Sthira: A Formal Approach to Minimize Voltage Guardbands under Variation in Networks-on-Chip for Energy Efficiency

Raghavendra Pradyumna Pothukuchi
Amin Ansari
Bhargava Gopireddy
Josep Torrellas

University of Illinois at Urbana-Champaign
Qualcomm
University of Illinois at Urbana-Champaign
University of Illinois at Urbana-Champaign

PACT, September 2017
Process Variations

• Variations in the transistor’s properties
  • Magnified by shrinking transistor dimensions and voltage

• Sources:
  • Random: Natural variations in the transistor atoms
  • Systematic: Manufacturing imprecisions

• Impact:
  • Variation in the minimum required voltage for correct operation

• Expensive to statically find the right voltage
  • Modern multicores have many complex components

• Designers use worst case voltage guardbands

Worst case voltage guardbands reduce energy efficiency in modern multicores

Process Variations in the NoC

- Network on Chip (NoC) spread over the chip
  - More susceptible for variations
- NoC accounts for 15 – 30 % of the chip energy
  - Energy inefficient guardbands have a big impact

How to operate each router at the right voltage to avoid errors and energy wastage?

Contributions

• **Sthira**: Reduce NoC voltage guardbands dynamically
  • Systematic approach using control theory
  • Reduce average energy consumption by 32%
  • Negligible performance impact

• **Sthira+**: Aggressive variant of Sthira
  • Uses a secondary network to further improve energy efficiency
  • Reduce average energy consumption by 36%
  • However, comes with additional network design cost
Reducing Voltage in an NoC Router

- Voltages lower than required can cause router errors
  - Timing errors: circuits cannot transfer values in the right time
- Impact of voltage on the probability of errors in a router

![Graph showing the relationship between router voltage and error rate with error-free, guardband, and nominal voltage regions.]
Timing Errors across All Routers

Each curve corresponds to a router in a 64-router NoC
Insights Guiding the Sthira Designs

• Desired operation is in the central region
  • Small number of data units (flits) suffering errors are retransmitted

• Precise voltage control is necessary
  • Probability of errors is exponentially related to voltage

• Ad hoc heuristics waste energy or induce many errors

Need control theoretic techniques that provide guarantees
Conceptual Sthira Design

Reference error rate → Formal Controller → Router voltage ($V_{dd}$) → Error rate i.e. $Number of errors/Number of flits$ in an interval

NoC

Error rate i.e. $Number of errors/Number of flits$ in an interval

Conceptual Sthira Design

Use the same controller
Maintain separate state

Routers have similar slopes (dynamics)

Reference error rate

Formal Controller

NoC
Adding Realistic Voltage Updates

Voltage Regulators (VR) update voltages router groups

VRs have a minimum step size (e.g., 10 mV)
Designing the Formal Controller

- PID (Proportional – Integral – Derivative) Controller
  - Accepts a reference error rate
  - Monitors the deviation of error rate from the reference
  - Changes router voltages at periodic intervals

New voltage = \( K_P \times \text{Deviation}_{\text{Current}} + K_I \times \text{Deviation}_{\text{Cumulative}} + K_D \times \text{Deviation}_{\text{Rate}} \)

- Proportional
- Integral
- Derivative

- Needs a model of the voltage – error rate relationship
  - Standard ‘S’ shaped functions
  - Taylor expansion

MATLAB aided design to set \( K_P, K_I, K_D \)

Ensure fast convergence, stability of decisions and small overshoots

Sthira+: Improving Sthira’s Efficiency

- Sthira retransmits flits that had errors on the main network
  - Possible that the flits suffer errors again
  - Voltages could be set to be slightly higher than necessary
- Sthira+ uses another network to retransmit flits that failed
  - Secondary network operates at error-free nominal voltage
  - Lightweight compared to the main network
  - Main network sees fewer errors
  + Allows the routers to reach the lowest possible voltage
    - Higher design cost
Secondary Network in Sthira+ 

- Balanced fat tree with routers at all levels
  - Link widths increase near the root
- Arranged to simplify the routing logic
  - $ID_{\text{left child}} < ID_{\text{parent}} < ID_{\text{right child}}$
  - Each subtree has contiguous IDs from $ID_{\text{subtree min}}$ to $ID_{\text{subtree max}}$

![Diagram of the Secondary Network in Sthira+](image-url)
Evaluation Infrastructure

- VARIUS NTV for modeling process variations
  - Inputs: synthesized router RTL, process variation parameters
  - Outputs: probability of timing errors in each path
- Cycle-level architecture simulator with NoC
  - Baseline NoC is a mesh
- MATLAB for PID controller design
- Multiprogrammed workloads
  - Commercial, scientific and engineering workloads
- Assumption
  - Minimum voltage regulator step is 10 mV
Architectures Evaluated

**Baseline**
Conventional NoC with none of our changes

**Tangle**
State-of-the-art, heuristic

**Tangle+**
Similar to Tangle, but router based

**Sthira**
PID controller managed voltage

**Sthira+**
Sthira, augmented with a secondary network

- **Baseline**
  - Conventional NoC with none of our changes
  - Voltage decreased periodically
  - On error, voltage increases for all routers on the flit’s path

- **Tangle**
  - State-of-the-art, heuristic
  - Voltage decreased periodically
  - On error, voltage is increased only for the erroneous router

- **Tangle+**
  - Similar to Tangle, but router based
  - Voltage decreased periodically
  - On error, voltage increases if error rate > reference (0.05%)

- **Sthira**
  - PID controller managed voltage
  - Voltage decreases periodically
  - On error, voltage is increased only for the erroneous router

- **Sthira+**
  - Sthira, augmented with a secondary network
  - Retransmission on the secondary network

---

Voltage Changes in a 64-router NoC

Each line corresponds to the voltage of a router over time.
Thick line shows the average voltage of all routers.

Sthira designs have smooth voltage changes and lower final voltages.

Results with Per-router Voltage Domains

Voltage Domain = Set of routers whose voltage is changed together

Energy savings (%)

- Tangle
- Tangle+
- Sthira
- Sthira+

Performance overhead (%)

- Tangle
- Tangle+
- Sthira
- Sthira+

Average error rate (%)

- Reference

Sthira designs save more energy and have low overhead.

Scaling the Number of NoC Routers

Sthira designs preserve energy benefits across NoC sizes

Secondary network utilization remains low even for large NoCs
Conclusions

• Process variations require large voltage guardbands in NoCs
• Sthira dynamically reduces NoC voltage while monitoring errors
  • Uses formal PID control
  • Saves 32% of NoC energy
  • Negligible performance overhead
• Sthira+ has a secondary network for retransmission
  • Saves 36% of NoC energy
  • However, added design cost makes it less competitive over Sthira
• Formal control enables Sthira to outperform state-of-the-art
  • Systematic approaches are key for future energy efficient systems
Backup
Additionally, in the paper

• Detailed router timing error modeling
• Controller design description
• Analysis of costs and design issues
• Multifaceted analysis of different architectures
• Design space exploration of Sthira design parameters
Impact of Technological Parameters

Smaller voltage steps enable smoother error rate control

Fewer routers per domain deliver bigger energy savings

4-Stage Controller Implementation

\[ E_{0} \]

\[ E_{n} \]

\[ E_{n+1} \]

\[ G_{I} \]

\[ G_{D} \]

\[ \sigma_{n}(i) \]

\[ \Delta E_{n}(i) \]

\[ \Delta V \]

\[ \text{Quantize} \]

\[ \text{Voltage Regulator} \]
Impact of Reference Error Rate
Secondary Network Utilization

![Bar chart showing secondary network utilization for various workloads and configurations. The chart includes workloads like mcf, leslie3d, GemsFD, xalancbmk, art, ocean, sap, sjas, swim, tpcw, and average. The chart is color-coded to distinguish between root and leaves.]
Maximum Error Rate

![Graph showing Maximum Error Rate for different voltage levels]
## Simulation Parameters

### Table I: Architecture and variation parameters. For the memory hierarchy, we give round-trip latencies from the core.

<table>
<thead>
<tr>
<th>Core Parameters</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Frequency</td>
<td>1 GHz</td>
</tr>
<tr>
<td>Fetch, issue, and commit</td>
<td>2 per cycle</td>
</tr>
<tr>
<td>ROB: 1st/2nd queue</td>
<td>64 entries: 16/16 entries</td>
</tr>
<tr>
<td>Issue queue: 1-fetch queue</td>
<td>64 entries: 32 entries</td>
</tr>
<tr>
<td>Branch (IR) predictor</td>
<td>Tournament (bimodal + 2-level)</td>
</tr>
<tr>
<td>BTB; history table</td>
<td>1024 entries, 2-way; 2048 entries</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Memory System Parameters</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1 data cache</td>
<td>32KiB, 2-way, 2 cyc. latency, 64B line</td>
</tr>
<tr>
<td>L1 instr. cache</td>
<td>32KB, 2-way, 2 cyc. latency, 64B line</td>
</tr>
<tr>
<td>L2 cache</td>
<td>32MB shared, 64-bank static addr</td>
</tr>
<tr>
<td>L2 cache bank</td>
<td>8-way, 64B line, 6 cyc. latency (local)</td>
</tr>
<tr>
<td>Main memory</td>
<td>260 cyc. latency, 4 mem. controllers</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Main Network-on-Chip Parameters</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Topology; routing</td>
<td>8x8 2D-mesh; X-Y routing, wormhole</td>
</tr>
<tr>
<td>Num. virtual channels</td>
<td>4 per physical channel</td>
</tr>
<tr>
<td>Buffer depth per virtual channel</td>
<td>4</td>
</tr>
<tr>
<td>Min. ( V_{dd} ) tuning step</td>
<td>10mV, 20 cycles latency</td>
</tr>
<tr>
<td>Retransmission buffer depth</td>
<td>8</td>
</tr>
<tr>
<td>Link width</td>
<td>128b</td>
</tr>
<tr>
<td>Nominal ( V_{dd} )</td>
<td>825mV (10% guardband)</td>
</tr>
<tr>
<td>Length of an epoch</td>
<td>50,000 cycles (min.)</td>
</tr>
<tr>
<td>Num. routers in ( V_{dd} ) domain</td>
<td>From 1 to 64; 1 is default</td>
</tr>
<tr>
<td>Penalty to Sthira/Sthira+ NoCs</td>
<td>10% power due to ( V_{dd} ) regulation</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Secondary Network Parameters</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Topology</td>
<td>AVL Tree</td>
</tr>
<tr>
<td>Routing</td>
<td>Least Common Ancestor routing</td>
</tr>
<tr>
<td>Link width</td>
<td>16b (lowest level), double every 2 levels</td>
</tr>
<tr>
<td>Number of buffers</td>
<td>One per link</td>
</tr>
<tr>
<td>Buffer depth</td>
<td>1</td>
</tr>
<tr>
<td>Nominal ( V_{dd} )</td>
<td>825mV</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Process Variation Parameters</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>( \text{Tech. node: } V_{dd} \text{ guardband} )</td>
<td>HoRM: 10%</td>
</tr>
<tr>
<td>( \text{Total } V_{th} \left( \sigma/\mu \right) )</td>
<td>12.5% (equal random &amp; systematic)</td>
</tr>
<tr>
<td>( \text{Total } L_{gff} \left( \sigma/\mu \right) )</td>
<td>6.25% (equal random &amp; systematic)</td>
</tr>
<tr>
<td>Correlation range ( \phi )</td>
<td>0.1 (for both ( V_{th} ) and ( L_{gff} ))</td>
</tr>
</tbody>
</table>