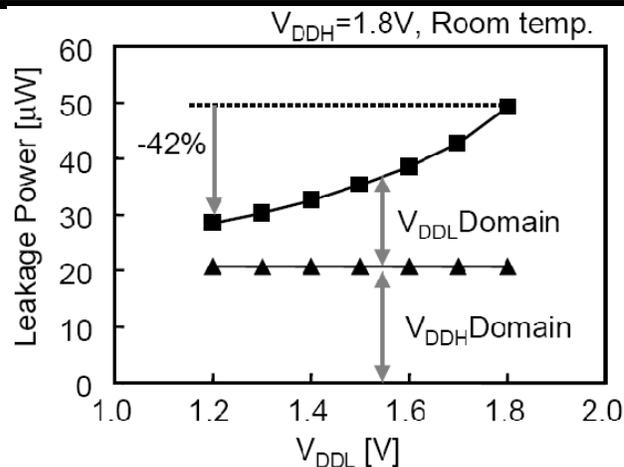
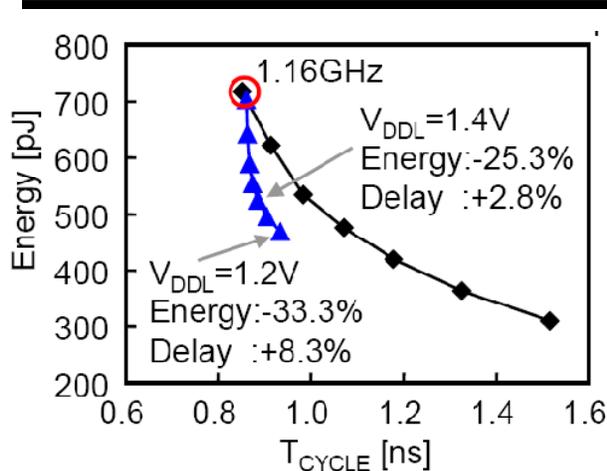
(a) circuit schematic



(b) layout

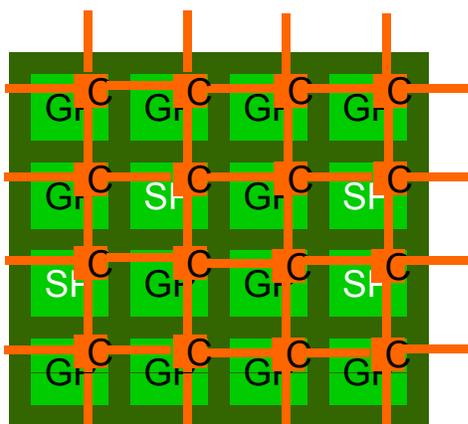
- Both circuits can be placed in the same N-well
- Cell layout becomes complex
- An intrinsic negative back-biasing of PMOS degrades speed

Dual Supply ALU Test Chip Results



- 1.16GHz 64bit ALU in GP 0.18 μ m bulk CMOS
- 25% energy saving with 2.8% delay increase
- 42% leakage reduction
- Watch for the voltage level converters overhead

Future Directions for Multi-Core Platforms



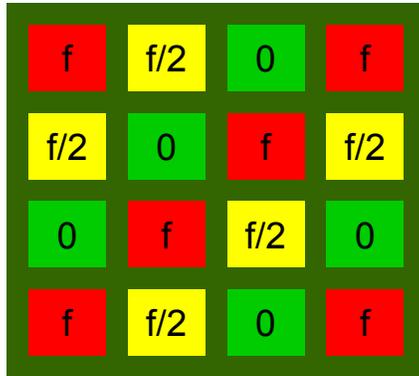
General Purpose Cores

Special Purpose HW

Interconnect fabric

Heterogeneous Multi-Core Platform—SoC

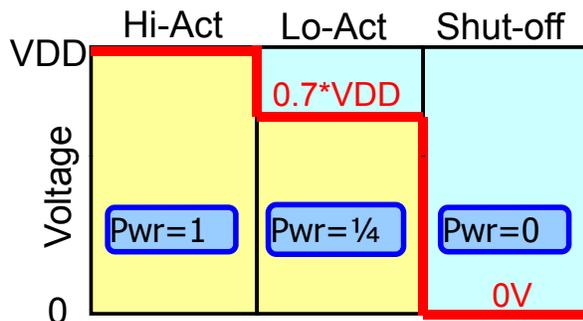
Fine Grain Power Management



Cores with critical tasks
Freq = f , at V_{dd}
TPT = 1, Power = 1

Non-critical cores
Freq = $f/2$, at $0.7 \times V_{dd}$
TPT = 0.5, Power = 0.25

Cores shut down
TPT = 0, Power = 0



RGM2- ISCA'10

[Adapted from S. Borkar, DAC 2007]

55

Summary

- Multiple voltage and clock domains are widely used in modern processor design to manage power and process scaling issues
- Optimize voltage/frequency operating point for each block to minimize power consumption
- The need to shut-off unused logic is driving a finer granularity clock and power gating
- Core and cache recovery enables multiple product options where disabled cores and cache slices are clock and power gated
- Increased use of Globally Asynchronous Locally Synchronous (GALS) clocking for large SoC designs
- Managing all these voltage and frequency domains requires increased software complexity

RGM2- ISCA'10

56

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

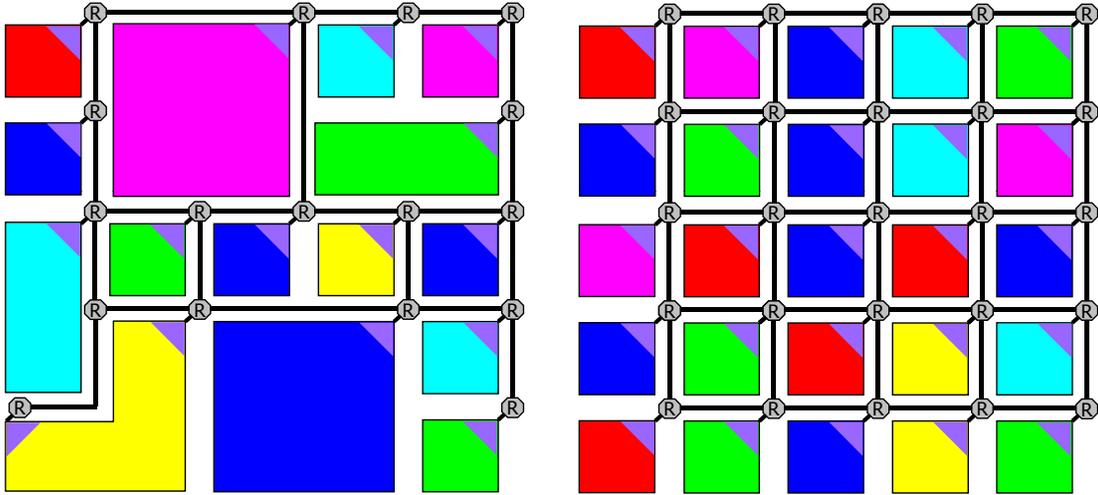
The Asynchronous NOC

Ran Ginosar

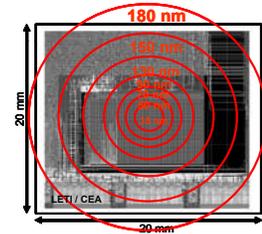
Technion

ran@ee.technion.ac.il

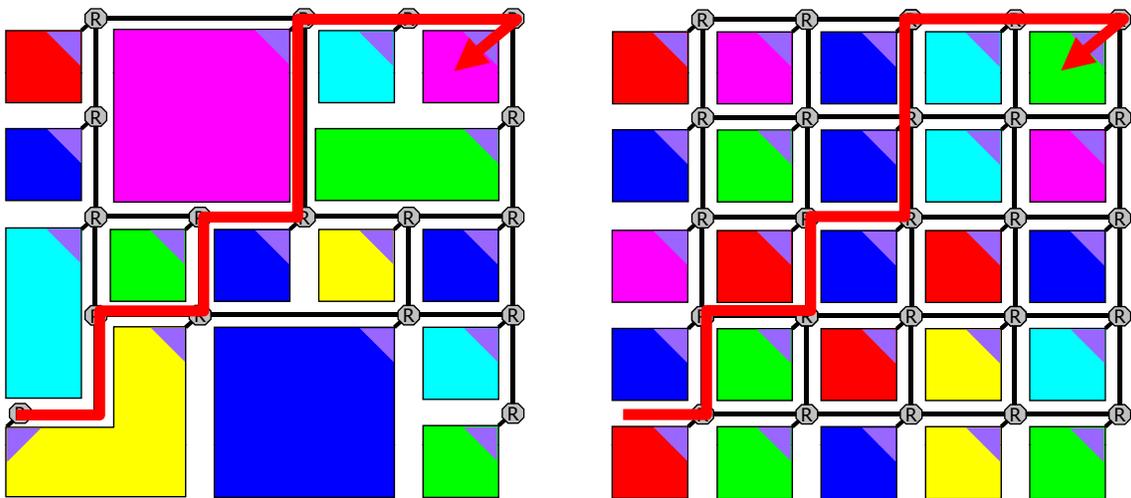
SoC and CMP



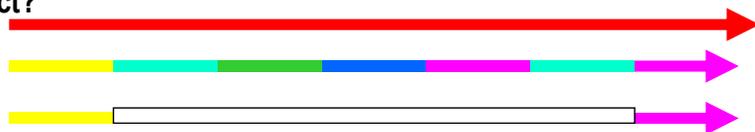
- Each block is a separate clock (+voltage) domain
 - ▼ Fast clocks cannot make it over long distances
 - ▼ They are hard to design
 - ▼ They consume too much power



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

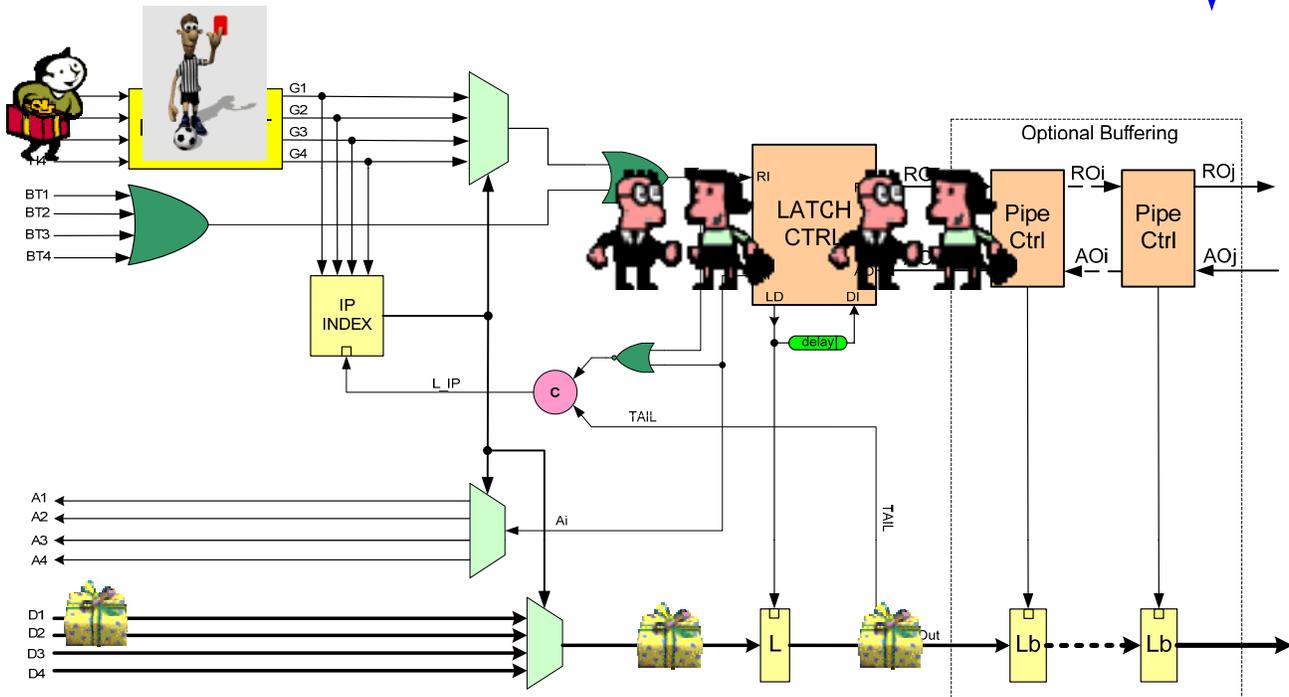
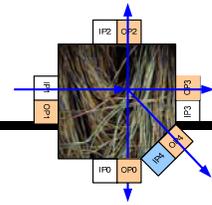
- ▼ Long data blocks – high bandwidth, long latency, low power, statistical delivery (“parcel post”)
- ▼ Fast quick short msgs – low latency guarantee, maybe high power, rare (“Fedex”)
- ▼ Others ...

■ 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

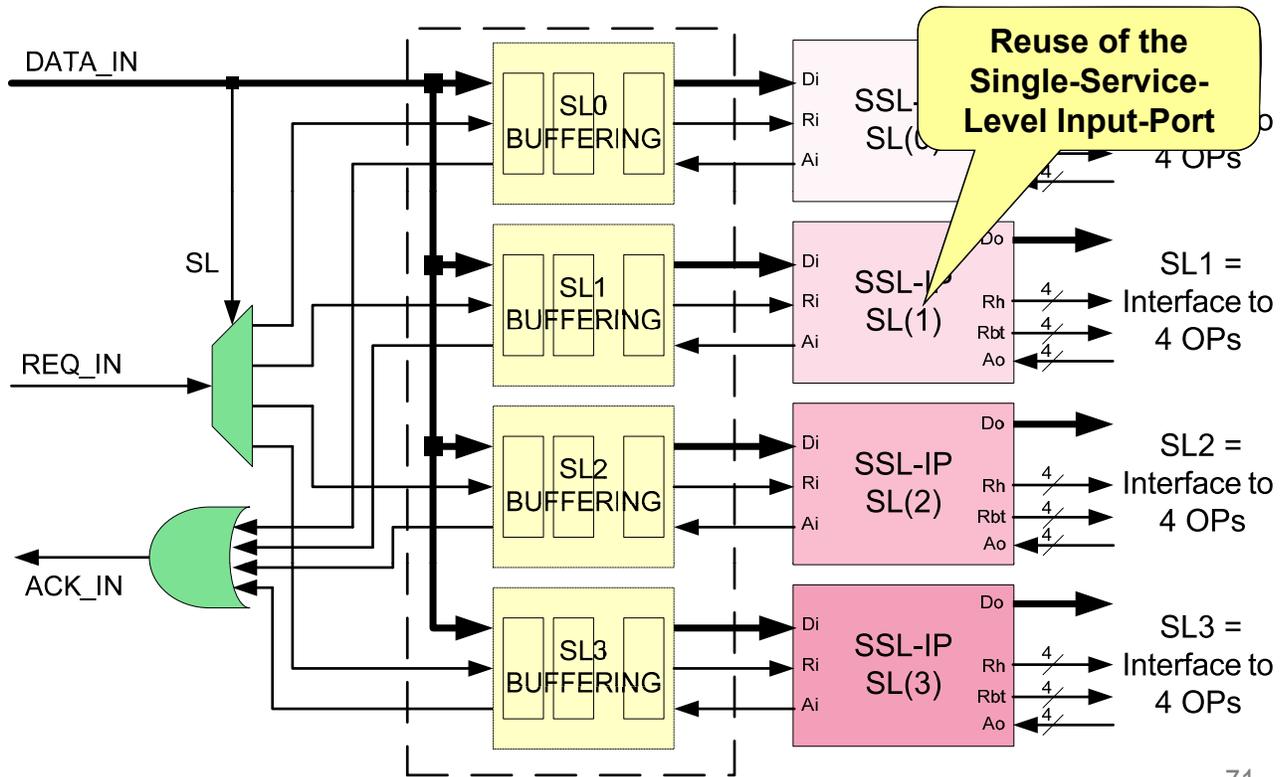
Async Single-Service-Level Output-Port



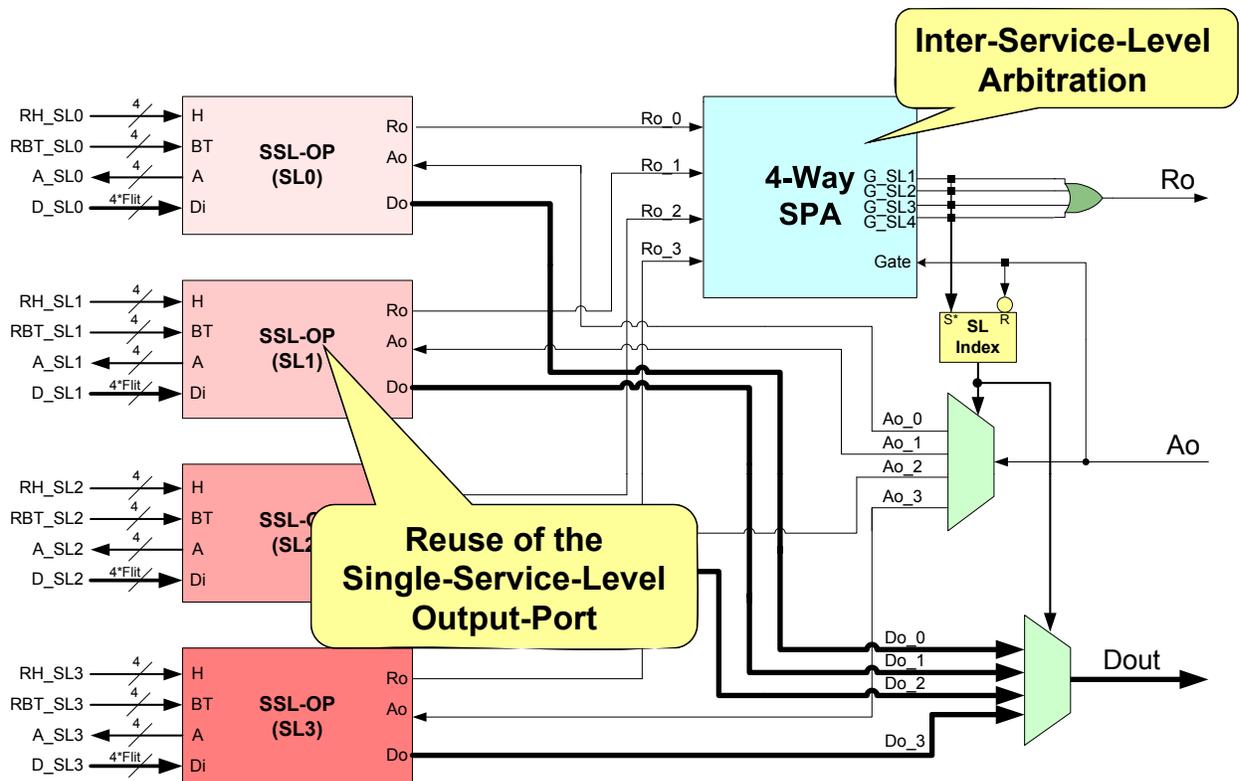
Adding multiple service levels for QoS

- 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

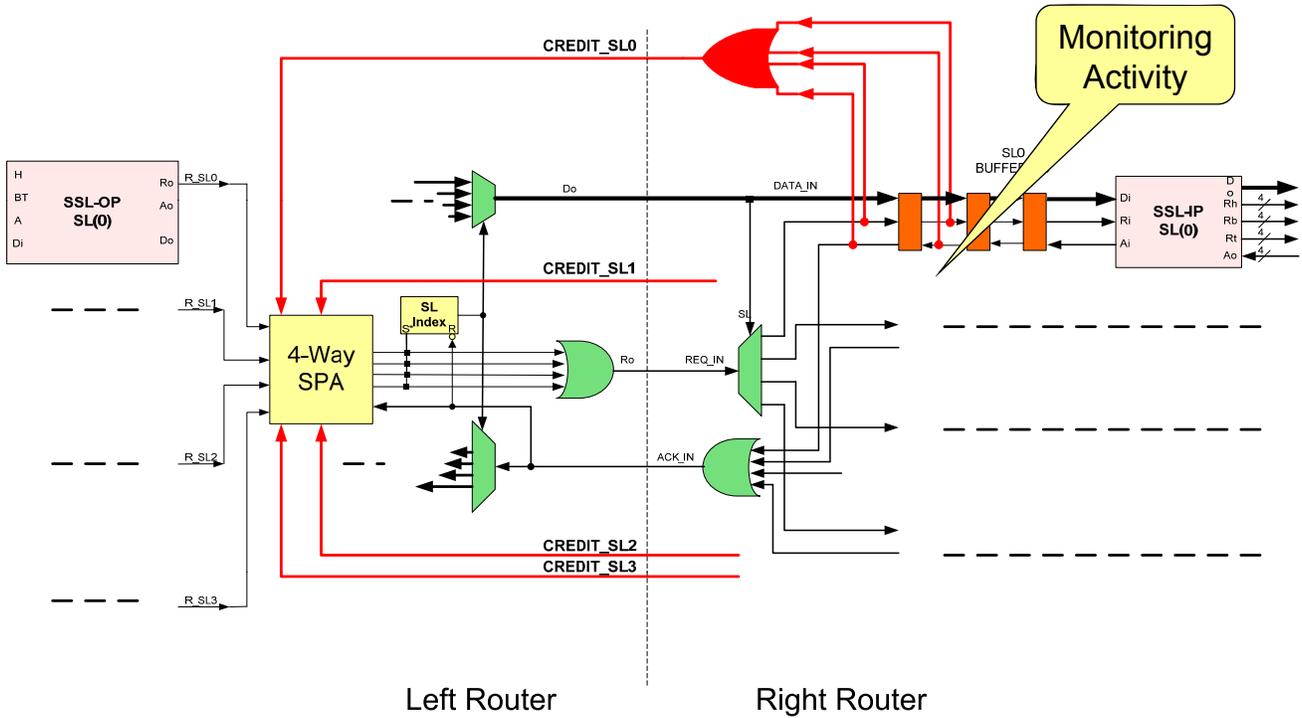
Multi-Service-Level Input-Port



Multi-Service-Level Output-Port



Buffering and Credits



Performance

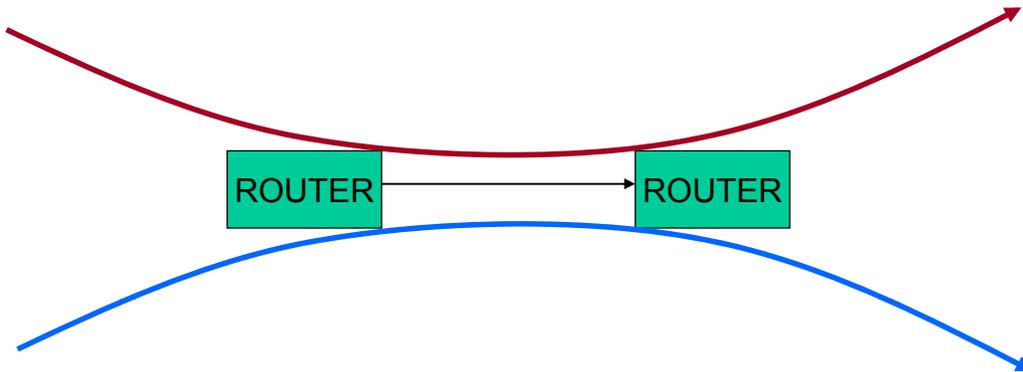
	SYNC router	ASYNCRouter	
Cell Area	960,000	470,000	µm ²
Equivalent Gates (2-in NAND)	17,500	8,500	Gates
Number of transistors	70,000	34,200	
Number of FFs+Latches	880	620	
Min Latency (Input to Output)	3.7 (1)	13.0/9.2 *	ns (CLK)
Data Cycle	14.8 (4)	13.3	ns (CLK)
Max Data Rate	67.6	75.2	Mflits/s
Max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

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

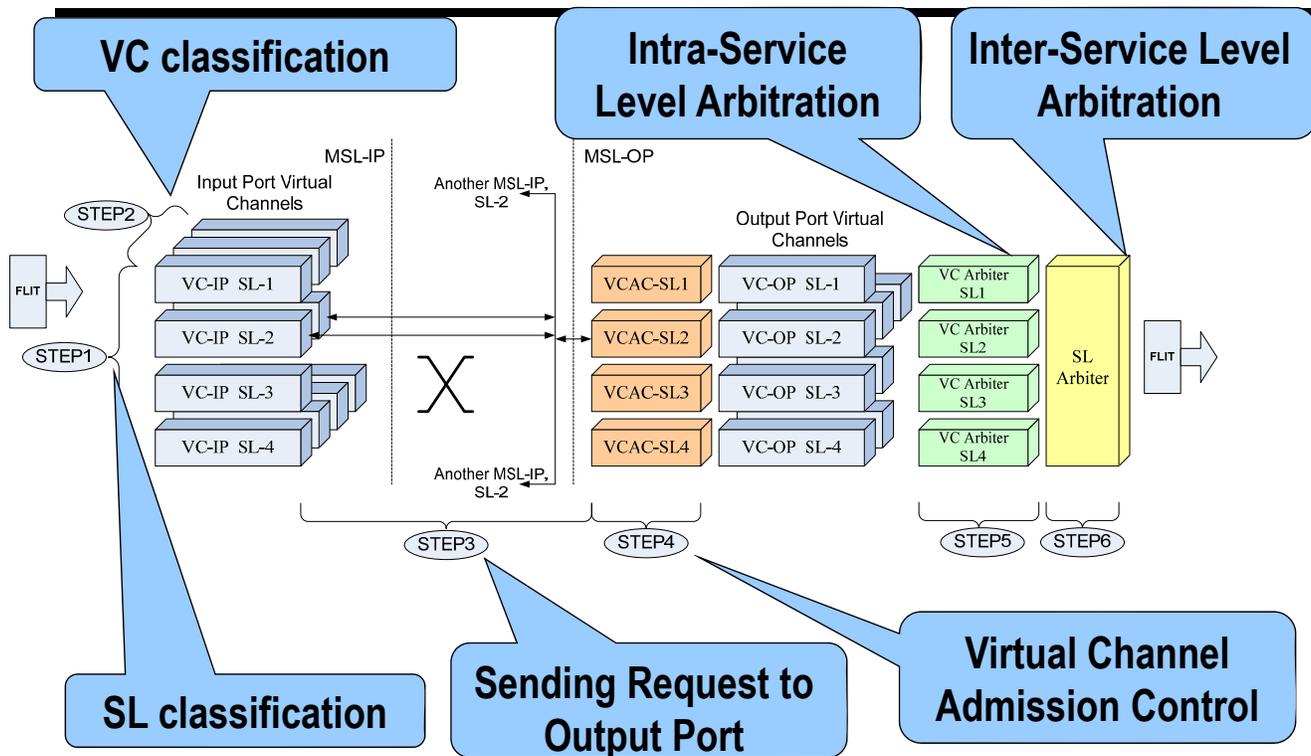
** Synchronous router has a critical path of ~20 FO4 gate delays, matching or outperforming other published results.

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

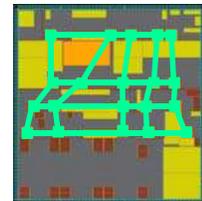
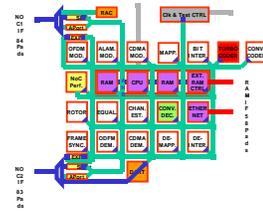


FAUST + ALPIN + MAGALI: A proof-of-concept in silicon

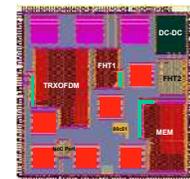
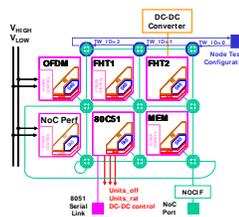
CEA-LETI
Grenoble, France

LETI's Research: 3x MPSoC with ANOC

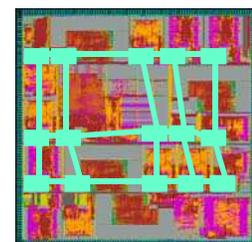
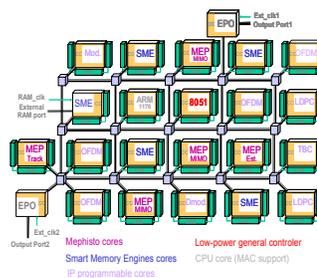
- FAUST (2005)
ST 130 nm – 20 nodes
ANOC



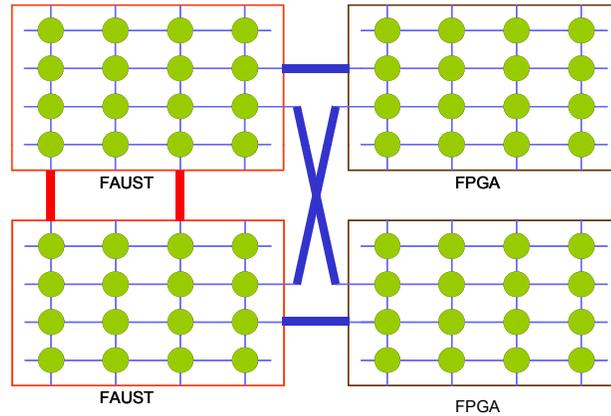
- ALPIN (2007)
ST 65 nm – 9 nodes
ANOC for low-power



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

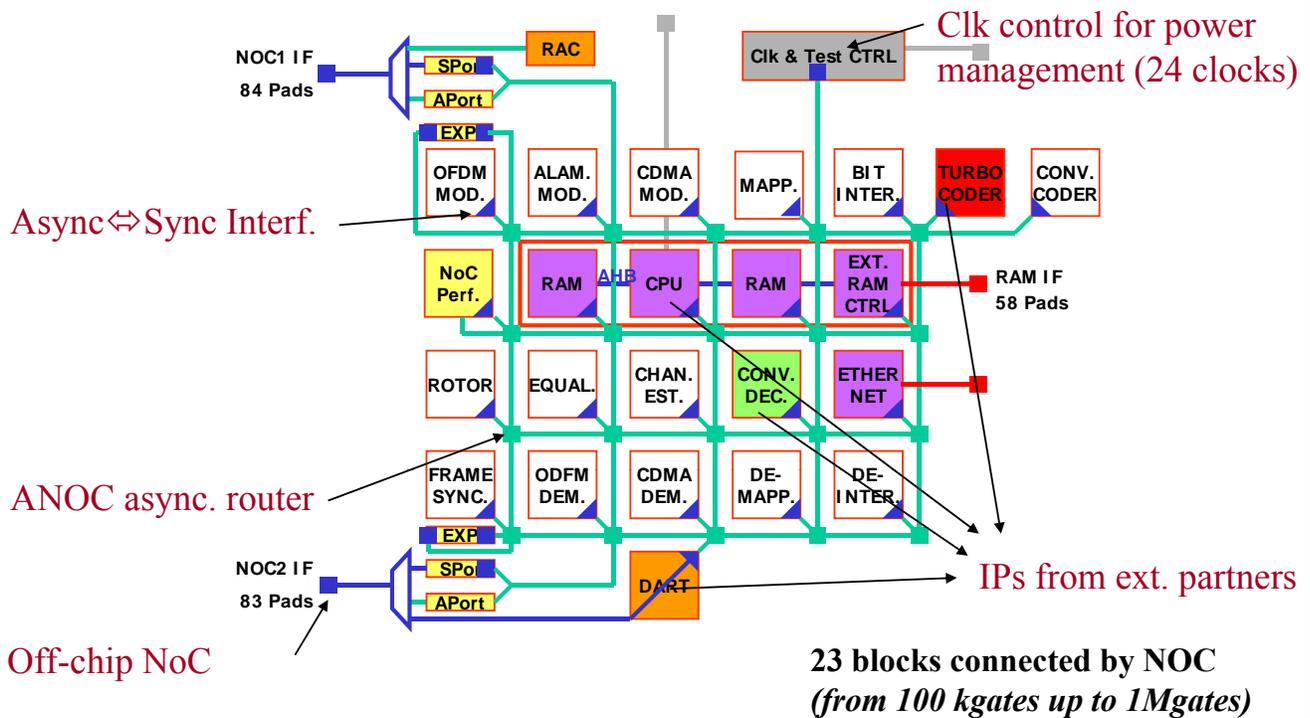


The FAUST environment: NOC off-chip, Async and Sync



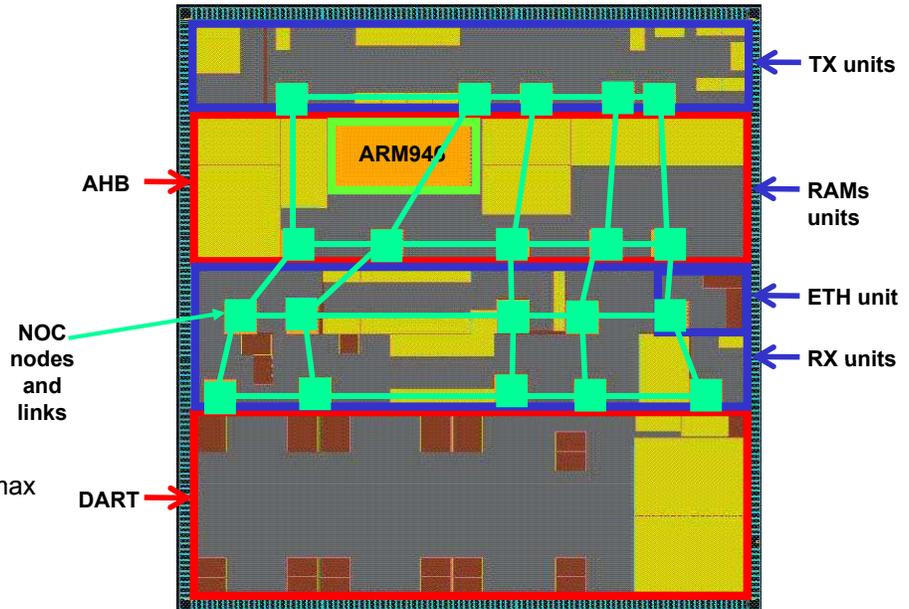
— SYNCHRONOUS OFF-CHIP LINK
 — ASYNCHRONOUS OFF-CHIP LINK

The FAUST architecture

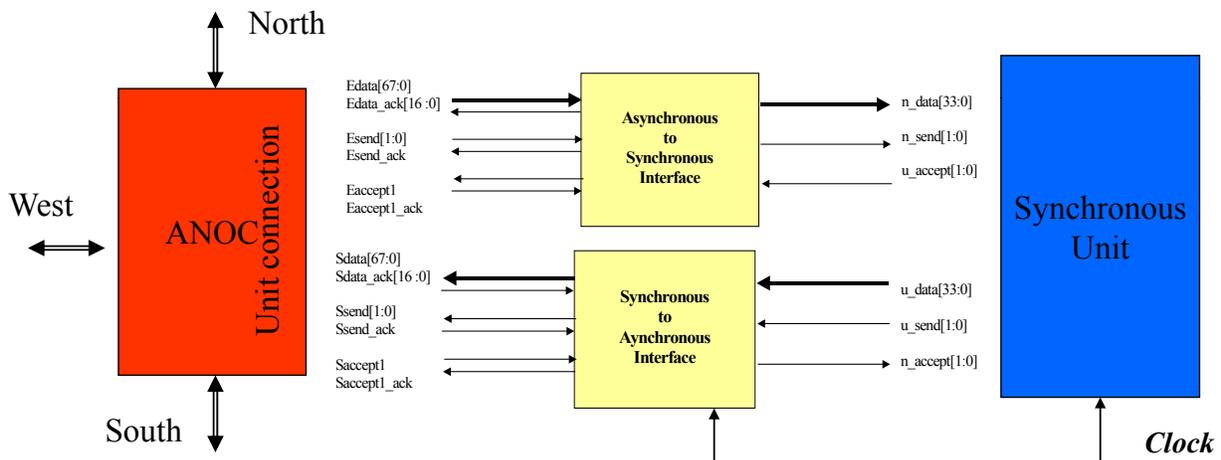


FAUST chip description & Floor-plan

- Tech.: STM/HCMOS9GPLL (0.13 μ)
- 23 NoC units (166 MHz)
- RAM blocks : 81
- CPU: ARM946
- 4.5 Mgate
- 275 I/Os
- Core area = 70 mm²
- Chip area = 80 mm²
- Package : TBGA 420
- Core power supply: 1.2 V
- I/O power supply: 3.3 V
- Power Consumption : 3 Watts max



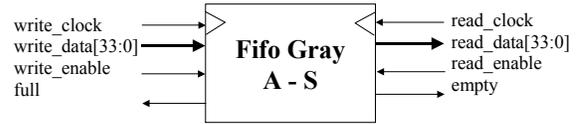
FAUST GALS interface



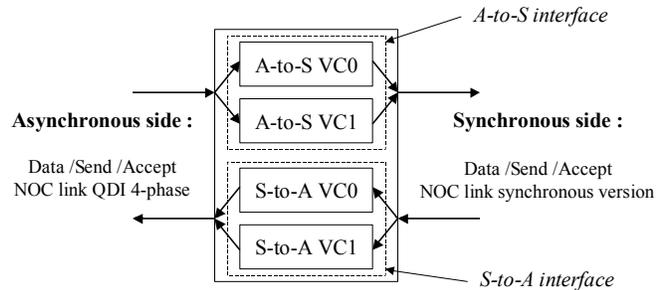
- The GALS interface: A synchronizer
- Uses dual-clock fifo (2 stages)
 - ▼ Allows sync & async transfers every clock cycles

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

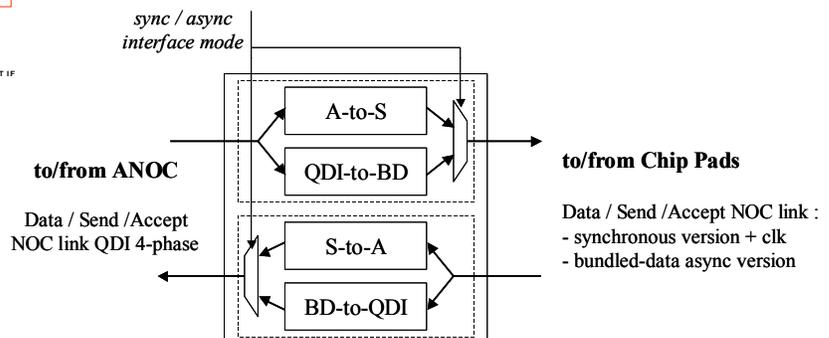
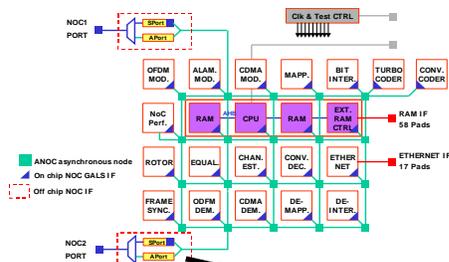


- NOC interface
 - ▼ A-to-S + S-to-A fifo's
 - ▼ One fifo per Virtual Channel



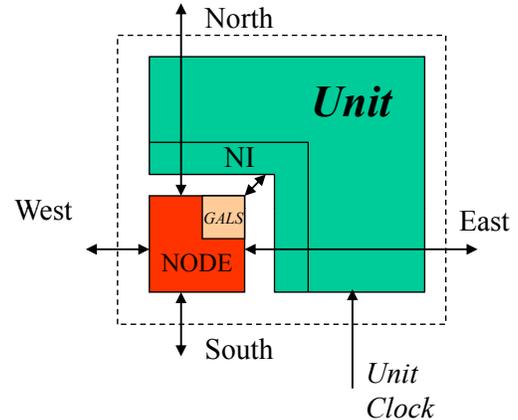
FAUST external interfaces

- 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



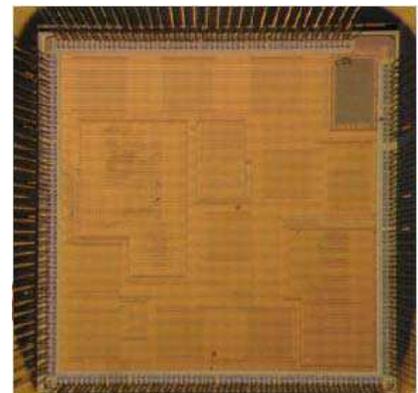
FAUST Performance Results

- 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

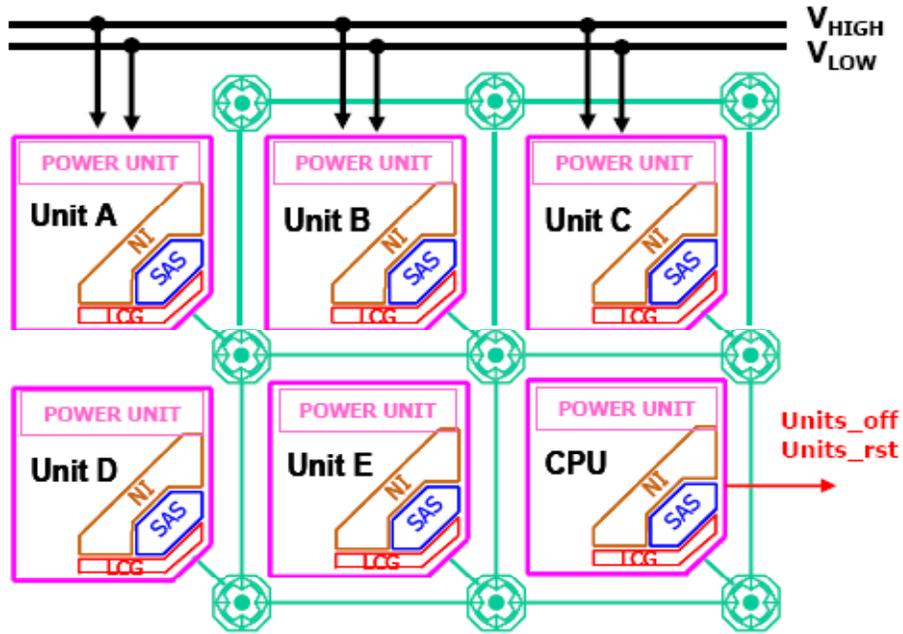


ALPIN

- 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



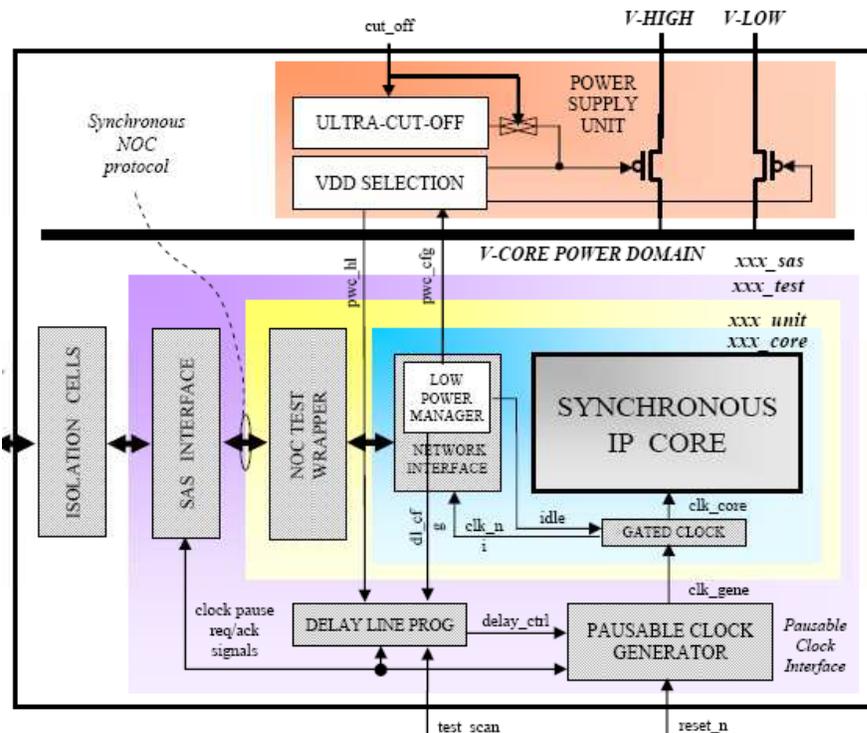
ALPIN



SAS=Sync/Async/Sync synchronizer

LCG=Local Clock Gating

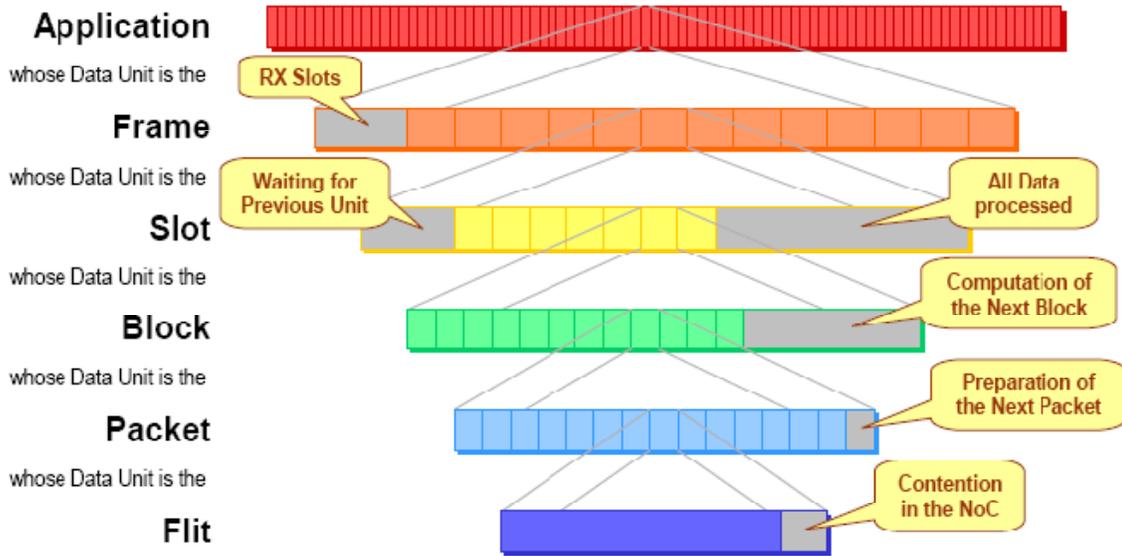
ALPIN Unit: Sync core, Async DVFS Wrapper



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

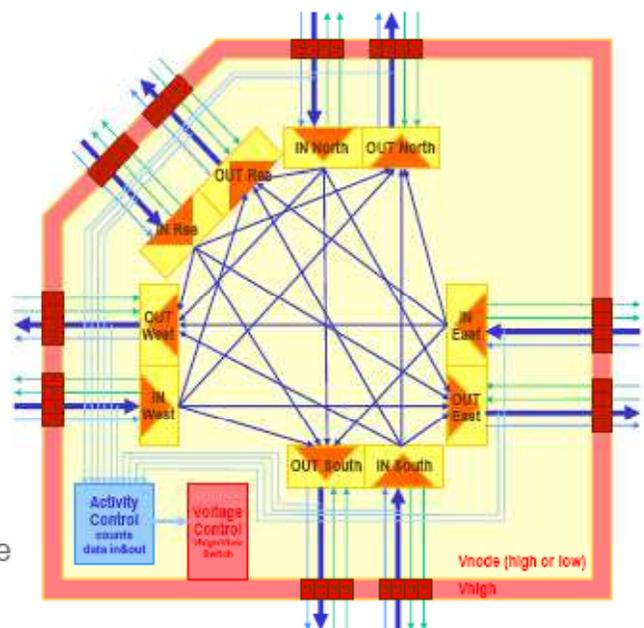
ALPIN auto-off router architecture

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

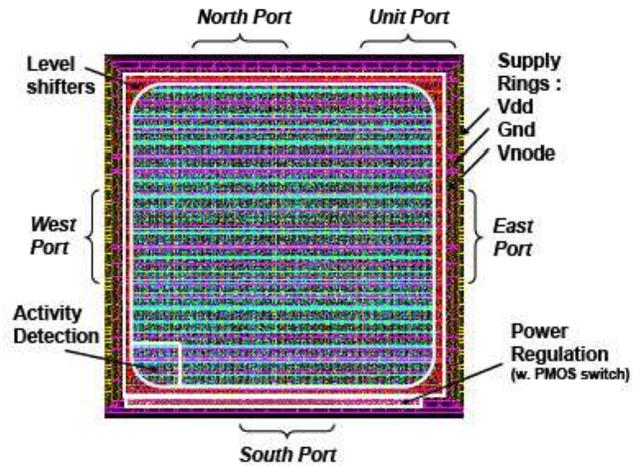
- Power switch
 - ♦ To regulate core voltage according to activity
- Level shifters on all I/Os
 - ♦ To maintain nominal voltage on network links



ALPIN auto-off router layout

■ Area:

- ▼ Typical unit 0.2 mm²
- ▼ Core: 85%
- ▼ Level shifters: 13%
- ▼ Activity detection: 2%
- ▼ Power switch: 0%



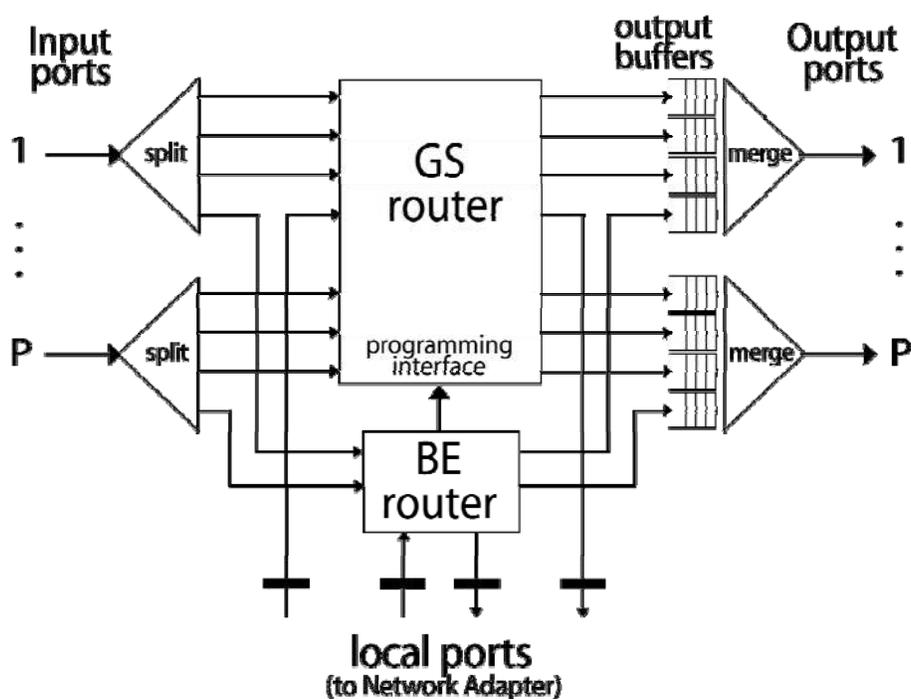
MANGO

Technical University Denmark

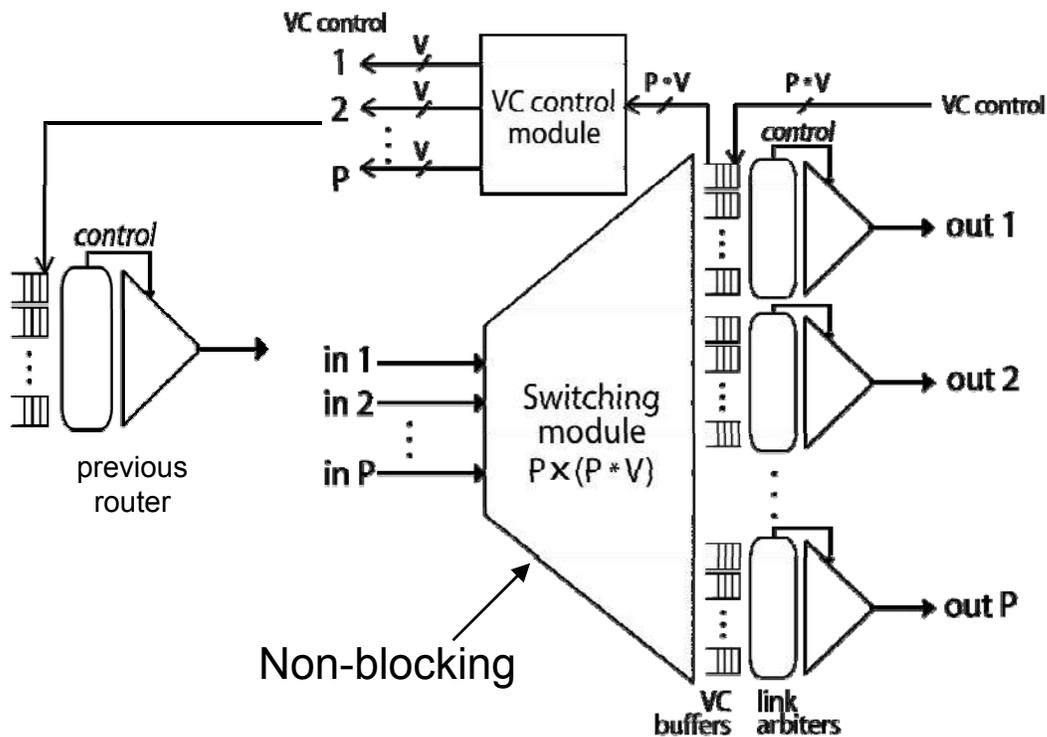
MANGO

- **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 mm² (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

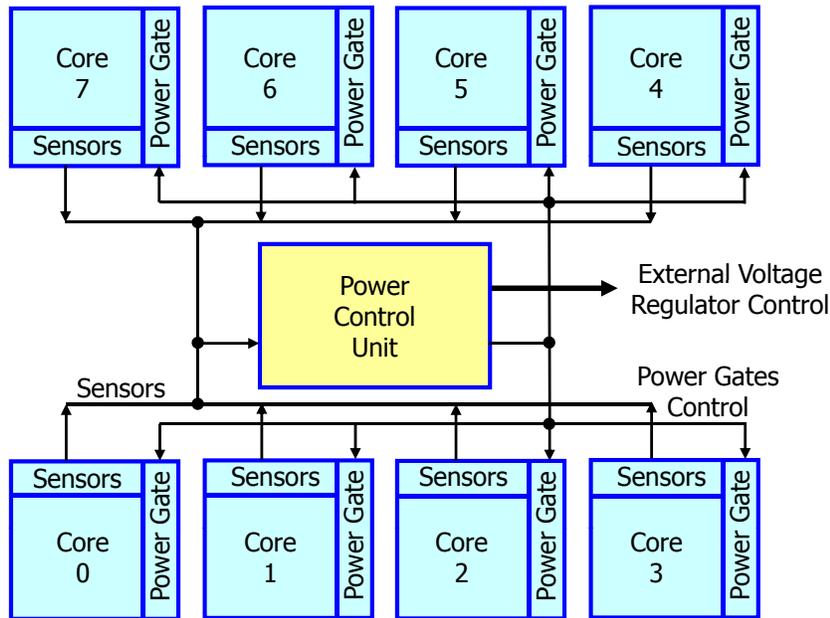
- **Part I: Multi-Domain Processors Design Overview (2:00-2:45PM)**
 - ▼ Multi-domain server, cell phone, and media processors
 - ▼ Power management techniques
- **Part II: Router Design and Synchronization Issues (2:45-3:30PM)**
 - ▼ Asynchronous router design
 - ▼ Quality of Service and virtual channels 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

Power Management Unit



RGM2- ISCA'10

[Rusu, A-SSCC 2009] 101

Outline

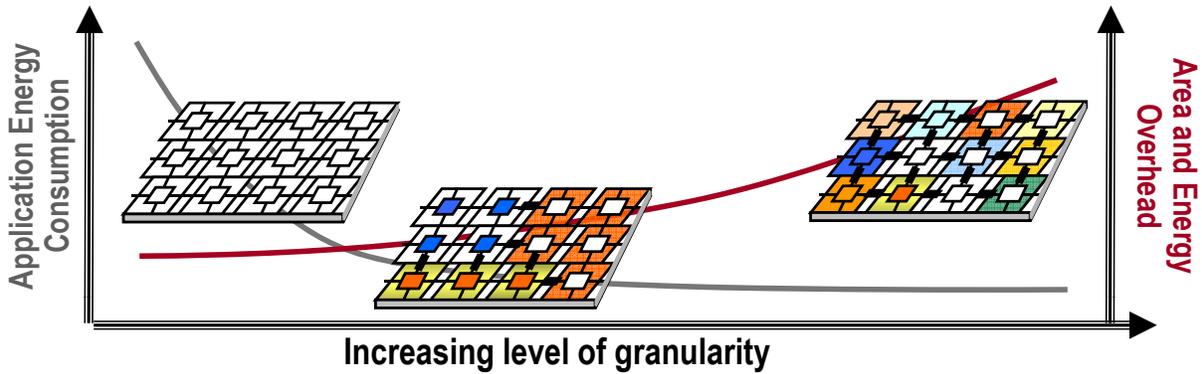
- VFI partitioning
 - ▼ Multi-VFI NoC designs
 - ▼ Partitioning and voltage assignment
 - ▼ Examples
- On-line control
 - ▼ State-based model construction
 - ▼ Feedback control architecture
 - ▼ Stability issues
- Summary

RGM2- ISCA'10

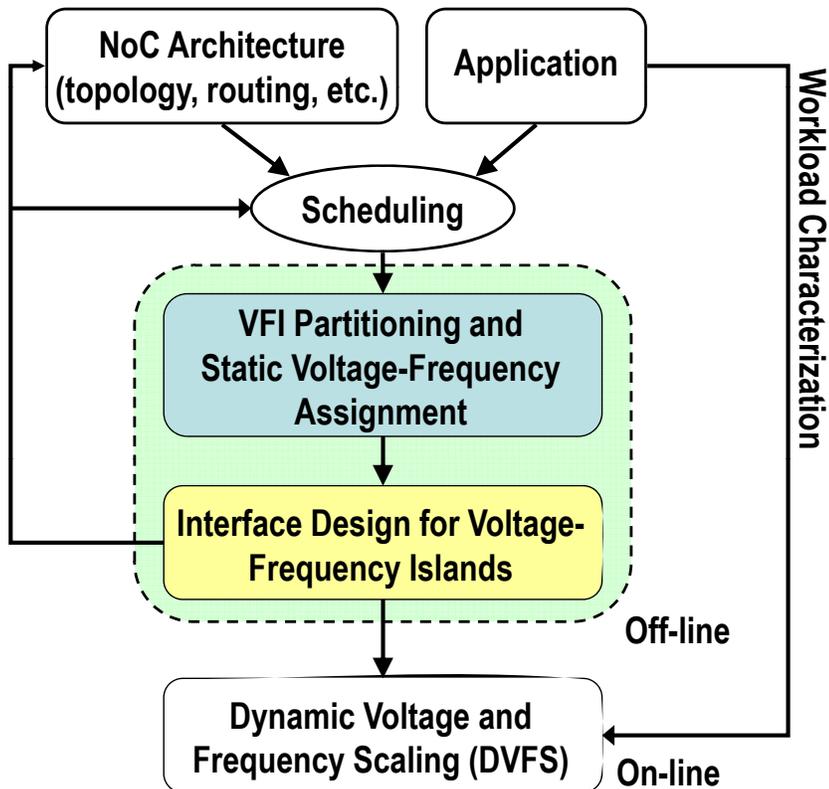
102

SoC VFI Partitioning

- What is the most efficient partitioning and what are the corresponding voltage/frequency assignments?



Design Methodology for Multi-VFI NoCs



VFI Partitioning Problem

- Given
 - ▼ NoC architecture and a schedule for the driver application
 - ▼ Maximum number of allowed VFIs and physical constraints
- Find
 - ▼ VFI partitioning (i.e., optimum number of VFIs, $n \leq M$)
 - ▼ Assignment of the supply and threshold voltages to each island
- Such that the *total energy consumption* is minimized

$$E_{Total} = \underbrace{E_{App}}_{\text{Application (useful) energy consumption (comp+comm)}} + \sum_{i=1}^n \underbrace{E_{VFI}(i)}_{\text{Overhead of } i^{\text{th}} \text{ VFI}}$$

Number of VFIs

$$E_{VFI} = E_{ClkGen} + E_{Vconv} + E_{MixClkFifo}$$

Voltage/Frequency Assignment Problem

- Given a *VFI partitioning*
- Find supply (V_i) and threshold (V_{ti}) voltage assignments
- Such that application energy consumption is minimized

$$\min E_{App} = \sum_{\forall i \in T} \underbrace{E_i(V_i, V_{ti})}_{\text{Energy consumed when the task is executed at } (V_i, V_{ti})} + \sum_{\forall i \in T} \sum_{\forall j \in T} \underbrace{vol(i, j) E_{bit}(i, j)}_{\text{Communication energy}}$$

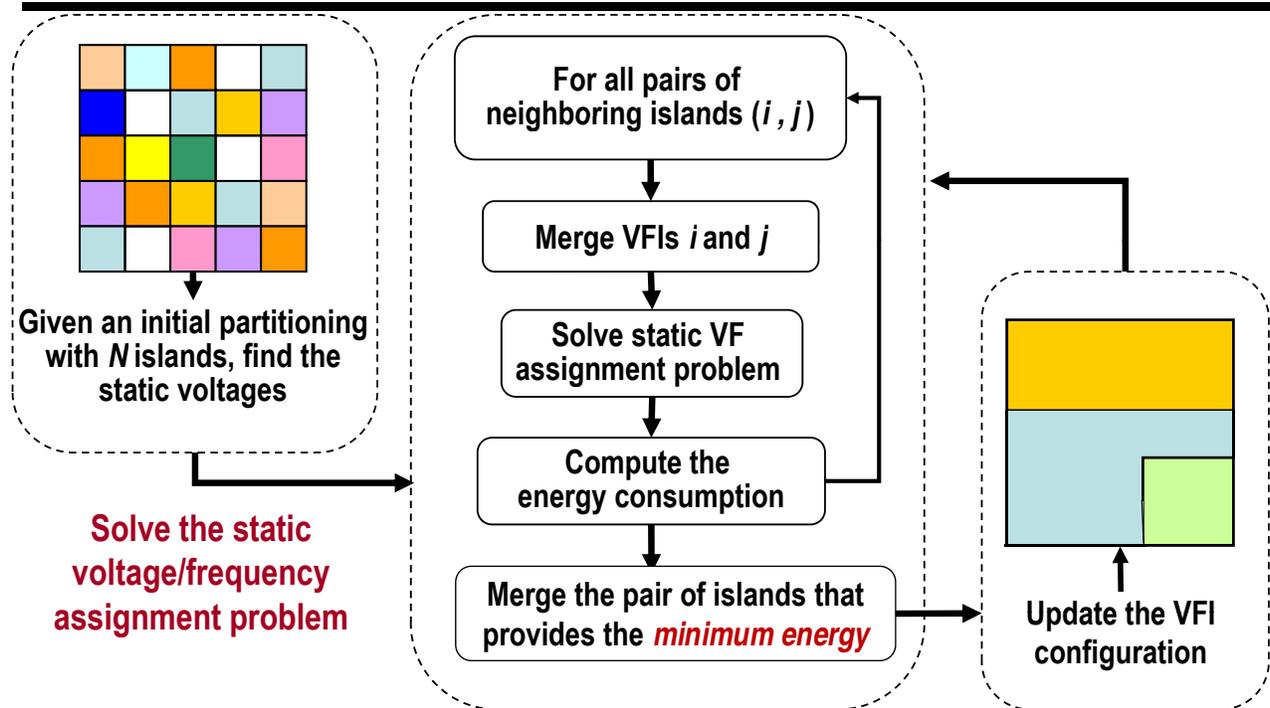
$$E_i(V_i, V_{ti}) = R_i C_i V_i^2 + T_i k_i V_i e^{-\frac{V_{ti}}{S_i}}$$

- ◆ Subject to the following deadline constraints per task t .

$$\underbrace{\frac{x_t}{f_t}}_{\text{Execution time}} + \underbrace{t_{Comm}^t}_{\text{Communication delay}} \leq \text{deadline}_t - \text{start_time}_t$$

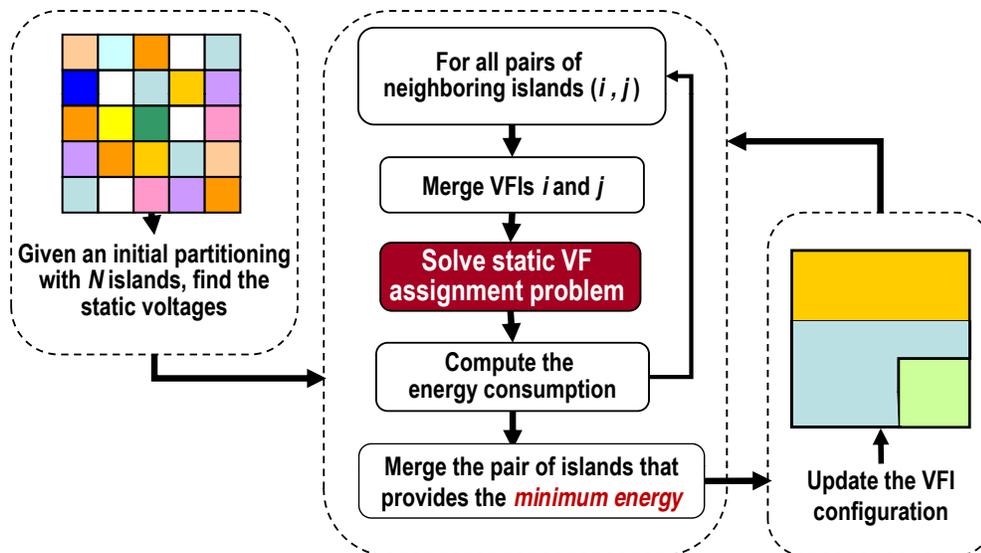
$$t_{comm}(src, dst) = \sum_{i \in P} \frac{\mu_s}{f_i} + t_{fifo} \left\lceil \frac{vol(src, dst)}{W} \right\rceil$$

VFI Partitioning and Voltage Assignment Algorithm



This can be also implemented as a branch & bound algorithm.
We can obtain *exact* results for small examples.

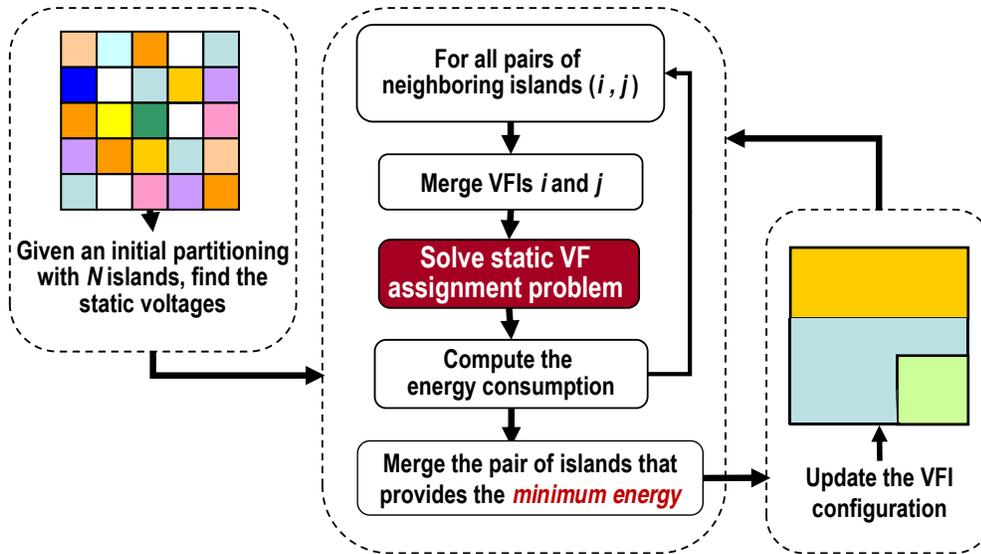
Voltage Assignment Algorithm



$$\min E_{App} = \sum_{\forall i \in T} E_i(V_i, V_{ti}) + \sum_{\forall i \in T} \sum_{\forall j \in T} vol(i, j) E_{bit}(i, j)$$

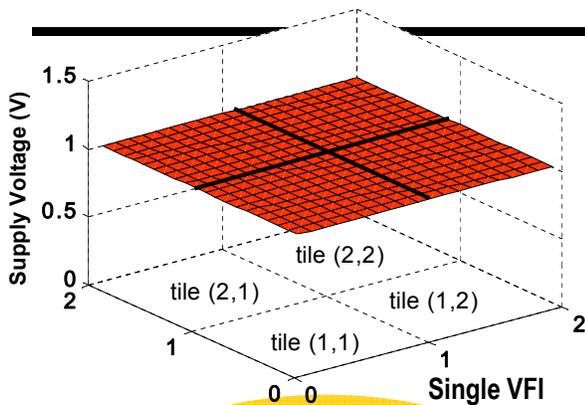
$$\text{subject to } \frac{x_t}{f_t} + t_{Comm}^t \leq \text{deadline}_t - \text{start_time}_t$$

Voltage Assignment Algorithm

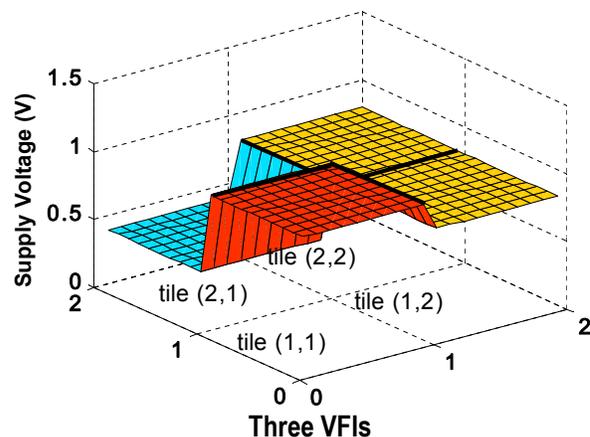
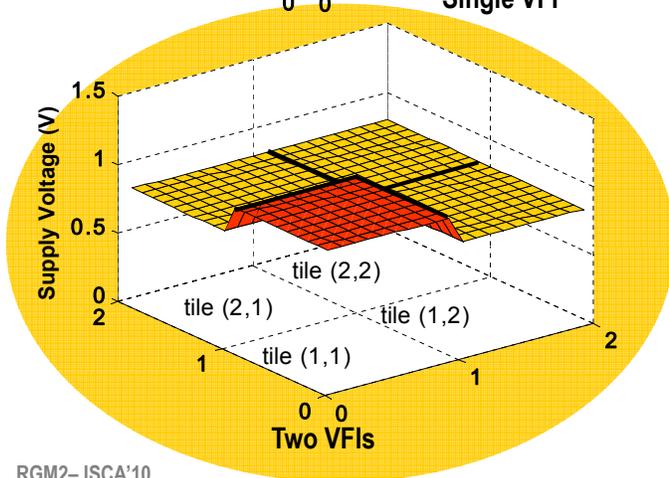


- **Constrained nonlinear optimization or nonlinear programming**
 - ▼ Finds a constrained minimum of a scalar function of several variables
 - ▼ Use Matlab nonlinear solver (*fmincon*)

Why Does VFI Partitioning Matters?

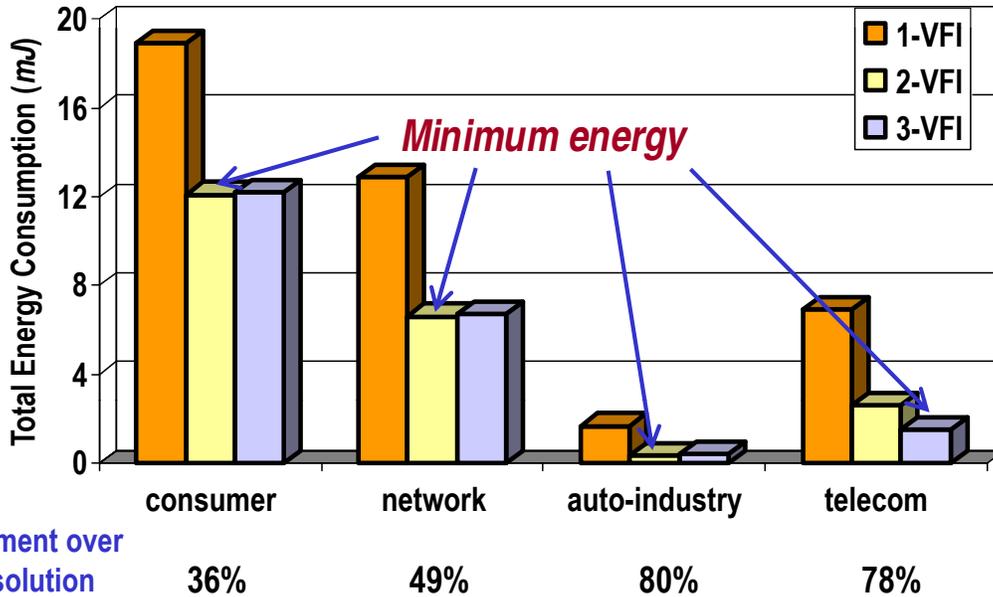


- u Small benchmark scheduled on a 2x2 network using EDF
 - u BPM 70 nm used for the technology parameters
 - u Energy consumption
 - 1-VFI: 10.5mJ
 - 2-VFI: 7.5mJ
 - 3-VFI: 7.6mJ
- ↻ 29%



Experiments with Realistic Benchmarks

- Several E3S benchmarks (*consumer, network, auto-industry, telecom*)
- Applications scheduled to NoCs ranging from 3×3 to 5×5



RGM2- ISCA'10

111

Outline

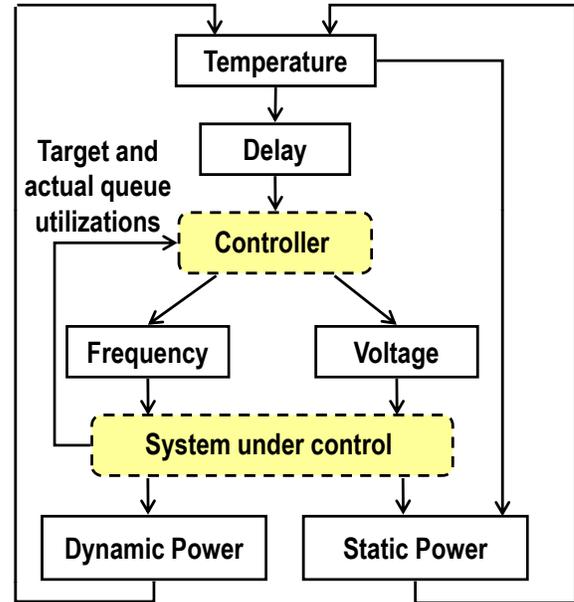
- VFI partitioning
 - ▼ Multi-VFI NoC designs
 - ▼ Partitioning and voltage assignment
 - ▼ Examples
- On-line control
 - ▼ State-based model construction
 - ▼ Feedback control architecture
 - ▼ Stability issues
- Summary

RGM2- ISCA'10

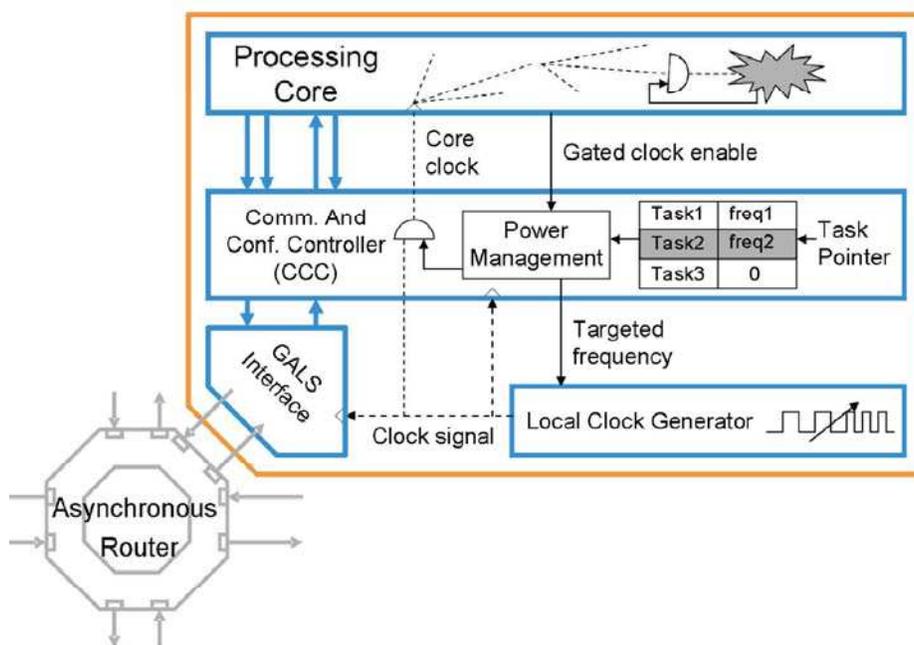
112

Why On-line Control?

- **Cannot rely on nominal values because they vary**
 - ▼ Sources of concern are workload, process, voltage, temperature variations
 - ▼ Cope with the parameter variations which cannot be predicted or accurately modeled at design time
- **Heuristic techniques and manual tuning won't work!**



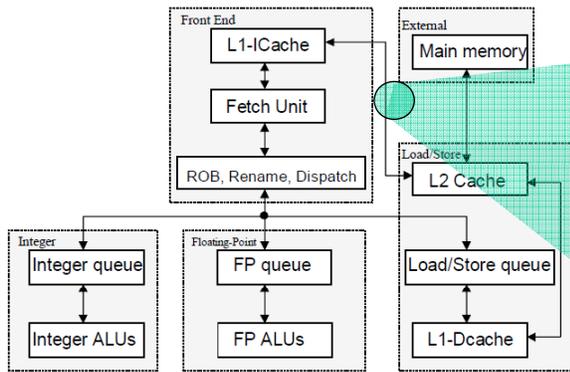
Distributed Power Management in Magali



A 477mW NoC-based digital baseband for MIMO 4G SDR chip organized around a 15-router asynchronous NoC that connects 22 processing units.

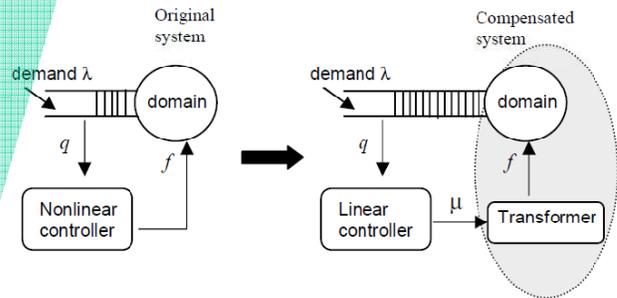
Local Control in Multi Clock Domain Processors

The clock domain partitions in an MCD processor



[Semeraro, et al, HPCA'02]

Interface model between the domains

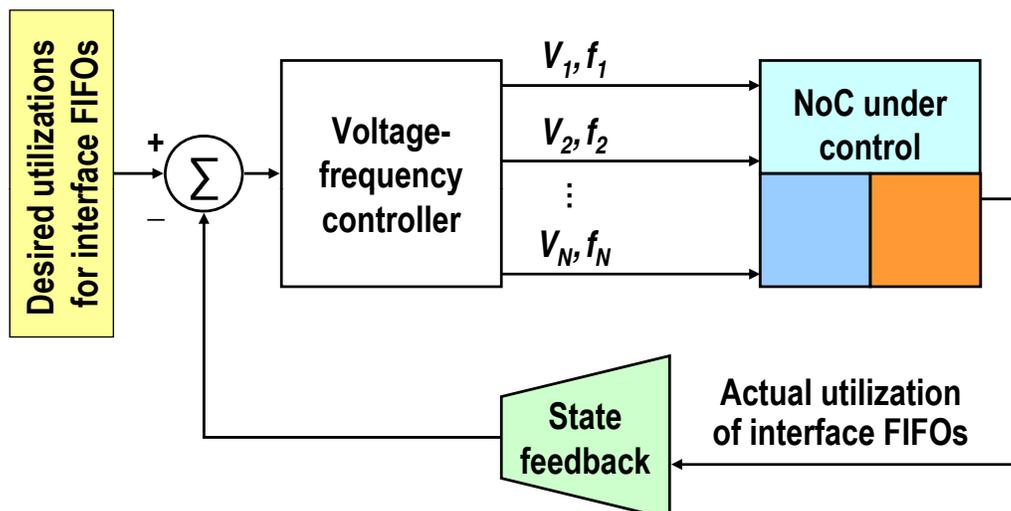


[Wu, et al, ASPLOS'04]

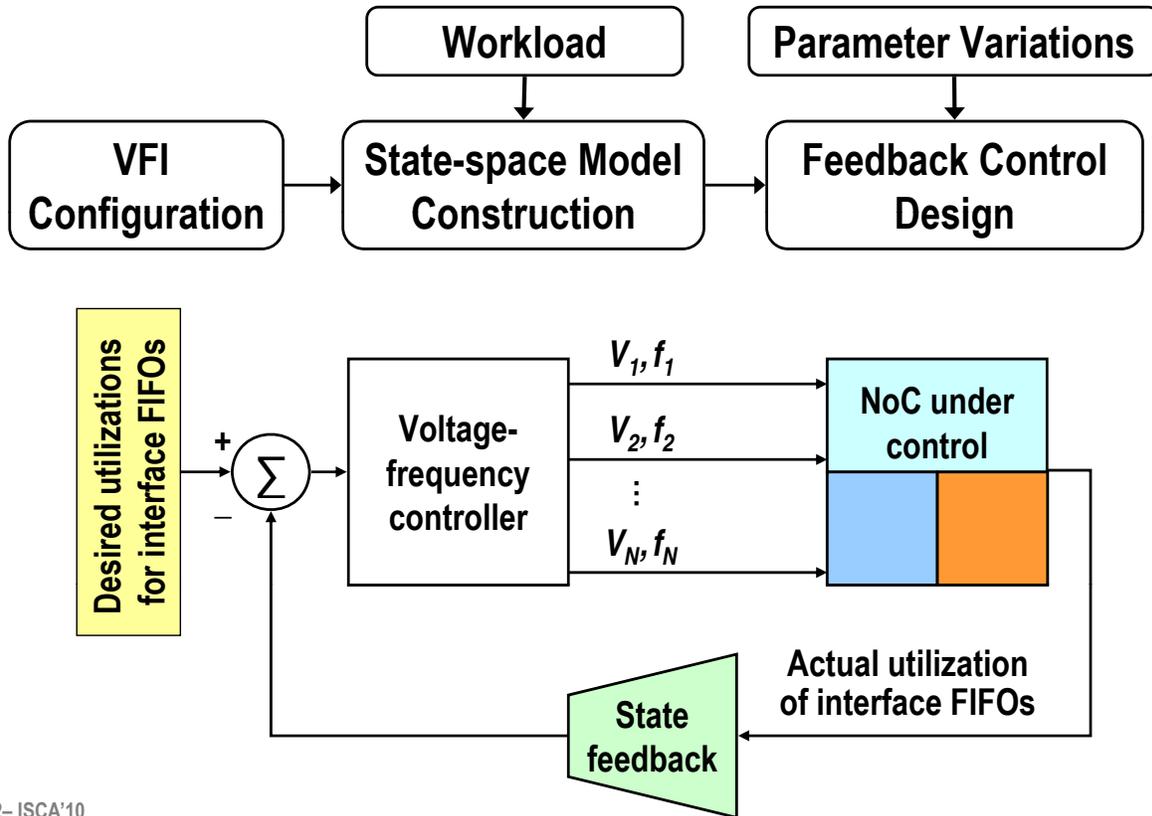
- PID controller for voltage/frequency control proposed previously using only local queue information
 - ▼ Ignores interactions among multiple queues
 - ▼ Works fine if frequency change in one clock domain has negligible impact on other domains
- For an MCD processor with arbitrary partitions and strong interactions among multiple queues, a centralized online DVFS scheme may be needed

Design Methodology for Multi-VFI NoCs

- Traditionally, PID controllers are used due to simplicity. However, state-space modeling brings new opportunities
 - ▼ Precise controllability and stability analysis
 - ▼ Pole placement, linear quadratic regulator, robust controller



Design Methodology for Multi-VFI NoCs



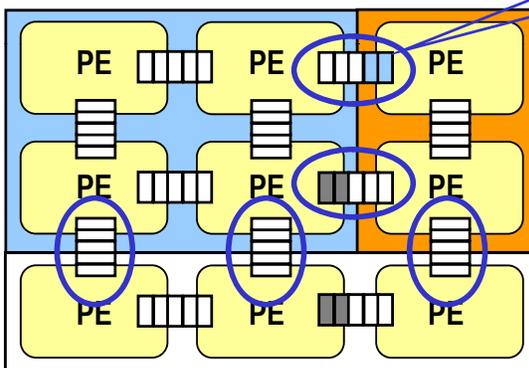
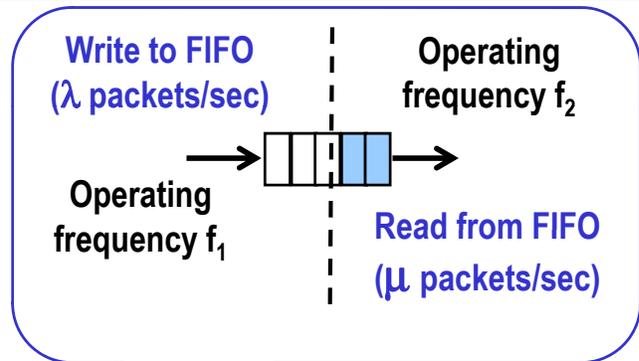
RGM2- ISCA'10

117

Formal Feedback Control

Multi-VFI Network-on-Chip

- Interface queue utilizations are the *states* of the system
- State feedback for voltage-frequency control
- Control interval is $T \mu\text{sec}$



State (queues utilization)

$$Q = [q_1, q_2, \dots, q_N]$$

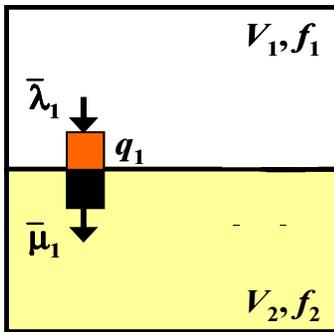
Input (clock speeds)

$$F = [f_1, f_2, \dots, f_M]$$

RGM2- ISCA'10

118

Step-by-step Model Construction (*one queue*)



Average utilization in
the k^{th} control interval

Amount of data (packets)
read from the queue

$$q(k) = q(k-1) + T\lambda_1(k-1) - T\mu_1(k-1)$$

Amount of data (packets)
written to the queue

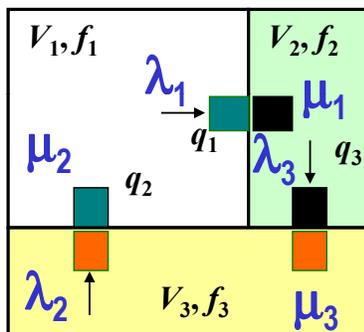
- If data read/write rates are proportional to the frequency of the VFI

$$\lambda_1(k-1) = \bar{\lambda}_1 f_1(k-1), \quad \mu_1(k-1) = \bar{\mu}_1 f_2(k-1)$$

- The state-space equation can be written as

$$q(k) = q(k-1) + T \underbrace{\begin{bmatrix} \bar{\lambda}_1 & -\bar{\mu}_1 \end{bmatrix}}_B \begin{bmatrix} f_1(k-1) \\ f_2(k-1) \end{bmatrix}$$

Step-by-step Model Construction (*three queues*)



$$B = \begin{bmatrix} & & \\ \underbrace{}_{f_1} & & \\ & \underbrace{}_{f_2} & \\ & & \underbrace{}_{f_3} \end{bmatrix}$$

First row $\rightarrow q_1$
Second row $\rightarrow q_2$
Third row $\rightarrow q_3$

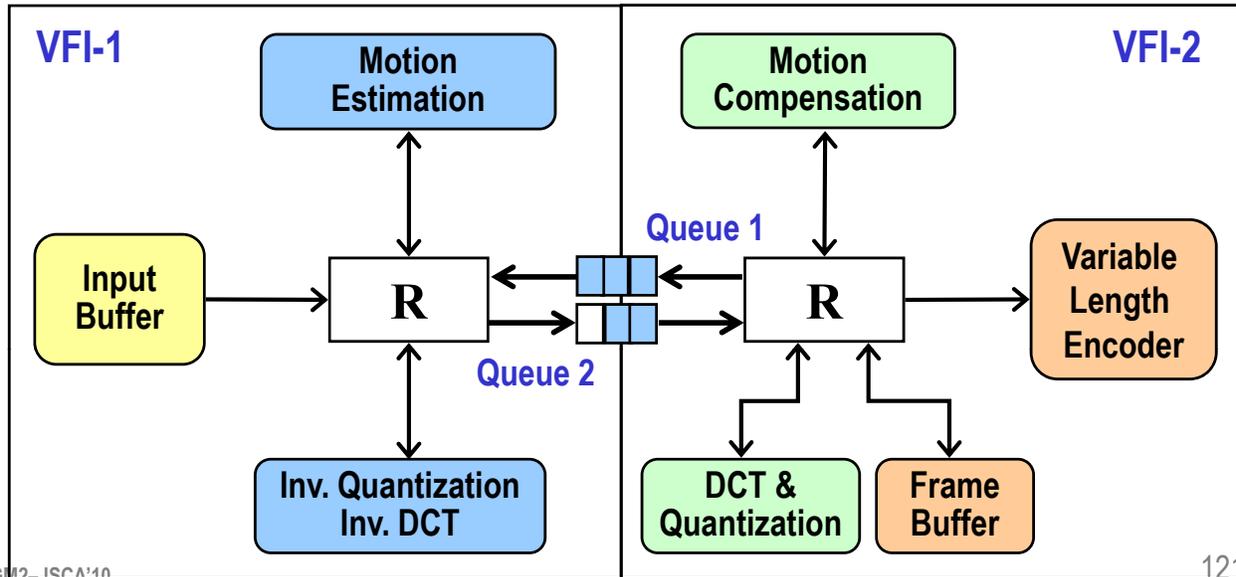
- The topology of the VFIs determines the matrix B
- An algorithm automatically constructs B
- The structure of the model is the same regardless of B

$$Q(k)_{N \times 1} = Q(k-1)_{N \times 1} + TB_{N \times M} F(k-1)_{M \times 1}$$

System Controllability

In the multiple voltage-frequency island system with M islands, utilization of at most M queues can be controlled.

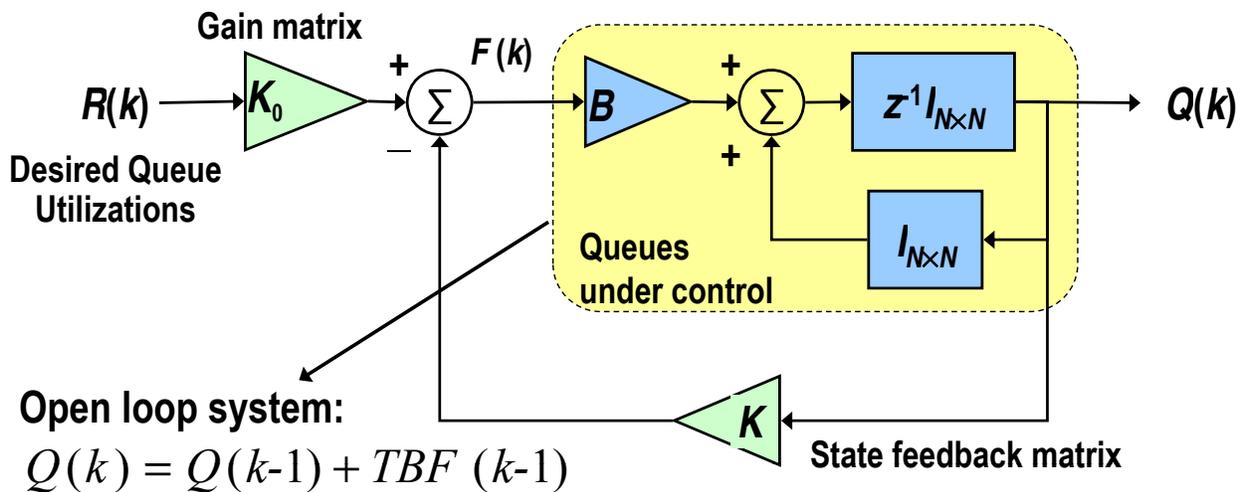
The system is controllable *iff* $\text{rank}(B) = N$ (i.e., number of controlled queues)



RGM2- ISCA'10

12

Feedback Control Architecture



Closed loop system:

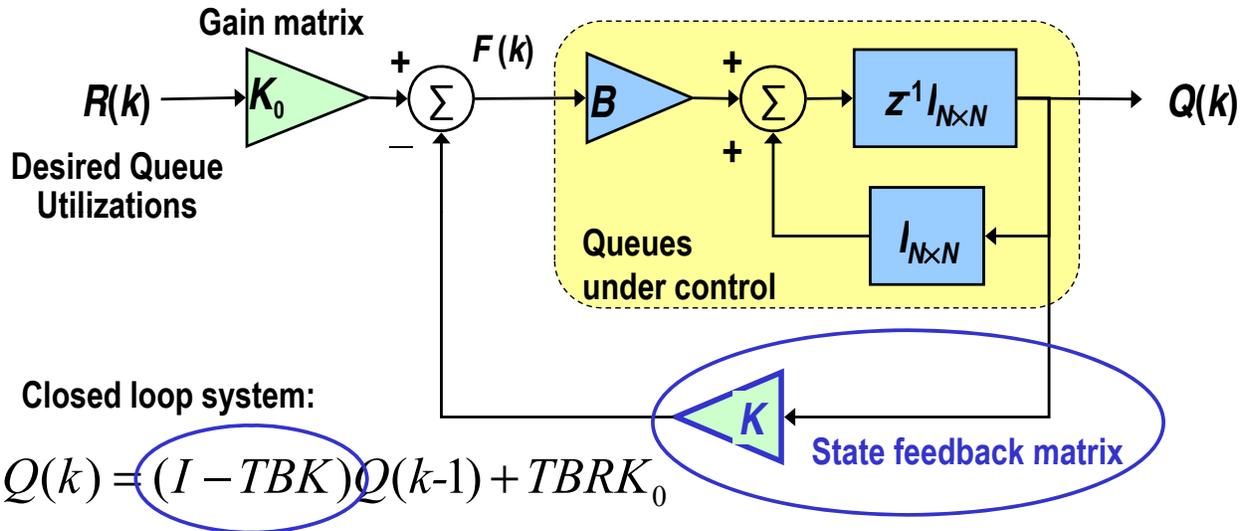
$$Q(k) = (I - TBK)Q(k-1) + TBRK_0$$

$$B = \begin{bmatrix} \bar{\lambda}_1 & -\bar{\mu}_1 & 0 \\ -\bar{\mu}_2 & 0 & \bar{\lambda}_2 \\ 0 & \bar{\lambda}_3 & -\bar{\mu}_3 \end{bmatrix}$$

RGM2- ISCA'10

122

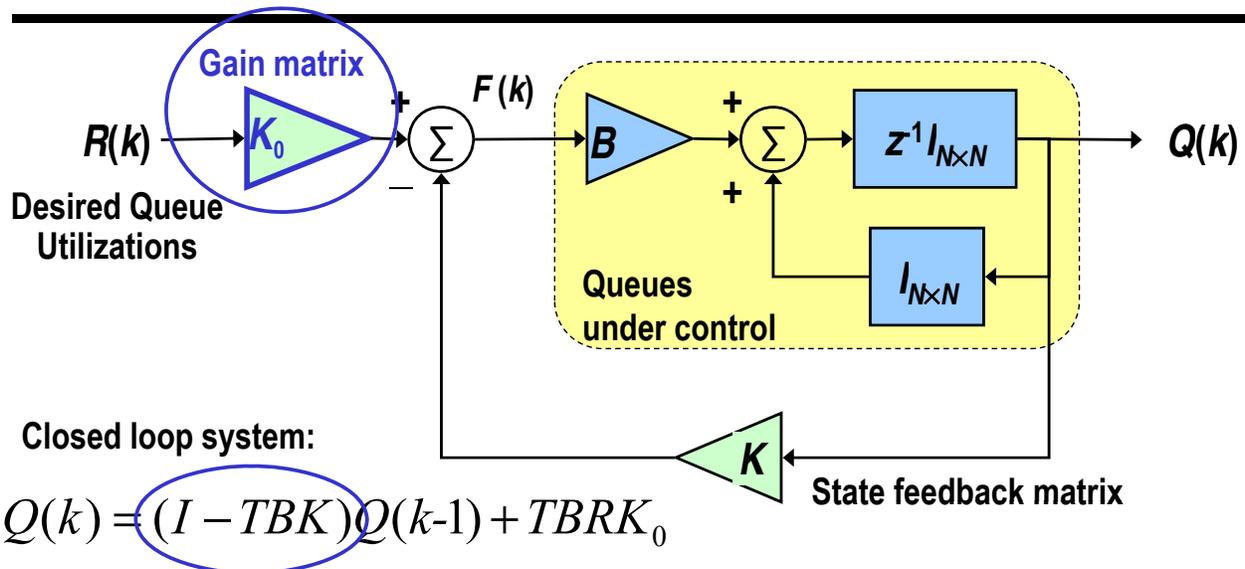
Feedback Control Architecture



■ Design of the state feedback matrix K

- ▼ Find K such that the eigenvalues of the closed loop system are inside the unit circle despite the workload variations
- ▼ Eigenvalue placement, linear quadratic regulator (LQR) design

Feedback Control Architecture



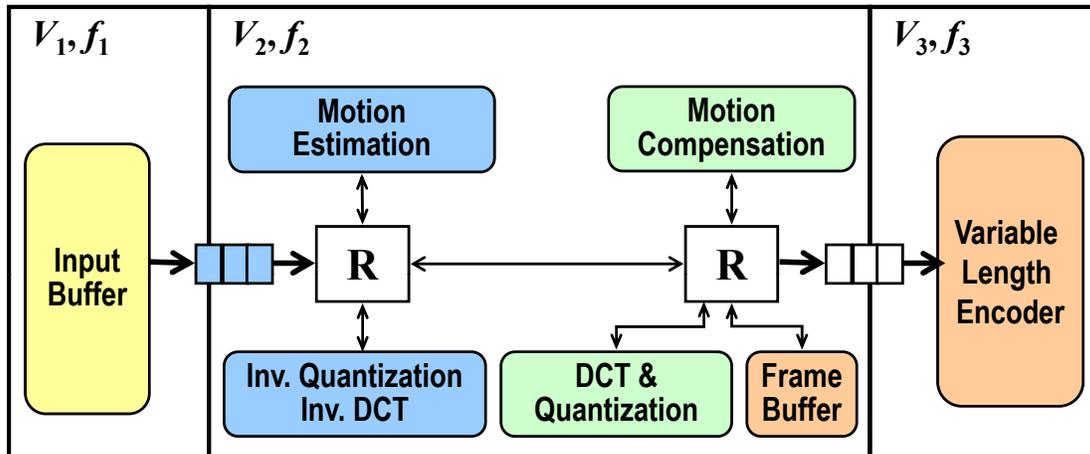
■ By finite value theorem, gain matrix $K_0 = K$

■ Possible extensions

- ▼ Adaptive techniques, such as gain scheduling
- ▼ Monitor the workload and compute K or use values computed off-line

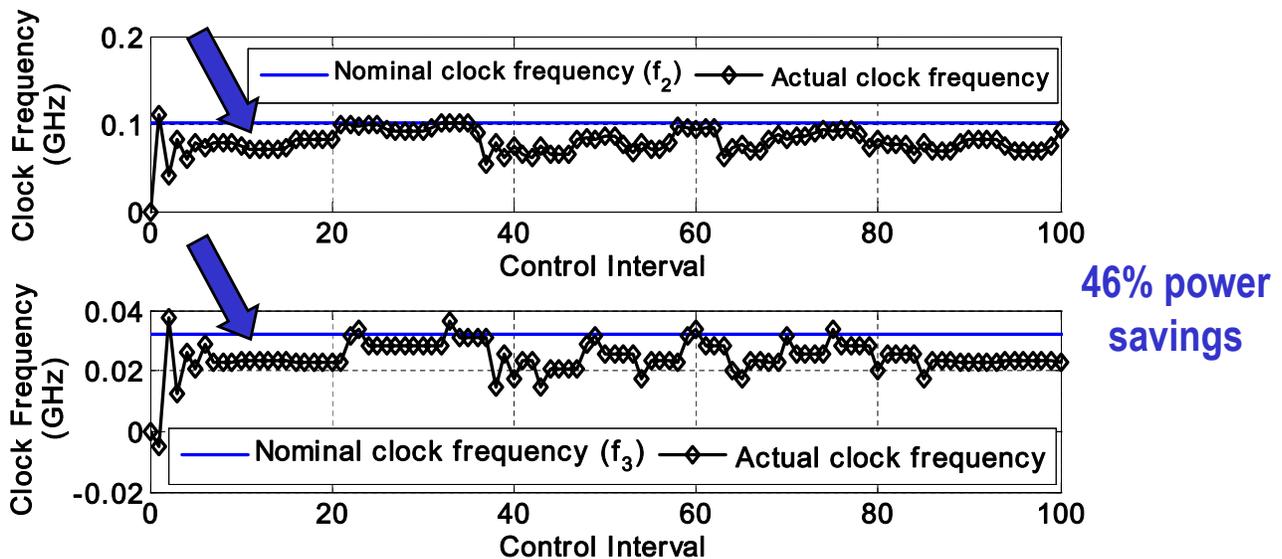
Experiments with MPEG-2 Encoder

- The encoder is divided into three VFI islands and mixed clock FIFOs are used at the interfaces
- The frequency of Variable Length Encoder is set to achieve the desired encoding rate



Frequency Tracking Capabilities

- 50 Frames/sec for 352x288 CIF frames
- f_1 is set to meet the target, f_2 and f_3 follow f_1

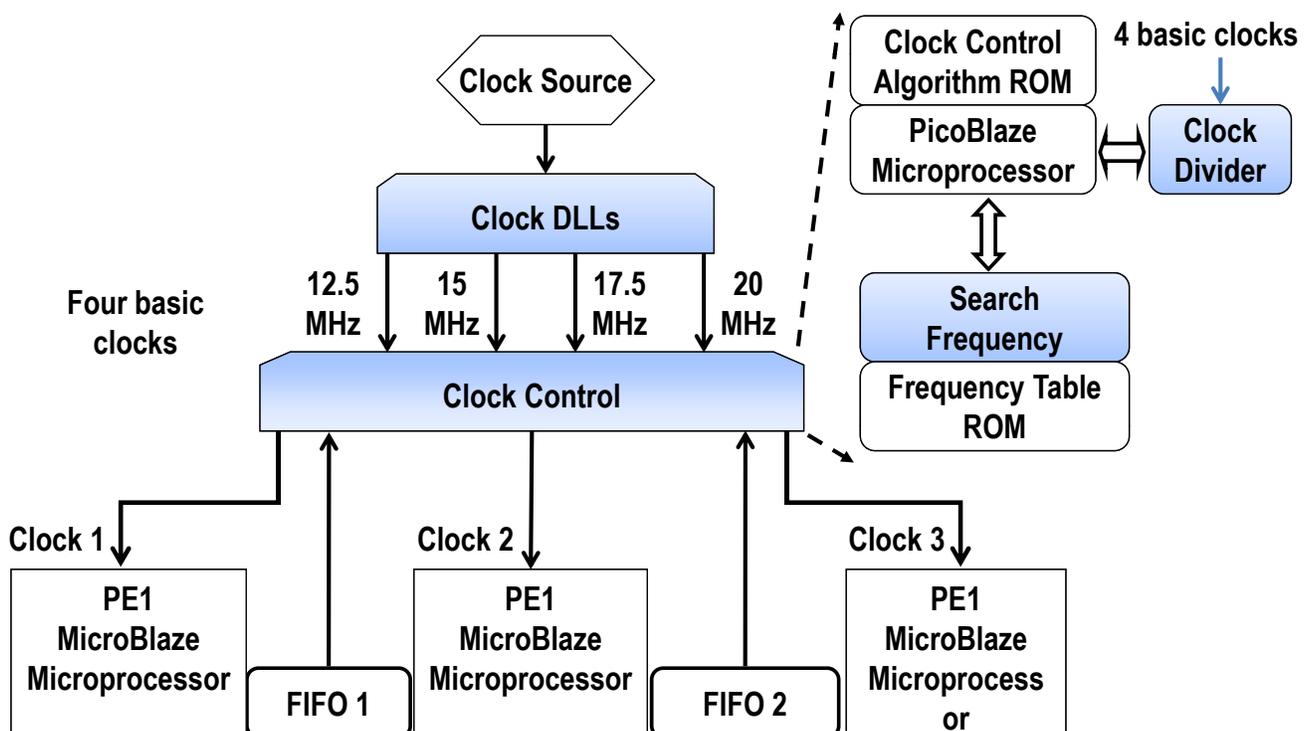


Results on FPGA prototyping

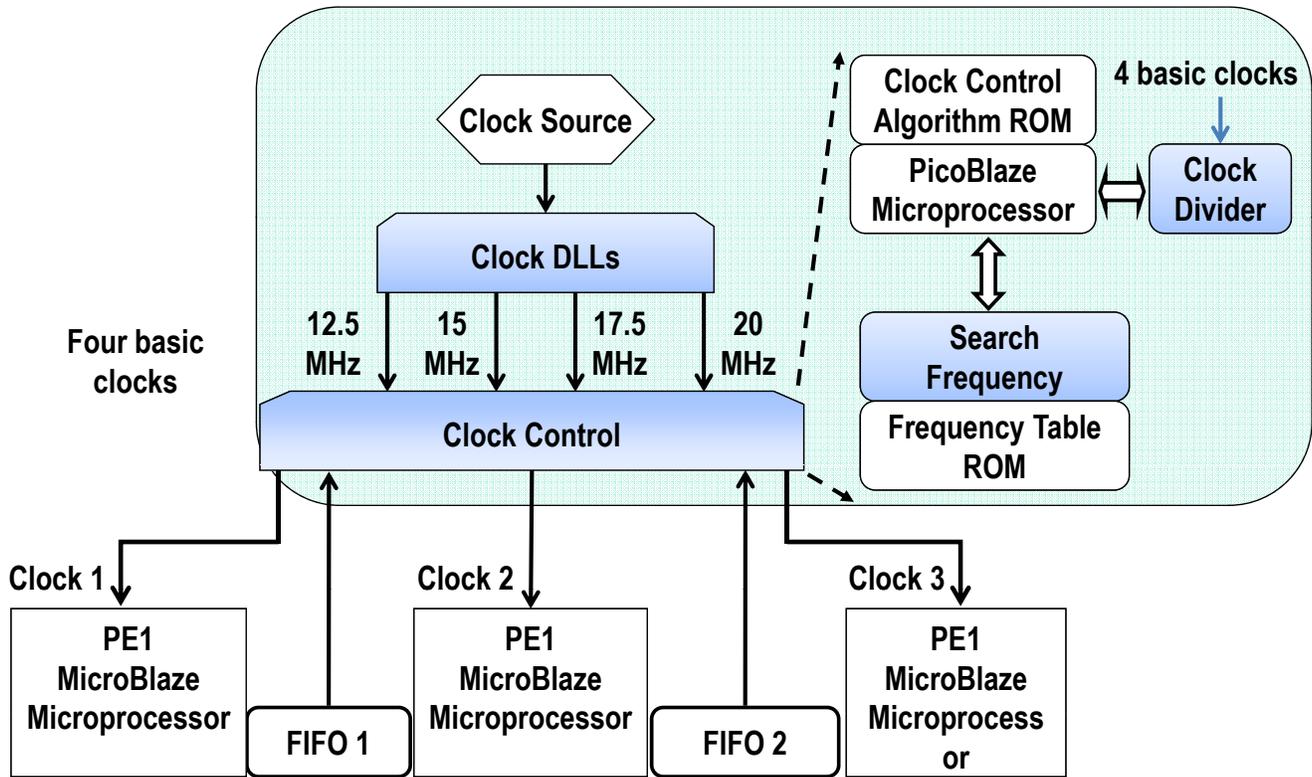
- Work on FPGA prototype using Virtex-II Pro FPGA from Xilinx
- Inter-domain communication
 - ▼ Delay Locked Loops (DLLs) used to generate individual clock signals
 - ▼ Block-RAM based mixed-clock FIFOs
 - ▼ Voltage conversion not supported yet by Xilinx boards
- MPEG-2 encoder design divided into three VFIs
 - ▼ Synchronous design utilizes 16966 LUTs
 - ▼ Design with three VFIs utilizes 19161 LUTs
 - ▼ Power consumption obtained using XPower
 - Without voltage scaling, power drops from 277W to 259W
 - Consistent with simulations

13% overhead

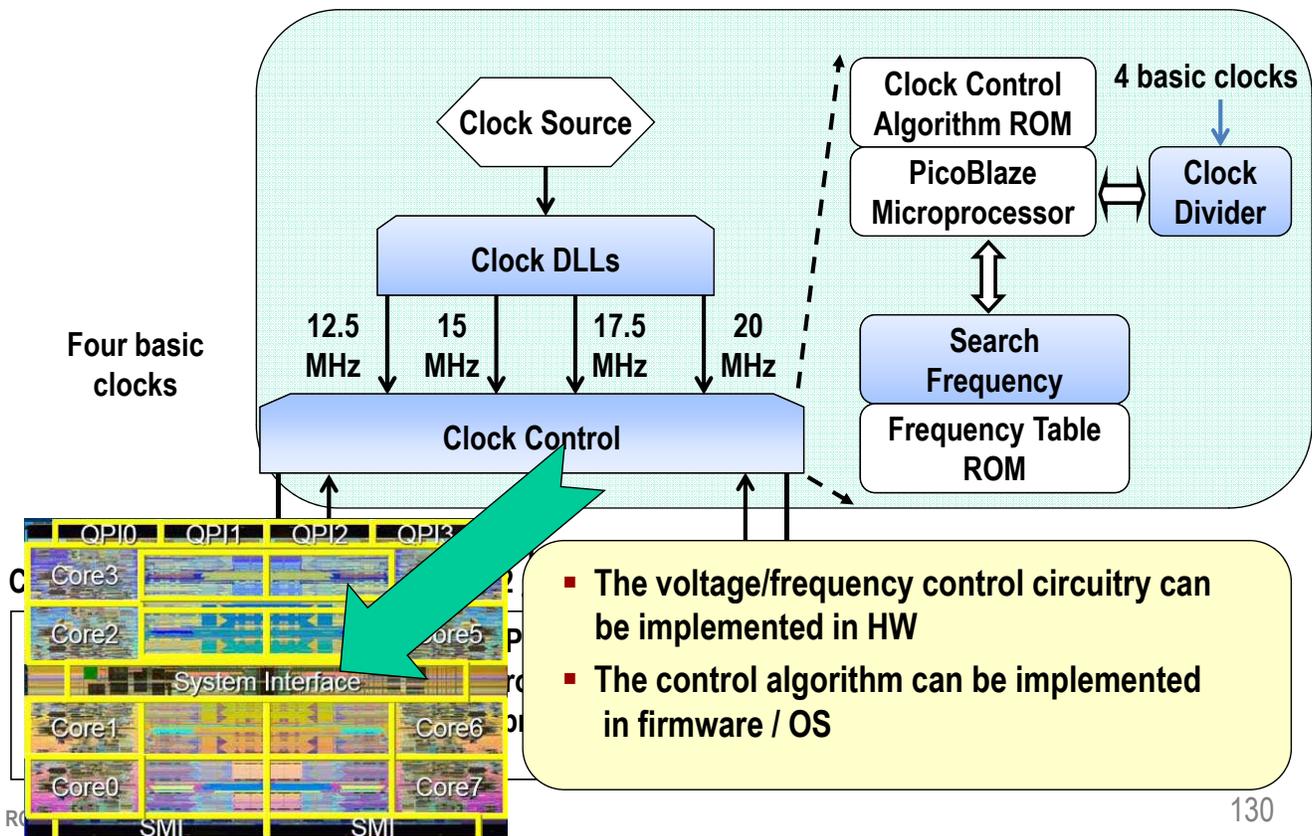
Clock Control Architecture



Clock Control Architecture



Clock Control Architecture



Summary

- **Energy issues in multi-VFI NoCs are crucial**
 - ▼ VFI synthesis via partitioning and voltage allocation
 - ▼ Other formulations are possible
- **Dynamic V/F control yields significant power savings over static approaches while being robust to workload variations**
 - ▼ DVFS controller smoothes out variations in workload characteristics
 - ▼ Precise controllability and stability conditions can be defined
- **More work needed to address**
 - ▼ Adaptive techniques for VFI control
 - ▼ Run-time optimizations for multiple applications
 - ▼ Impact of dynamic traffic on overall DVFS-based power management

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

DVFS in Presence of Process Variations

Diana Marculescu

Carnegie Mellon University

dianam@cmu.edu



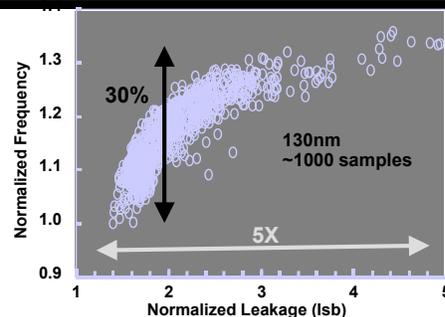
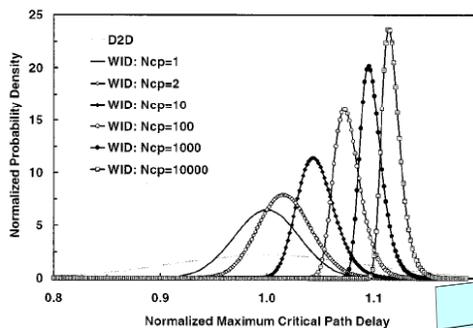
Energy Aware Computing Research Group



Performance – Energy – Variability interactions

- System performance and leakage power severely affected by variability

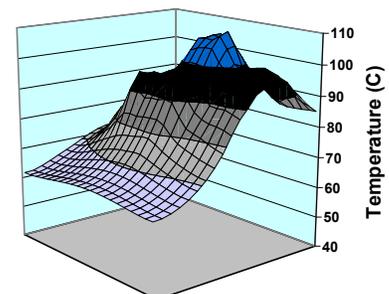
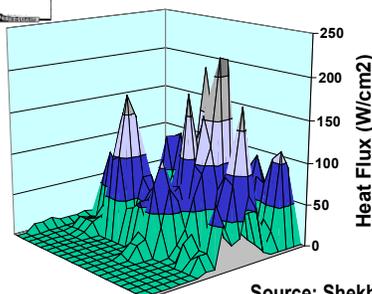
Source: Shekhar Borkar, Intel, DAC 2004

Frequency
~30%Leakage Power
~5-10X

- A whole generation of performance could be lost due to variability

Source: Bowman et. al., JSSC 2002

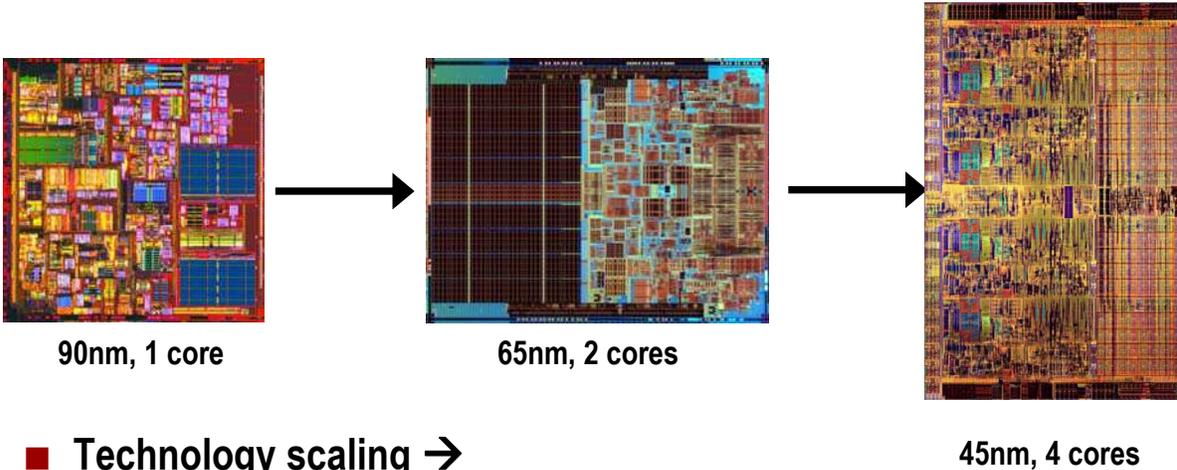
- Increased power density creates hotspots → negatively impacts variability further



Source: Shekhar Borkar, Intel, DAC 2004

Multi-Core, Variations, and Power Management

- Chip-multiprocessor is here → New use of Moore's Law: 2x number of cores every technology generation



- Technology scaling →
 - + Smaller cores (more cores per die)
 - Increased process variability
- Designs moving toward local clock/voltage control → power management per core/DVFS per voltage-frequency island (VFI)

RGM2- ISCA'10

135

Design Variability and Power Management

- Existing power management techniques
 - ▼ Oblivious to manufacturing process induced uncertainty...
 - ▼ ... while relying exclusively on workload-induced variations, without considering design variations
- To be able to cope with increased design variability, power management mechanisms need to:
 - ▼ Incorporate reliable models for design variability early in the process
 - ▼ Be able to work with variability models for *system components* to determine system behavior
 - ▼ Allow for *seamless adaptation* to hardware characteristics, while relying on existing dynamic power management

RGM2- ISCA'10

136

Outline

✓ Motivation

■ Variability and Power Management

- ▼ Variation-aware DVFS
- ▼ Body-Biasing and interaction with DVFS
- ▼ Limits for dynamic power management

■ Summary

Core-to-Core Variations

■ Variations in physical/electrical parameters

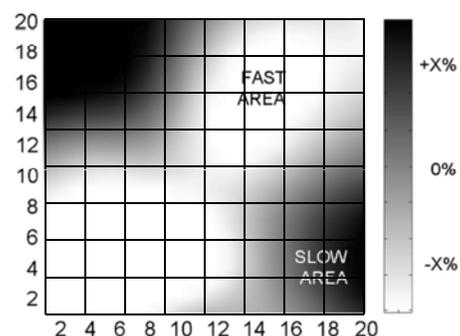
- ▼ Channel length, threshold voltage, oxide thickness
- ▼ Can be lot-to-lot, wafer-to-wafer, die-to-die, *within-die*
- ▼ Can be random and uncorrelated, *random and correlated, systematic*

■ Traditionally, die-to-die component dominated

- ▼ Handled with margins, speed-binning

■ Core sizes decreasing and core counts increasing

- ▼ Intra-die process variations manifest as core-to-core frequency variations



Source: Abulafia et al., TVLSI'06

Dynamic Voltage/Frequency Scaling

- **Dynamic voltage/frequency scaling**
 - ▼ Reduces both dynamic and static power
 - ▼ Control algorithm attempts to lower performance cost

- **Current algorithms are unaware of variations**
 - ▼ [Juang et al. ISLPED'05], [Isci et al. MICRO'06], [Herbert and Marculescu ISLPED'07]
 - ▼ Treat all cores as being identical
 - ▼ Result in wasted power
 - ▼ Single software-based exception [Teodorescu ISCA'08]

- **Variability-aware DVFS attempts to improve energy-efficiency**
 - ▼ Reduce power while maintaining performance [Herbert and Marculescu HPCA'09]

Possible Solutions

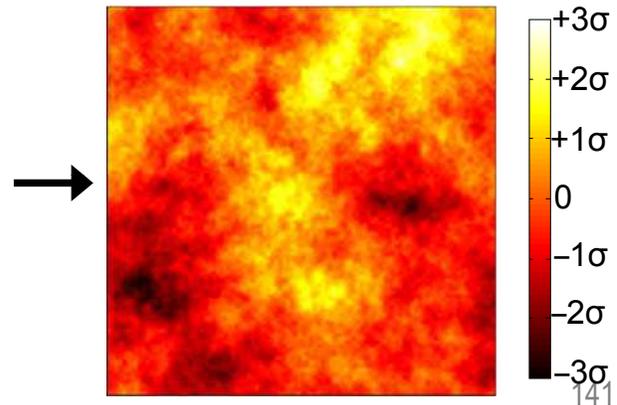
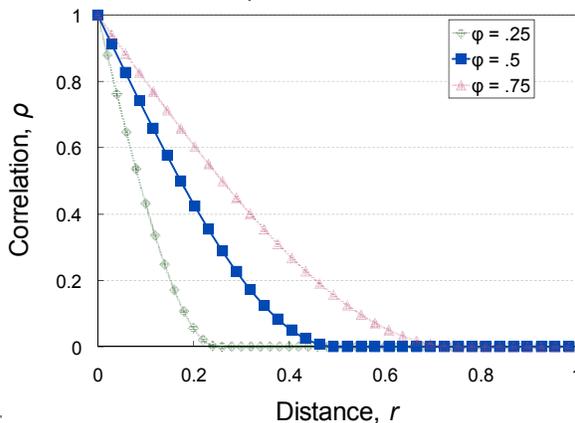
- **Develop dynamic control algorithms from scratch**
 - ▼ Characterize each die separately post-manufacturing
 - ▼ Include those characteristics in newly developed algorithms
 - + Attacks the problem with a specific solution
 - May make a wealth of already existing algorithms potentially unusable
 - Requires pre-characterization at test/post-manufacturing

- **Adapt existing dynamic control algorithms to including variations**
 - ▼ Modify existing algorithms to include per core/per VFI models for process variations
 - + Easy to reuse existing control algorithms
 - Some pre-characterization at test/post-manufacturing required

Physical Parameter Modeling

- Model of spatially-correlated intra-die V_{th} and L_{eff} variation
- Multivariate normal with spherical correlation function [Sarangi TSM'08]

$$\rho(r) = \begin{cases} 1 - \frac{3r}{2\phi} + \frac{1}{2} \left(\frac{r}{\phi} \right)^3 & r \leq \phi \\ 0 & r > \phi \end{cases}$$



RGM2-

141

Core Power/Performance Modeling

- Fit response surface models to SPICE data [Herbert and Marculescu HPCA'09, TCOMP'09]
 - ▼ 22nm hi-K metal gate PTM models [Zhao TED'06]
 - ▼ Lump L_{eff} and V_{th} variation into process variation parameter $p \rightarrow$ capture correlations
- Models for frequency and leakage
 - ▼ As functions of p , V_{dd} , and temperature T
 - ▼ Model form is $e^{\text{polynomial}}$
 - ▼ Frequency model tracks 13-stage FO4 ring oscillator
 - ▼ Leakage model tracks I_{sd} of FETs with $|V_{ds}| = V_{dd}$ and $V_{gs} = 0$
- Models fit to minimize maximum absolute percent error
 - ▼ Iterative pruning removes coefficients until error doubles

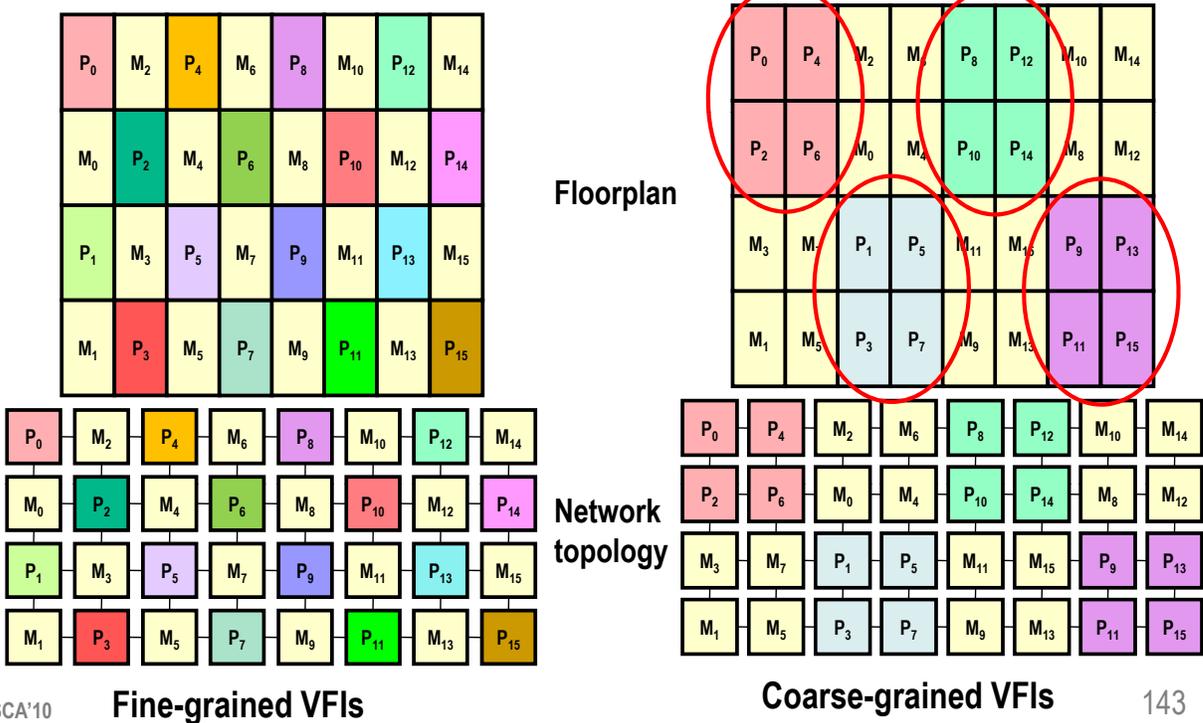
RGM2- ISCA'10

142

Core-to-core Variability Characterization

■ Example: Two chip-multiprocessor designs

▼ Different voltage/frequency island granularity



RGM2- ISCA'10

Fine-grained VFIs

Coarse-grained VFIs

143

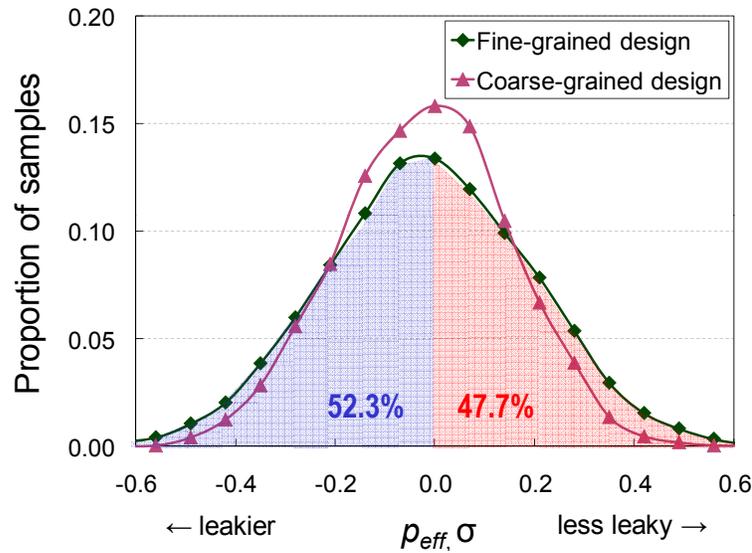
Exposing Variations to DVFS Algorithms

- Variability model has many points per VFI, each with own p
 - ▼ Need a simple way of aggregating and exposing this information at microarchitecture level
 - ▼ p_{eff} is p value that best tracks VFI's leakage across V_{dd} and temperature
- Computed based on four measurements of VFI leakage
 - ▼ High and low temperature and V_{dd}
 - ▼ Can be done based on IDDQ test results
- Negligible error compared to modeling many points per core
 - ▼ Tested across 1,000 dies and 13,456 (V_{dd} , T) pairs
 - ▼ Single-VFI MAPE of 0.011% for single-core VFIs
 - ▼ Single-VFI MAPE of 0.014% for quad-core VFIs

RGM2- ISCA'10

144

Exploiting Variability Information



Bias towards lower V/F levels
Greater power reduction
Greater performance reduction

Bias towards higher V/F levels
Smaller power increase
Smaller performance increase

- Use frequency asymmetry to reduce performance loss
 - ▼ Run leakier cores at lower voltages, but at higher than normal frequencies for those voltages

Threshold DVFS Algorithm

- Keeps utilization in a specified range [Herbert and Marculescu ISLPED'07]
 - ▼ Utilization = (instructions retired / number of retire slots)
 - ▼ Higher VF level → lower utilization
 - ▼ Lower utilization target → higher VF level
- *Threshold-unaware* uses range of [0.2, 0.4] for all VFIs
- *Threshold-aware* sets threshold based on each VFI's variations
- Exponential scaling applied
 - ▼ VFI i thresholds set to $[e^{p_{eff,i}} \cdot T_{down}, e^{p_{eff,i}} \cdot T_{up}]$
 - ▼ Target band [0.279, 0.558] for core in 5th percentile of p distribution (most leaky)
 - ▼ Target band [0.139, 0.278] for core in 95th percentile of p distribution (least leaky)

Greedy DVFS Algorithm

- Attempts to minimize power/throughput (energy per instruction)
 - ▼ [Herbert and Marculescu ISLPED'07] based on [Magklis et al. ISLPED'06]
 - ▼ Performs greedy search to find the optimal VF level
 - ▼ Upon overshooting, moves back to optimal and holds for H intervals
- *Greedy-aware* scales energy per instruction (EPI) to bias VFIs
- Exponential scaling applied
 - ▼ VFI i metric for interval n is $SEPI_{n,i} = EPI_{n,i} \cdot (e^{p_{eff,i}})^{L-n}$

VF level, L	V_{dd}	SEPI scaling for 5 th percentile (leakier)	SEPI scaling for 95 th percentile (less leaky)
0	0.9 V	1.00	1.00
1	0.8 V	0.70	1.39
2	0.7 V	0.49	1.94
3	0.6 V	0.34	2.71

RGM2- ISCA'10

147

Custom Solution: Linear Optimization (*LinOpt*)

- **Linear programming:**
 - Maximize objective function: $f(x_1, \dots, x_n)$, with x_1, \dots, x_n independent
 - Subject to constraints such as: $g(x_1, \dots, x_n) < C$
 - f, g are linear functions
- **Unknowns:** voltages V_1, \dots, V_n for all cores
- **Objective function:** maximize CMP throughput
 - Throughput (MIPS) = Frequency \times IPC = $f(V_1, \dots, V_n)$
- **Constraint:** keep power under P_{target}
 - Power = $g(V)$

- Possible issue: functions f and g are NOT linear; worst-case complexity is exponential [Teodorescu et al. ISCA'08]

RGM2- ISCA'10

148

Experimental Setup

- **SimFlex infrastructure [Hardavellas et al. SIGMETRICS'04]**

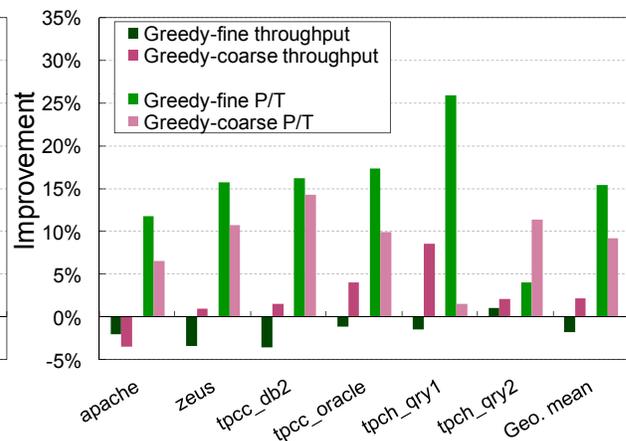
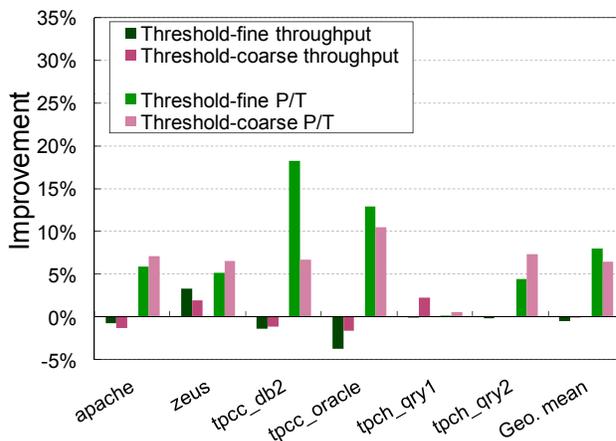
- ▼ Flexus CMPFlex.OoO simulator
- ▼ SMARTS statistical sampling
- ▼ Modified to include power, thermal, variability models, plus a full mesh for on-chip communication

- **Commercial multithreaded workloads**

- ▼ **Web server**
 - SPECweb99 on Apache & Zeus
- ▼ **Online Transaction Processing**
 - TPC-C on Oracle and DB2
- ▼ **Decision Support Systems**
 - TPC-H queries 1 & 2 on DB2

Parameter	Value
Number of cores	16
Nominal frequency	3.0 GHz
Pipeline configuration	8 stages deep 4 instructions wide
ROB/LSQ size	128
Store buffer size	64
L1-I/D cache	Private, 64 KB
L2 cache	Shared, 16 × 1 MB
Main memory	60 ns random access
DVFS interval	50 μs (150K cycles @ 3GHz)

Power-Performance



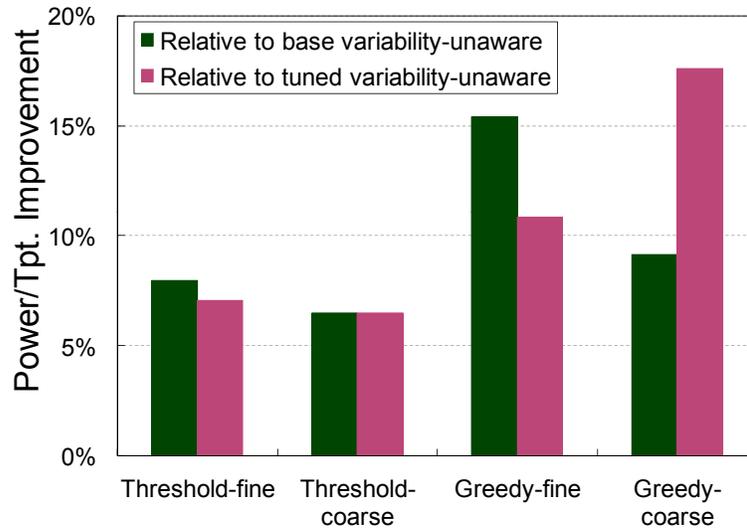
- **Low throughput loss**

- ▼ Worst-case throughput loss under 4%

- **Significant improvement in power/throughput**

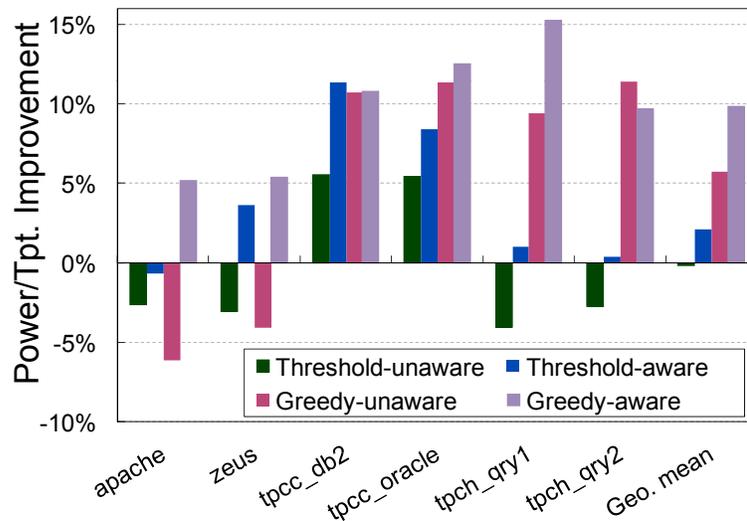
- ▼ 8.0%/6.5% for fine-/coarse-grained VFIs on *Threshold*
- ▼ 15.4%/9.2% for fine-/coarse-grained VFIs on *Greedy*

Comparison at Iso-throughput



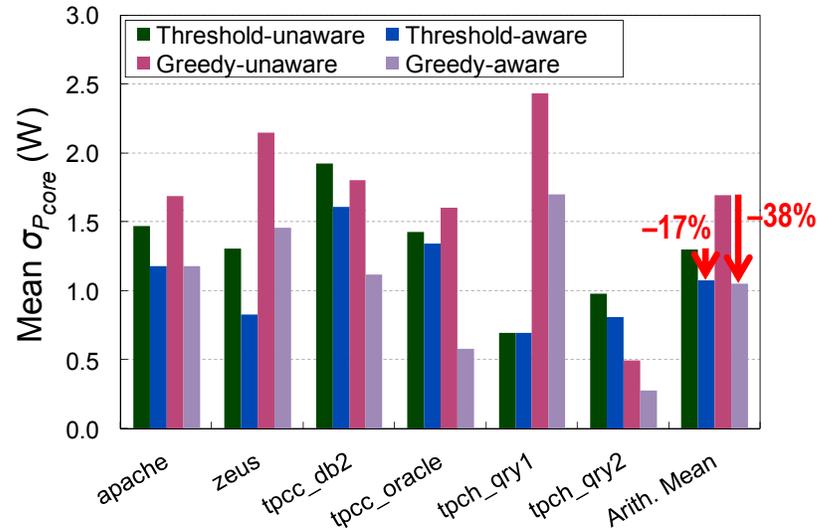
- Tweaked p_{eff} values to achieve iso-throughput between variability-aware scheme and tuned variability-unaware → to quantify the effect of including variability effects when throughput is matched
- Lose some benefit for fine-grained VFIs, gain for coarse-grained

Var.-aware Threshold or Greedy vs. LinOpt



- *LinOpt* power budget set equal to power used by alternate scheme
- Variability-awareness mean benefits up to 10%
- *LinOpt* performance highly dependent on linearizing accuracy

Power Profile Results



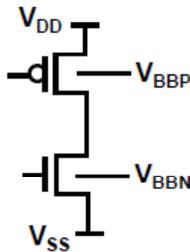
- Variability-awareness smoothes power profile
- Reduces standard deviation of per-core power distribution
 - 17% reduction in mean for *Threshold*
 - 38% reduction in mean for *Greedy*

Outline

- ✓ Motivation
- ✓ Variability and Power Management
 - ✓ Variation-aware DVFS
 - ▼ Adaptive Body-Biasing and interaction with DVFS
 - ▼ Limits for dynamic power management
- Summary

Adaptive Body-Biasing (ABB)

- Powerful technique to mitigate variations in leakage power dissipation [Tschanz et al. JSSC'02]



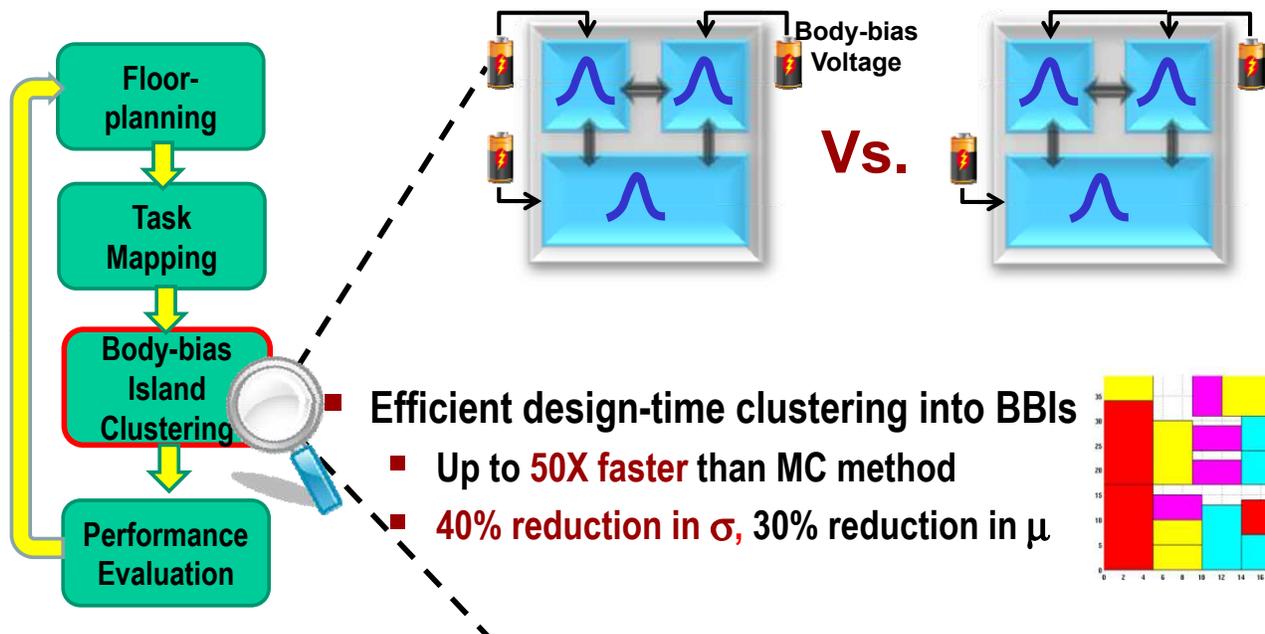
$V_{BBN} > 0$ (Forward Body-bias): \uparrow leakage, \downarrow delay

$V_{BBN} < 0$ (Reverse Body-bias): \downarrow leakage, \uparrow delay

- Each fabricated die tested and assigned appropriate body-bias voltage based on measured leakage current
 - Global Body-bias : single body-bias voltage for entire die
 - Multiple Body-bias : die partitioned into “body-bias islands”

Methodology for a Good Static Solution

- System-level Adaptive Body-bias with Multiple Body-Bias Islands (BBIs) [Garg and Marculescu CODES-ISSS'08]



How About Combining Body Biasing & DVFS?

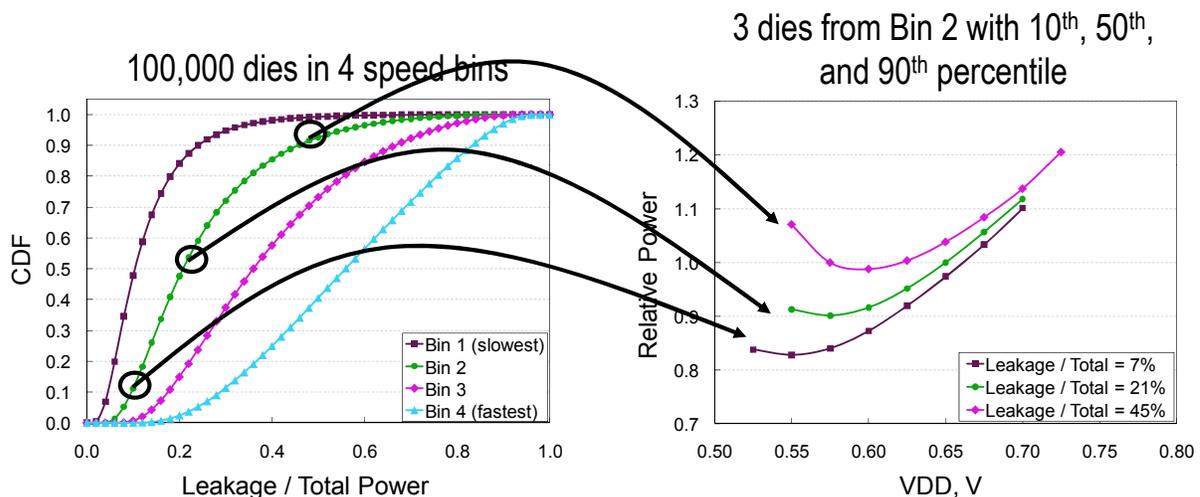
- V_{DD} and body biasing provide two power/performance knobs
 - ▼ Nominal V_{DD} with RBB may not yield best power for given frequency
 - ▼ V_{DD} has stronger effect on dynamic power, BB has stronger effect on static power
- Independent implementation of DVFS and body biasing
 - ▼ DVFS algorithms may stay unchanged
 - ▼ Reclaim frequency margin by adjusting body bias until frequency is exactly met
- *Integration of DVFS and body biasing*
 - ▼ The system specifies desired frequency
 - ▼ Goal is to choose the V_{DD} / BB combination with the lowest total power

RGM2- ISCA'10

157

Main Motivation

- Large variation in leakage as percent of total power
- Leads to large variation in optimal V_{DD} / BB combination
 - ▼ Lower % leakage needs lowest V_{DD} for minimum power
 - ▼ Higher % leakage needs highest V_{DD} for minimum power



RGM2- ISCA'10

158

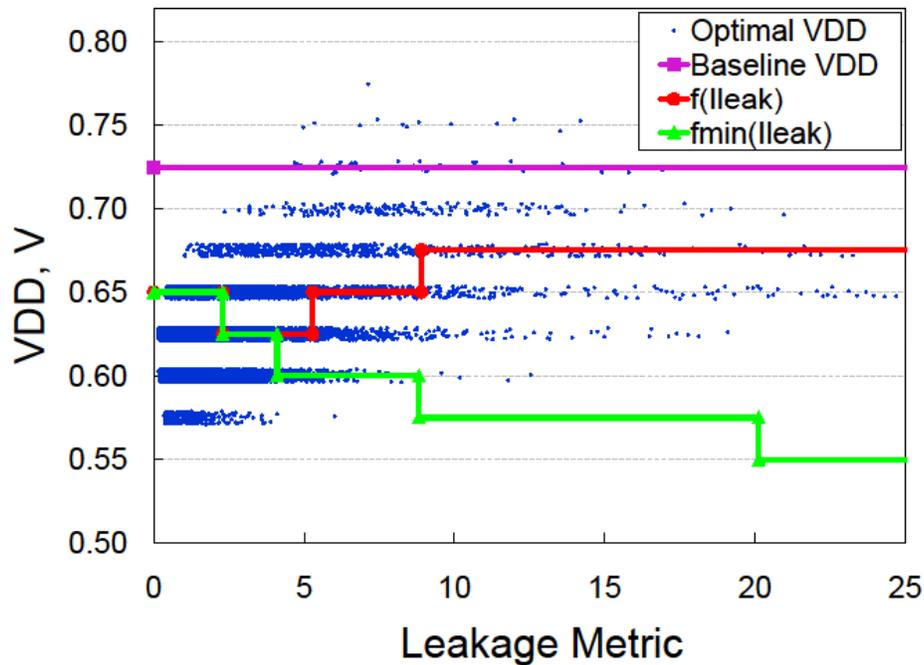
Proposed Approach

- Delay the $\{ V_{DD}, f \}$ mapping until test-time [Bonnoit et al. ISLPED09]
 - ▼ Exploit variability information for each core (or VFI)
- Available frequency levels are chosen as usual
- V_{DD} for each f is chosen at test based on a single leakage measurement
- Function mapping leakage to V_{DD} for each f defined once per technology / product
- Body biasing used to reclaim performance margin

Determining leakage to V_{DD} mapping $f(I_{LEAK})$

- Need to consider both optimal and minimum V_{DD}
 - ▼ Need to ensure most dies can meet frequency with V_{DD} at worst-case temperature
 - ▼ Want to get as close to optimal V_{DD} as possible at typical temperature
 - ▼ Two dies might have same leakage, but optimal V_{DD} of one is lower than minimum V_{DD} of the other
- Methodology – determine :
 - ▼ For each chip
 - Speed bin
 - For each core
 - Leakage @ typical temperature, highest V_{DD} , zero BB
 - For each frequency level
 - » Optimal V_{DD} @ typical temperature, with body biasing
 - » Minimum V_{DD} @ highest temperature, full FBB

Example Mapping

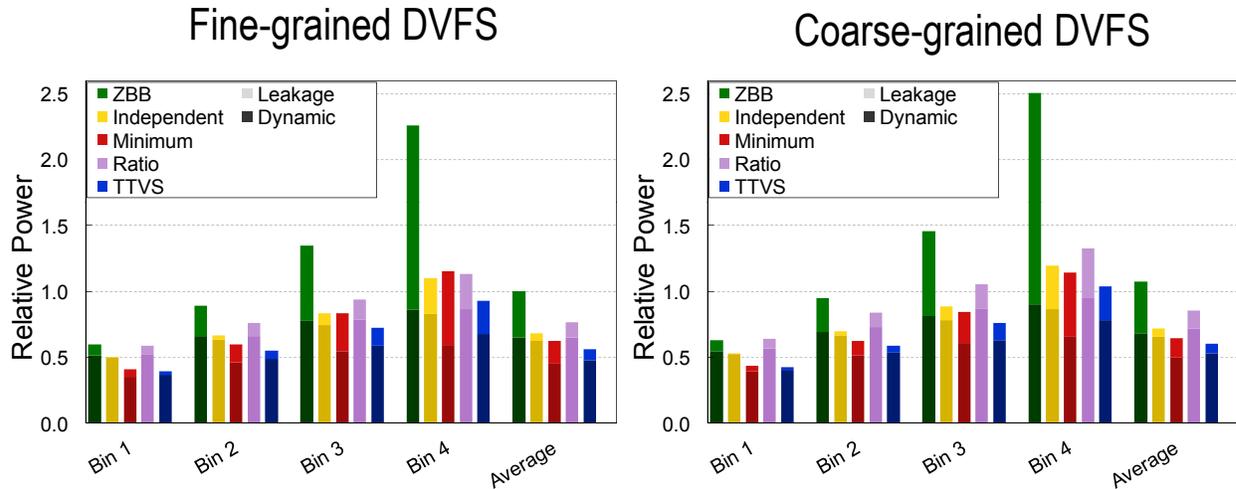


- For each speed bin and frequency level, find Vdd by minimizing the overall error from the optimal Vdd across all cores

Example Schemes

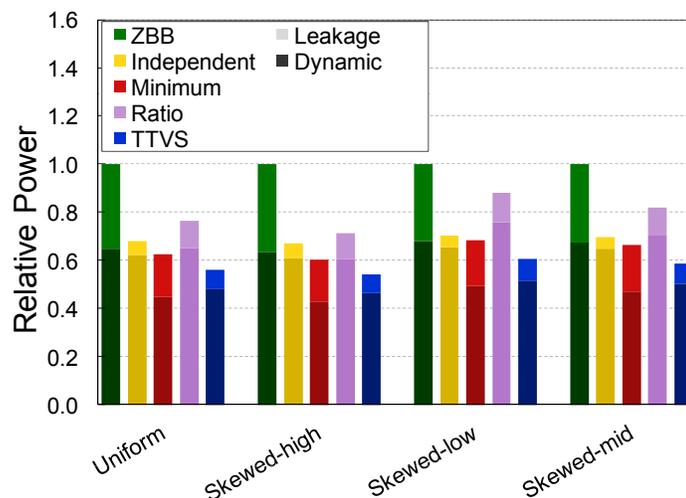
- **ZBB**: Traditional DVFS with no body-biasing
- **Independent**: Independent ABB and DVFS
- **Minimum**: ABB and DVFS, run each core at its minimum V_{DD}
 - ▼ Assume this could be found at test (even though cost would be prohibitive)
- **Ratio**: V_{DD} and BB set to meet f , achieve target ratio of switching to total power [Nomura JSSC'06]
 - ▼ Ratio chosen to minimize total power across all frequency levels in Monte Carlo
- **TTVS**: Proposed scheme

Results by Speed Bin and DVFS Granularity



- **Minimum** generally reduces power, but not when leakage is high
- **Ratio** does poorly because optimal power ratio highly variable
- **Results very consistent**
 - ▼ **TTVS** saves 18% / 16% of power relative to *Independent*
 - ▼ **TTVS** saves 11% / 7% of power relative to *Minimum*

Results by DVFS Time Distribution



- **Same trends hold across various distributions of time per voltage level**
 - ▼ **Uniform** → the amount of time spent at each voltage level is uniformly distributed
 - ▼ **Skewed-high, low or mid** → time spent at each level is proportional to the level i , $n-i$, or $\min(n, n-i)$ (n = total number of levels)

Outline

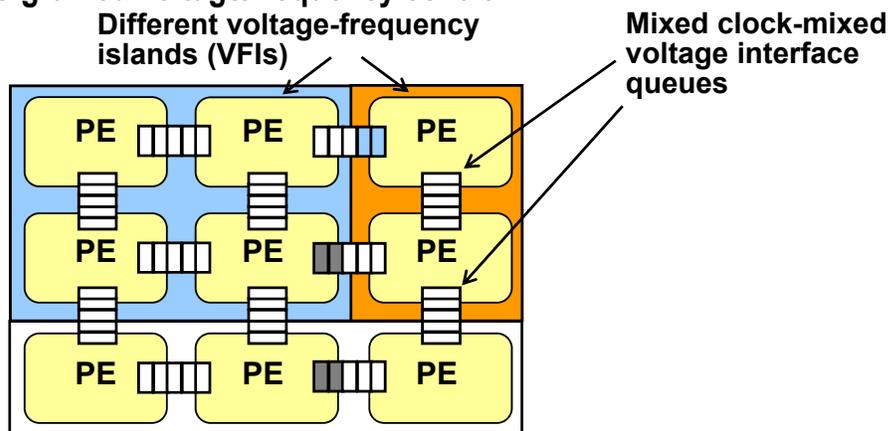
- ✓ Motivation

- ✓ Variability and Power Management
 - ✓ Variation-aware DVFS
 - ✓ Body-Biasing and interaction with DVFS
 - ▼ Limits for dynamic power management

- Summary

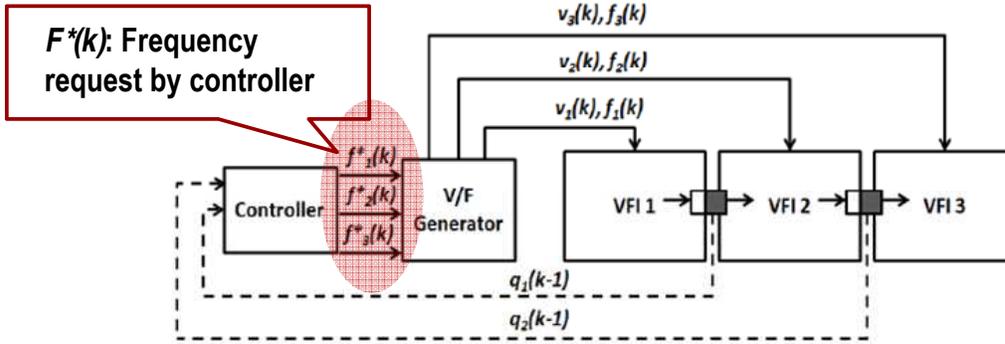
What Are the Limits of Controllability?

- Consider the case of systems with on-chip network-based communication and multiple VFIs
 - ▼ Increased scalability
 - ▼ Reduced design complexity
 - ▼ Fine-grained voltage/frequency control



- What are the limits in controlling such systems *in the presence of variations?*

Technology-driven Limits



- Due to technology constraints, V/F controller cannot always ensure $F^*(k) = F(k)$

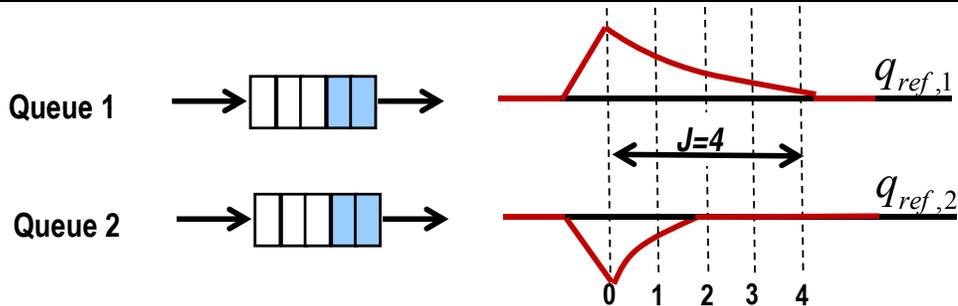
- ▼ Reliability driven upper limit on voltage/frequency

$$F(k) \leq F_{\max}$$

- ▼ Inductive noise and voltage-regulator speed limited constraint on maximum frequency increment

$$|F(k+1) - F(k)| \leq F_{\text{step}}$$

Performance Specification



- Reference queue occupancies: $Q_{\text{ref}} \in \mathbb{R}^N$
- Assume $Q(0) \neq Q_{\text{ref}}$ at control interval 0
- The controller must bring the queues back to the reference value in **at most** J control intervals: $Q(J) = Q_{\text{ref}}$

How do technology-driven factors impact DVFS control performance?

Time-optimal Control

- Queue occupancies after J steps can be written as:

$$Q(J) = Q(0) + TB \sum_{k=0}^{J-1} F(k) \quad (1)$$

- Technology-driven constraints from before:

$$F(k) \leq F_{\max} \quad (2)$$

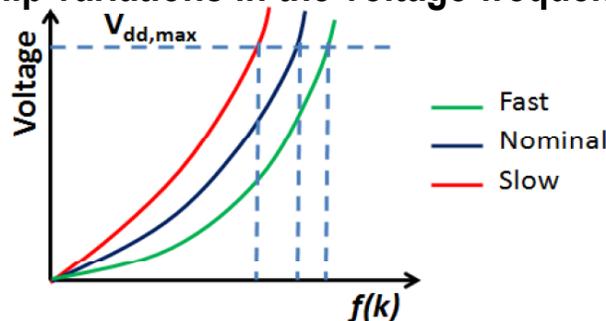
$$|F(k+1) - F(k)| \leq F_{\text{step}} \quad (3)$$

- (1), (2) and (3) define a Linear Program (LP)

- ▼ If the LP is *feasible* the solution $[F(0), F(1), \dots, F(J-1)]$ is a time-optimal control strategy
- ▼ If the LP is *infeasible* the performance specification cannot be met, irrespective of the control algorithm

Process Variations

- Impact of process variations on DVFS controller performance [Garg et al. DAC09]
- Chip-to-chip variations in the voltage-frequency curves of VFIs



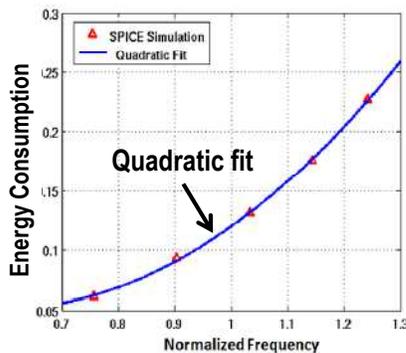
- Assuming maximum supply voltage is fixed, F_{\max} will be different for each manufactured die
- Probability of Controllability (PoC): % of manufactured die that meet performance specification

Explicit Energy Minimization

- Objective: minimize energy spent by controller to reach the steady state

$$E_{total} = \sum_{i=1}^M \sum_{k=1}^J TC_i V_i(k)^2 f_i(k)$$

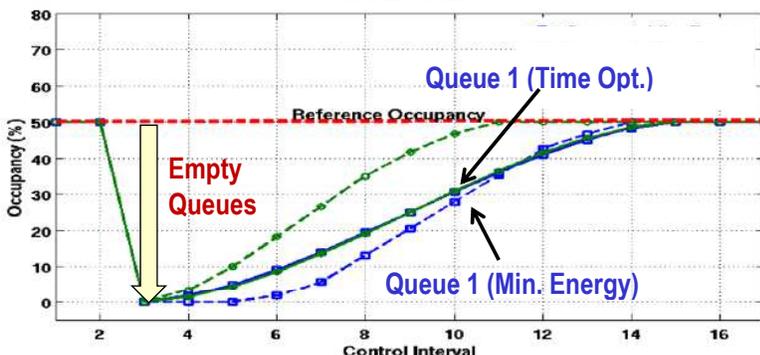
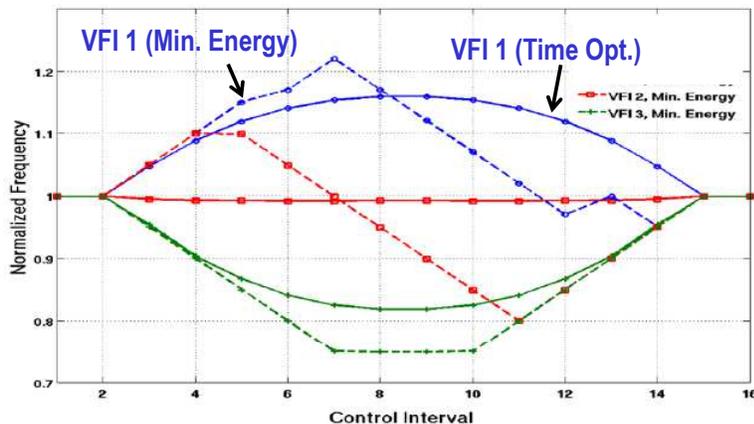
- Approximate E_{total} only as a function of $f_i(k)$ since constraints are also functions of the VFI frequencies



$$E_{total} \approx \sum_{i=1}^M \sum_{k=1}^J \alpha_i f_i(k)^2 + \beta_i f_i(k) + \lambda_i$$

Previous Linear Programming (LP) problem converted to a Quadratic Program (QP)

Experimental Results



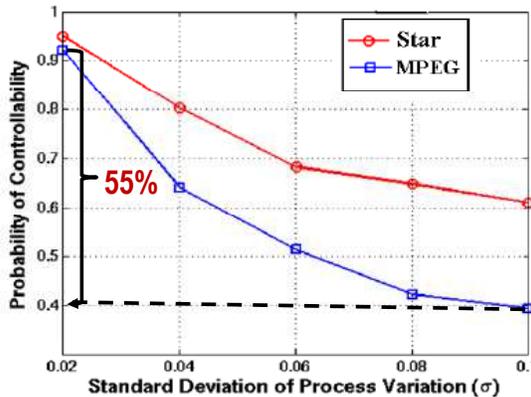
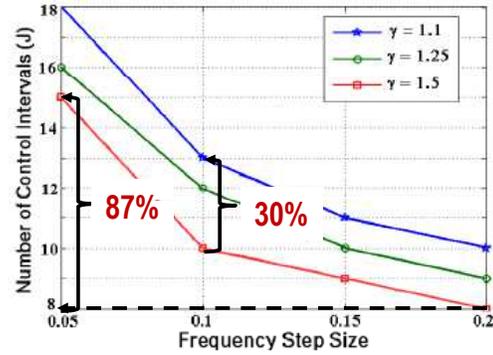
- Controller response for MPEG-2

- Both queues empty
- $J = 10$
- $\gamma = f_{max}/f_{nom} = 1.5$
- $f_{step} = 10\% f_{nom}$

- Observed about 9% energy savings for minimum energy controller

Experimental Results

- Up to 87% loss in performance as step size is reduced
 - ▼ Up to 30% loss in performance as γ is reduced



- ◆ Up to 55% reduction in PoC for increasing magnitude of process variations

Summary

- Including process variation information in power control algorithms is essential
 - ▼ Lose significant power savings at iso-throughput or performance at iso-power
 - ▼ Customized algorithms suboptimal \rightarrow best bet are variation-aware versions of already existing dynamic control algorithms
- Scalability and controllability of dynamic power management algorithms are key
 - ▼ Especially with increased number of cores ...
 - ▼ ... and increased impact of variations multi-core systems

Conclusions

- **Multiple voltage and clock domains are widely used in modern processor design to manage power and process scaling issues**
 - ▼ Increased use of GALS clocking for large SoC designs
- **NOCs are for large SoCs**
 - ▼ Large SoCs = multiple clock domains
 - ▼ NoCs should be asynchronous
- **Dynamic V/F control yields significant power savings over static approaches while being robust to workload variations**
 - ▼ Precise controllability/stability conditions can be defined for DVFS control
- **Including process variation information in power control algorithms is essential**
 - ▼ Lose significant power savings at iso-throughput or performance at iso-power
- **Great research area to work on!**

RGM2- ISCA'10

175

References relevant to this tutorial (General)

- G. De Micheli, L. Benini, 'Networks on Chips: Technology and Tools,' Morgan Kaufmann, 2006.
- W. J. Dally and B. Towles, Principles and Practices of Interconnection Networks. San Mateo, CA: Morgan Kaufmann, 2004
- J. Duato, S. Yalamanchili, and L. Ni, Interconnection Networks: An Engineering Approach. San Mateo, CA: Morgan Kaufmann, 2002
- R. Marculescu, et al., 'Computation and Communication Refinement for Multiprocessor SoC Design: A System-Level Perspective,' in ACM TODAES, Vol.11, No.3, July 2006.
- R. Marculescu, et al., 'Outstanding Research Problems in NoC Design: System, Microarchitecture, and Circuit Perspectives', in IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems (TCAD), vol. 28, no. 1, pp. 3-21, Jan. 2009
- T. Bjerregaard and S. Mahadevan, "A survey of research and practices of network-on-chip," ACM Comput. Surv., vol. 38, no. 1, pp. 1-51, Mar. 2006
- J. Henkel, W. Wolf, and S. Chakradhar, "On-chip networks: A scalable, communication-centric embedded system design paradigm," in Proc. VLSI Des., Jan. 2004, pp. 845-851
- Milos Krstic, et al, "Globally Asynchronous, Locally Synchronous Circuits: Overview and Outlook", IEEE Design and Test of Computers, Sept.-Oct. 2007
- A large selection of NoC papers is available at <http://www.cl.cam.ac.uk/~rdm34/onChipNetBib/browser.htm>
http://www.ocpip.org/university/biblio_main/comparison/

RGM2- ISCA'10

176

References (Part I)

- S. Rusu, et al, "A Dual-Core Multi-Threaded Xeon® Processor with 16MB L3 Cache", ISSCC 2006
- S. Tam, et al, "Clock Generation and Distribution of a Dual-Core Xeon Processor with 16MB L3 Cache", ISSCC 2006
- J. Dorsey, et al, "An Integrated Quad-Core Opteron Processor", ISSCC 2007
- U. Nawathe, et al, "An 8-Core 64-Thread 64b Power-Efficient SPARC SoC", ISSCC 2007
- P. Teehan, et al, "A Survey and Taxonomy of GALS Design Styles", IEEE Design & Test of Computers, Sept–Oct 2007
- B. Stackhouse, et al, "A 65nm 2-Billion Transistor Quad-Core Itanium® Processor", JSSC, Jan. 2009
- Y. Yoshida, et al, "A 4320MIPS Four-Processor Core SMP/AMP with Individually Managed Clock Frequency for Low Power Consumption", ISSCC 2007
- G. Gerosa, et al, "A Sub 2W Low Power IA Processor for Mobile Internet Devices in 45nm Hi-K MG CMOS", A-SSCC 2008
- S. Rusu, et al, "A 45nm 8-Core Enterprise Xeon® Processor", ISSCC 2009
- T. Hattori, et al, "A power management scheme controlling 20 power domains for a mobile processor", ISSCC 2006
- M. Ito, et al, "An 8640 MIPS SoC with Independent Power-Off Control of 8 CPUs and 8 RAMs", ISSCC 2008
- B. Nam, et al, "A 52.4mW 3D Graphics Processor with 141Mvertices/s Vertex Shader and 3 Power Domains of Dynamic Voltage and Frequency Scaling", ISSCC 2007
- Y. Ueda, et al, "A Power, Performance Scalable 8-Cores Media Processor for Mobile Multimedia", A-SSCC 2008
- Y. Shimazaki, et al, "A Shared-Well Dual-Supply-Voltage 64-bit ALU", ISSCC 2003
- J. Shin, et. al., "A 40nm 16-Core 128-Thread CMT SPARC® SoC Processor", ISSCC 2010
- D. Wendel, et. al., "The Implementation of POWER7™, a Highly Parallel, Scalable Multi-Core High End Server Processor", ISSCC 2010
- J. Howard, et. al., "A 48-Core IA-32 Message-Passing Processor with DVFS in 45nm CMOS", ISSCC 2010

References (Part II)

- R. Dobkin, , et al, An asynchronous router for multiple service levels networks on chip, ASYNC 2005.
- R. Dobkin, , et al, QNoC Asynchronous Router with Dynamic Virtual Channel Allocation, NOCS 2007.
- R. Dobkin, R. Ginosar and A. Kolodny, QNoC Asynchronous Router, Integration—The VLSI Journal, 42(2):103-115, 2009.
- E. Beigné, , et al, An Asynchronous NOC Architecture Providing Low Latency Service and its Multi-level Design Framework, ASYNC 2005.
- E. Beigné, F. Clermidy, S. Miermont, P. Vivet, Dynamic Voltage and Frequency Scaling Architecture for Units Integration within a GALS NoC, ASYNC 2008.
- Y. Thonnart, E. Beigné, A. Valentian, P. Vivet, Automatic Power Regulation based on an Asynchronous Activity Detection and its Application to ANOC Node Leakage Reduction, NOCS 2008.
- F.Clermidy, , et al, "A 477mW NoC-Based Digital Baseband for MIMO 4G SDR", International Symposium on Solid State Circuits, ISSCC'10, San-Francisco, USA, Feb. 2010.
- Y. Thonnart, P. Vivet, F.Clermidy "A Fully-Asynchronous Low-Power Framework for GALS NoC Integration", Design Automation and Test in Europe, DATE'10, Dresden, Germany, April 2010.
- F. Clermidy, R. Lemaire, Y. Thonnart and P. Vivet "A Communication and Configuration Controller for NoC based Reconfigurable Data Flow Architecture", Network-on-Chip Symposium (NOCS'09), San-Diego, USA, May 11 - 14, 2009.
- T. Bjerregaard, J. Sparso, A scheduling discipline for latency and bandwidth guarantees in asynchronous network-on-chip, ASYNC 2005.
- T. Bjerregaard, J. Sparso, A router architecture for connection-oriented service guarantees in the MANGO clockless network-on-chip, DATE 2005.
- J. Bainbridge, S. Furber, Chain: a delay-insensitive chip area interconnect, IEEE Micro 22 (5) (2002) 16–23.
- T. Felicijan, S.B. Furber, An asynchronous on-chip network router with Quality-of-Service (QoS) support, Int. SOC Conf. (2004) 274–277.

References (Part III)

- U. Y. Ogras, et al., 'Design and Management of Voltage-Frequency Island Partitioned Networks-on-Chip,' in IEEE Trans. VLSI, March 2009.
- P. Choudhary, D. Marculescu, 'Power Management of Voltage/Frequency Island-Based Systems Using Hardware Based Methods,' in IEEE Trans. on VLSI, March 2009
- T. Simunic, S. P. Boyd, P. Glynn, 'Managing power consumption in networks on chips,' in IEEE Trans. on VLSI, Jan. 2004.
- A. Alimonda, et al., 'Feedback-Based Approach to DVFS in Data-Flow Applications,' in IEEE Trans. on CAD of Integrated Circuits and Systems 28(11): 1691-1704 (2009)
- E. Beigne, et al., 'Dynamic voltage and frequency scaling architecture for units integration within a GALS NoC,' in Proc. Int. Symp. Netw. Chip, 2008, pp. 129–138.
- C.-L. Chou and R. Marculescu, 'Energy- and performance-aware incremental mapping for networks on chip with multiple voltage levels,' IEEE Trans. Comput.-Aided Design Integr. Circuits Syst., vol. 27, no. 10, pp. 1866–1879, Oct. 2008
- Q. Wu, et al, 'Formal Online Methods for Voltage/Frequency Control in Multiple Clock Domain Microprocessors', in Proc. ASPLOS 2004
- G. Semeraro, et al, 'Energy efficient processor design using multiple clock domains with dynamic voltage and frequency scaling,' in Proc. HPCA 2002
- U. Y. Ogras, et. al, 'NoC Prototyping Using FPGAs: Challenges and Promising Results in NoC Prototyping Using FPGAs,' in IEEE Micro, September/October 2007
- Clermidy et al, 'A 477mW NoC-Based Digital Baseband for MIMO 4G SDR,' IEEE ISSCC, February 2010
- P. Juang, et al. 'Coordinated, distributed, formal energy management of chip multiprocessors,' Proc. ISLPED 2005..

References (Part IV)

- Y. Abulafia and A. Kornfeld. Estimation of FMAX and ISB in microprocessors. IEEE Trans. on VLSI Systems, 13(10), Oct 2006.
- A. Bonnoit, S. Herbert, D. Marculescu and L. Pileggi. Integrating Dynamic Voltage/Frequency Scaling and Adaptive Body Biasing using Test-time Voltage Selection. In Proc. of IEEE/ACM ISLPED, Aug. 2009.
- K. Bowman, S. Duvall, and J. Meindl. Impact of die-to die and within-die parameter fluctuations on the maximum clock frequency distribution for gigascale integration. IEEE Journal of Solid-State Circuits, 37(2), Feb 2002.
- S. Garg, D. Marculescu. System-Level Mitigation of WID Leakage Variations using Body-Bias Islands. In Proc. ACM/IEEE CODES+ISSS, Atlanta, GA, October 2008.
- S. Garg, D. Marculescu, R. Marculescu and U. Ogras. Technology-driven Limits on DVFS Controllability of Multiple Voltage-Frequency Island Designs. In Proc. of IEEE/ACM Design Automation Conference (DAC), Jul. 2009.
- S. Herbert and D. Marculescu. Analysis of dynamic voltage/frequency scaling in chip-multiprocessors. In ISLPED '07: Proc. of the 2007 ISLPED, 2007.
- S. Herbert and D. Marculescu. Variation-Aware Dynamic Voltage/Frequency Scaling. In Proc. of the 15th HPCA, Feb. 2009.
- C. Isci, A. Buyuktosunoglu, C.-Y. Cher, P. Bose, and M. Martonosi. An analysis of efficient multi-core global power management policies: Maximizing performance for a given power budget. In MICRO '06, 2006.
- S.R. Sarangi, B. Greskamp, R. Teodorescu, J. Nakano, A. Tiwari and J. Torrellas. VARIUS: A Model of Process Variation and Resulting Timing Errors for Microarchitects. IEEE Transactions on Semiconductor Manufacturing (IEEE TSM), February 2008.
- R. Teodorescu and J. Torrellas. Variation-aware application scheduling and power management for chip multiprocessors. In ISCA'08: Proc. of the 35th ISCA, 2008.
- J.Tschanz, J.T. Cao, S.G. Narendra, R. Nair, D.A. Antoniadis, A.P. Chandrakasan, V. De. Adaptive Body Bias for Reducing Impacts of Die-to-Die and Within-Die Parameter Variations on Microprocessor Frequency and Leakage. IEEE Journal of Solid-State Circuits, Vol. 37, No. 11, Nov. 2002.
- W. Zhao and Y. Cao. New generation of predictive technology model for sub-45nm early design exploration. IEEE Trans. Electron Devices, vol. 53, no. 11, pp. 2816–2823, Nov. 2006.
- **This list of references is NOT exhaustive. There are many good contributions not mentioned here due to involuntary omissions or space limitations.**