Tackling Voltage Emergencies in NoC through Timing Error Resilience

Rajesh Jayashankar Shridevi, Dean Michael Ancajas, Koushik Chakraborty, Sanghamitra Roy
USU BRIDGE LAB, Electrical and Computer Engineering
Utah State University
Logan, United States of America

Abstract—Aggressive technology scaling exacerbates the problem of voltage emergencies in emerging MPSoC systems. Network-on-Chips, the de-facto standard for connecting on-chip components in forthcoming devices play a central role in providing robust and reliable communication. In this work, we propose DrNoC (droop resilient network-on-chip)—two microarchitectural techniques to mitigate voltage emergency-induced timing errors in NoCs and preserve error-free communication throughout the network. DrNoC employs frequency downscaling and a pipeline error-recovery mechanism to reclaim corrupted flits in the router. Compared to the recently proposed NSFTR fault-tolerant technique, DrNoC offers a 27% improvement in energy-delay efficiency.

Keywords—network-on-chip, timing errors, voltage emergency

1. INTRODUCTION

Integrated circuits designed in current and future technology nodes are susceptible to timing errors, where the circuit delay exceeds an estimated threshold. While many factors—static and dynamic—lead to timing errors, voltage emergencies have emerged as one of the key causes. Consequently, many recent works have explored effective ways to mitigate voltage emergencies at the processor cores. However, voltage emergencies and timing errors in Network-on-Chips (NoCs)—the de facto standard for the on-chip communication in modern many-core systems—has received little attention.

Several technology trends are colluding to raise the importance of Voltage Emergencies on a NoC (VEN). First, technology scaling offers substantially more energy savings in computation than in communication. Consequently, NoCs consume a significant fraction of the chip power today (e.g., 36% [9]). Second, with the growing power footprint of the NoC, it draws a large current in its circuit components. Fluctuations in the current drawn by the NoC can lead to VENs, causing sporadic timing errors. Using a rigorous cross-layer infrastructure, we show that VENs impose a key challenge for a NoC design. In fact, mitigating VEN induced timing errors through voltage guardbands alone, can severely degrade the energy efficiency.

In the light of these trends, we explore efficient microarchitectural techniques to recover from timing errors in a NoC. Timing errors present unique challenges in a NoC router pipeline, compared to those in a processor pipeline [5]. For example, pipeline flush and recovery mechanisms do not have an equivalence in the realm of the NoC pipeline. To overcome this limitation, we propose two circuit-architectural techniques in the NoC architecture. To the best of our knowledge, this is the first work that aims to tackle timing errors in a NoC router microarchitecture. Existing related works can be broadly categorized into two classes: (a) voltage emergency mitigation; and (b) fault tolerant NoC. Many recent works attempt to mitigate voltage emergencies, all at the processor core levels (e.g., [7][11][14][15]). These include predicting voltage emergencies in a core pipeline [14], thread scheduling to tackle droop at the multicore level [12], among others. However, none of these tackle VENs. With modern chips dedicating a standalone power delivery network for the NoC [16], these works will be ineffective in combating VENs. On the other hand, a plethora of works in fault tolerant NoCs cover a range of faults: transient, intermittent, and permanent. Broadly their underlying principles are based on redundant NoC resources (e.g., links or routers) or routing multiple copies of a given packet through distinct routes. Unfortunately, duplicated links or routers may not offer timing error resilience. Routing techniques based on flooding may tackle VEN induced timing errors. One such recent technique is NSFTR that aims to avoid errors due to permanent and intermittent faults [13], including timing faults caused by VEN. However, the incurred overhead can be high as these techniques are not specifically targeting timing error resilience. We demonstrate that our proposed circuit-architectural error mitigation techniques offer compelling advantages over NSFTR.

We make the following contributions in this paper:

- We analyze the trends of VENs and timing errors with technology scaling in a modern system running real world traffic. We demonstrate the critical need for timing error resilience in modern NoCs (Section II).
- We present two novel schemes, collectively referred as DrNoC (droop resilient NoC architecture), to mitigate errors due to VEN (Section III). Enabled by DrNoC techniques, NoCs can operate at a tighter voltage guardband, where VEN induced timing errors are detected and tolerated with minimal performance loss.
- We perform a cross-layer analysis of our schemes by measuring the area, power, performance and Energy-Delay Product (EDP) (Section V). Compared to the recently proposed NSFTR [13], our best scheme offers 27% improvement in energy-delay efficiency.
In this section, we demonstrate the need to understand and alleviate the VEN induced timing errors in a NoC. After a brief background on VEN (Section II.A), we outline our circuit-architectural methodology for estimating VEN (Section II.B). Then, we present the trends of VEN, timing errors and their impact on energy efficiency (Section II.C) along with the need for energy efficient techniques for tolerating VEN-induced timing errors in NoCs.

A. VEN - A Key Cause of Timing Errors

There are two key factors behind voltage droop: resistive drop (IR) and inductive drop (dI/dt). Resistive drop (IR) is a function of the circuit topology and device parameters driven by a technology node. On the other hand, inductive drop is application dependent. Circuit utilization and traffic patterns dictate the current drawn in a circuit and its respective cycle by cycle alteration, thereby causing inductive drop.

B. Cross-Layer Methodology for VEN

Estimating VEN on many-core systems presents a methodological challenge, as voltage droop depends on many cross-cutting factors. We briefly outline our cross-layer methodology here, while presenting a detailed description in Section IV. First, to reflect the trends in modern many-core systems [16], we consider a dedicated power grid for the NoC. Second, we use the technology parameters predicted in ITRS [1] to assess the resistive network of the NoC power delivery model for 16nm and 45nm technology. Third, we perform cycle accurate simulation on the NoC, with the traffic driven by real world applications (PARSEC benchmarks). The router energy statistics are collected using the DSENT tool [17]. Finally, we feed the energy values of the pipeline stages along with the resistive network into a recent voltage droop estimation tool [4], and get cycle accurate VEN characteristics.

C. Scaling Trends of VEN and Timing Errors

Technology Scaling Impact: Fig. 1(a) presents the VEN analysis across two technology nodes 45nm and 16nm. The fault coverage (y-axis) is defined as the percentage of routers whose peak voltage droop lies within a corresponding voltage guardband (x-axis). To guarantee timing error-free operation of a circuit at a given frequency, it must be able to tolerate the peak voltage droop observed during its execution. For a distributed system such as a NoC, different NoC components (e.g., routers) may observe different levels of voltage droop. To capture this aspect, we consider a voltage guardband for the entire NoC, and estimate the corresponding fault coverage. Thus, different voltage guardbands lead to different fault coverages as seen in Fig. 1(a).

We make two pertinent observations from Fig. 1(a). First, we notice that the minimum voltage guardband required for 100% fault coverage, referred as $V_{FULL}$, is substantially more at the lower technology node (16nm). This is because, growing power density, interconnect resistivity and the reduction in supply voltage combine to exacerbate VEN with technology scaling. Second, we notice that there is a significant spread in the required voltage guardbands that offer partial coverage compared to $V_{FULL}$. For example, at 45nm technology node, a 20% fault coverage can be achieved with a 16% guardband, while its $V_{FULL}$ is 21%. For the 16nm node, the corresponding guardbands are 27% and 40%, respectively.

Energy Cost from Voltage Guardbanding: Fig. 1(b) shows the frequency of timing errors in the routers of a 8x8 NoC for the voltage guardbands ($V_{x1}$, $V_{x2}$, $V_{x3}$) set at (22%, 26%, and 30%) above the nominal supply voltage. For a change in voltage guardband of 8% (22% to 30%), the error rates vary by more than 45X (0.92% to 0.02%). This behavior is an aspect of modern synthesis tools aiming to balance delay in all paths. Fig. 1(c) presents the energy cost to operate the system at these voltage guardbands, as well as $V_{FULL}$ with respect to the nominal voltage at the 16nm node. We see that to guarantee 100% fault coverage, by operating at $V_{FULL}$, the NoC incurs a 45% loss in energy efficiency because a major portion of the NoC will be running at a much higher supply voltage than their individual requirements.

Timing errors from VEN can result in data corruption, flit redirection and other functional errors. Therefore, it is imperative to design energy efficient techniques to precisely detect and tolerate the VEN induced errors. Future systems like exascale computers are likely to employ massive NoCs. Fault free data communication in NoCs cannot be realized unless we can efficiently tackle timing errors.
III. DESIGN OVERVIEW

In this section, we discuss the design of DrNoC. We outline the design challenges of DrNoC (Section III.A). DrNoC comprises two key parts: (a) the error detection and confinement system (Section III.B), and (b) the recovery mechanisms used to recover corrupted flits (Section III.C). We explore two contrasting recovery mechanisms, and discuss the design aspects of voltage guardband selection in Section III.D

A. Design Challenges of DrNoC

NoC router is a pipelined microarchitecture like a microprocessor. However, the working principles of a NoC router pipeline are fundamentally different from a processor pipeline. Briefly, the design challenges are as under:

- **Error Confinement**: While detecting a timing error is identical in both the NoC and the processor, confining that error is totally different. Post error detection, the processor pipeline can squash all in-flight instructions to achieve confinement. However, in a router pipeline, the confinement strategy can change depending on the progress of the flit in the router pipeline.

- **Lack of Architected State**: The NoC router lacks an architected state to restore corrupt flits. To overcome this, we explore two contrasting schemes detailed in Section III.C. We augment the router microarchitecture to facilitate two possible error recovery locations: source node of the flit (RT) and in-situ node where the error is detected (SRE).

- **Managing Overhead**: To design a practical solution, it is critical to manage its circuit-architectural overhead in terms of area, power and performance. The distinct error profile and its corresponding mitigation overhead entails a new efficiency tradeoff exploration in a NoC.

B. Error Detection and Confinement

The timing errors caused by VEN are detected at the NoC router pipeline registers using shadow flip-flops [5]. These shadow flip-flops are fed with a delayed clock, allowing double sampling of the combinational logic output. A mismatch between the sampled data in the regular flip-flop and its corresponding shadow flip-flop indicates a timing error. Since the circuit paths are substantially more uniform in a router pipeline compared to a typical processor pipeline, inserting shadow flip-flops is fairly straightforward in a NoC router. Fig. 2 outlines the circuit-level modifications discussed above in a NoC router with 4 pipe stages: input buffer/route calculation, VC allocation, switch traversal and output buffer.

Once an error is detected, restoring error-free communication can only proceed after the error is confined within the router pipeline. A traditional NoC pipeline cannot stop a flit from transmission after it has reached the switch traversal stage. To account for this, we discuss two strategies for error confinement based on the error location:

- **Error before switch traversal** (i.e. stage 3 in Fig. 2): We block the flit before switch traversal by marking the VC as free, and increase the credit for that port. In a subsequent cycle, a new flit is allocated to the free VC entry, thereby overwriting the corrupted flit.

- **Error during switch traversal**: We add a poison bit to every output buffer entry. When an error is detected on a flit during switch traversal, poison bit is set. Subsequently, the link traversal is annulled for that flit in the next cycle, and the buffer and poison bit are cleared to reclaim that entry.

C. Recovery

We explore two variants for DrNoC based on the tradeoff in performance and complexity overhead. The first technique **Router Temporization (RT)** is a low-complexity source-based recovery technique that relies on flit retransmission (Section III.C.1)). The second technique **Selective Router Echo (SRE)** is an in-situ dynamic recovery mechanism with a low performance overhead (Section III.C.2)).

1) **Router Temporization (RT)**

DrNoC’s first recovery scheme uses a combination of flit retransmission and temporary frequency scaling to implement error-free communication in the presence of VEN. First, RT utilizes the acknowledgment system currently present in most NoC designs to guide a flit retransmission. Second, RT ensures that timing errors are eliminated during retransmission by halving the operating frequency while it is recovering its supply voltage.
Retransmission: The NoC router recovers flits by checking at the source if an acknowledgment (ACK) packet has been received to verify receipt of the data at the destination. After a set amount of time has passed without an ACK packet (i.e. timeout), the router assumes that the flit has been dropped. The router sends the same flit again until it receives an ACK.

Frequency Scaling: To prevent recurring corruption of flits, the routers throttle down the frequency once a threshold of dropped flits is exceeded. RT reverts to the original frequency by using an approach based on the exponential back-off algorithm. Initially, a 16-cycle recovery mode (i.e. frequency is halved) is used to check if there are no more VEN induced errors. At the end of the recovery mode, the frequency is reset back to the original level. Typically, voltage droops last for a short time span (few ns) [11]. Hence, our initial recovery mode suffices in most cases. However, if errors persist, RT will induce a longer recovery mode of 32-cycles. This step recurs until the errors stop. This technique allows recovery without using complex circuitry such as voltage sensors.

2) Selective Router Echo(SRE)
The second scheme in DrNoC architecture, Selective Router Echo, is a self-contained error recovery system embedded in the NoC router pipeline. Implementing a recovery mechanism in the NoC pipeline presents extra challenges as it does not maintain an architectured state. To mitigate this problem, we augment the router microarchitecture to mimic a processor pipeline. Fig. 2 shows the pipeline for the SRE-enabled router. We add extra virtual channels in the router, called Reserve VCs (RVCs) to keep a copy of all in-flight flits that have crossed the input buffer stage. Collectively, input buffers and RVCs contain all the flits present in various stages of the router pipeline. In the event of a VEN, the RVCs are used to replay erroneous flits in the pipeline. The specific steps in the recovery mechanism are:

- **Stall:** After a VEN induced timing error, the router is stalled. All incoming flits to the router are temporarily delayed by broadcasting zero credits to adjacent nodes. We use a short stalling period of 16 cycles in SRE, as most VENs do not last beyond a few nanoseconds [11].
- **Restart:** After stall completion, we restart the router and allow the remaining flits waiting to be routed in the input buffers are allowed to pass through. This process also clears the input buffers to prepare for the recovery of the flits from the RVCs.
- **Restore:** We restore entries from the RVC to the input buffers. Note that these are the flits that have crossed the input buffer stage. We are effectively restoring the router to an earlier state. Our technique exploits the ability of flits emitting from the router pipeline in an out-of-order fashion, unlike instructions in a processor pipeline.
- **Resume:** Finally, we resume normal operation of the router by lifting credit restrictions, and sending the flits in the input buffer to their target output buffers.

The hardware overhead of SRE comes from the shadow flip-flops, RVCs and the control logic required to facilitate retransmitting flits. SRE performs better than RT when the voltage drop is localized in a few routers only. In this case, the overhead of retransmitting all the flits in RT can possibly double their latency.

D. Voltage guardband Selection

Since DrNoC can mitigate timing errors due to VEN, it allows the NoC to operate at a lower voltage guardband, potentially improving energy efficiency. However, the eventual impact on energy efficiency depends on the chosen voltage guardband, timing errors from that guardband, and the runtime overhead associated with mitigating those errors. We explore three guardbands to study this interplay on energy efficiency ($\text{VG}_x1$, $\text{VG}_x2$ and $\text{VG}_x3$). Fig. 1(b) shows the frequency of timing errors in the NoC for these guardbands. The $\text{VG}_x3$ guardband has a low error rate of 0.02%. The $\text{VG}_x2$ guardband is at the other end of the spectrum with an error rate more than 40× the lowest one (0.92%), while the $\text{VG}_x1$ guardband has an error rate of 0.2%. Note that the 45× increase in the error rate is triggered by change in the guardband of only 8%. This landslide behavior of the error-rates beyond a particular voltage is also seen by other studies, and is attributed to modern synthesis tools aiming to balance delay in all paths [10].

While Fig. 1(b) may imply that the amount of errors experienced across different benchmarks are comparable, the distribution of these errors vary significantly. Fig. 3 shows the distribution of error frequencies, assuming $\text{VG}_x2$ guardband, for swaptions and fluidanimate. The error characteristics depend on both the router placement in the network, as well as, specific traffic patterns. The routers at the central portion of the network experience a higher error rate compared to peripheral routers due to a notably higher utilization. Our cross-layer evaluation carefully considers these traffic induced voltage drop distributions in analyzing the efficacy of DrNoC.

IV. CROSS-LAYER METHODOLOGY

In this section, we discuss our two stage cross-layer CAD methodology shown in Fig. 4. In the first stage (Section IV.A), we analyze the effect of power supply noise (PSN) in
the NoC. In the second stage (Section IV.B), we evaluate the efficacy of our microarchitectural techniques using three key metrics: area, power and performance.

A. Stage 1: PSN Analysis

The first stage in our methodology involves analyzing the impact of real-world benchmarks on the power supply network of a NoC. We combine a recently proposed power-supply analysis tool [4] with Netrace [8] to obtain cycle-by-cycle droop characteristics while running the benchmark programs on a 8×8 NoC. The accurate evaluation of power supply noise hinges on three critical data:

- **Router Energy Consumption**: We use the DSENT tool [17] to derive the router energy of the different pipeline stages of a NoC for the 16nm technology. The router microarchitectural parameters serve as inputs to DSENT.

- **Traffic**: The NoC PSN tool traditionally uses NoXim [6] to generate traffic files for synthetic workloads. We incorporate Netrace [8] simulation inside NoXim to drive real workload traffic patterns (PARSEC benchmarks).

- **RLC Components**: The resistive, inductive and capacitance values of the power delivery network are obtained for 16nm technology using the ASU PTM [2] based on technology parameters predicted by ITRS [1].

B. Stage 2: DrNoC Evaluation

We evaluate our proposed DrNoC on three metrics: area, power and performance overhead.

1) Area and Power

For the area and power overhead analysis, we implement our techniques in Verilog RTL on top of an open source NoC router model [3]. The NoC router model has four pipeline stages and is a Virtual-Channel router with 8 VCs per port and 5 input and output ports each. We add the shadow flip-flops, additional VCs and the control logic to implement the functionality of our schemes. We then use Synopsys Design Compiler to synthesize the final RTL using a TSMC 45nm library and scale the resulting area and power values to 16nm node.

2) Performance

The performance overheads are evaluated using cycle-accurate simulation of an 8×8 NoC running real world applications (PARSEC benchmarks). The NoC router in our performance model has been configured to match both the RTL model and the PSN model (Section IV.A). We model voltage droop events in the simulation by inducing errors based on the traffic characteristics experienced by the NoC and the data we obtain on the PSN analysis stage. We use Booksim 2.0 as our architectural simulator and Netrace as the engine to drive real-world traffic to Booksim. The packets are routed using Dimension-Order XY Routing that guarantees to avoid deadlock/livelock.

V. RESULTS

We show a comprehensive analysis of three metrics across the comparative schemes (Section V.A). Section V.B compares the energy efficiency, Section V.C compares the performance, and Section V.D compares the area/power overhead.

A. Comparative Schemes

We evaluate two different comparative schemes in our work. All schemes operate at the same frequency, and enable error-free communication, either by removing or mitigating timing errors due to VEN.

- **NSFR**: NSFTR is a fault tolerant routing scheme that combines probabilistic flooding and turn-model routing techniques. As two copies of a packet are sent over the network, this scheme can tolerate timing errors in one of the routes taken by a copy.

- **VG FULL**: This scheme models a NoC with the minimum guardband (VG FULL) required to remove all timing errors.
• **RT and SRE**: Our proposed detection and recovery schemes allow a guardband lower than $V_{\text{FULL}}$. We present results for three different guardbands, $V_{G\times1}$ (22%), $V_{G\times2}$ (26%) and $V_{G\times3}$ (30%) (Section III.D).

**B. Energy Delay Product (EDP) Comparison**

Fig. 6 shows the EDP, normalized to $V_{\text{FULL}}$, for the three chosen guardbands. On an average, RT and SRE show 9% and 12% EDP improvements over the $V_{\text{FULL}}$ under the $V_{G\times2}$ guardband, respectively. On the other hand, NSFTR offers substantial deterioration under $V_{G\times1}$ and $V_{G\times2}$ guardbands, while achieving energy-delay characteristics similar to $V_{\text{FULL}}$ at $V_{G\times3}$ guardband. To guarantee 100% delivery, tackling VEN through voltage guardbanding proves more efficient than NSFTR. While SRE is superior to RT on an average, for some benchmarks (bodytrack and vips), RT offers better energy efficiency under the $V_{G\times1}$ guardband. In these cases, reactive error avoidance by frequency scaling in RT offers a better efficiency compared to repeated attempts to send the flit at the same frequency level in SRE.

**C. Impact on Communication Latency**

Fig. 5 shows the performance overheads, measured by the average packet latency, over the $V_{\text{FULL}}$ for all the schemes across two guardbands ($V_{G\times1}$ and $V_{G\times2}$). We do not show overheads of SRE and RT at $V_{G\times3}$ guardband, as the performance overhead is found to be negligible at these error rates. But NSFTR has an average overhead of 7% for $V_{G\times3}$ guardband. In Fig. 5(b), SRE has an average overhead of 0.9%, while RT has an 8.16% overhead. At low error rates, downscaling of frequency by RT causes a queuing delay that can degrade the performance.

We observe interesting behaviors at relatively high error rates (Fig. 5(a): $V_{G\times1}$ guardband). As the number of VEN-induced errors increase, SRE’s overhead inflates more than that of RT for some benchmarks. SRE operates by stalling the upstream routers until all erroneous flits in the route are recovered. This will cause severe backpressure in the upstream routers waiting for availability of downstream routers. We call this phenomenon **serializati** **on**. We can see this in two of our benchmarks with high injection rates: bodytrack and vips. For bodytrack, SRE suffers a severe performance hit of 60%, while RT performs better incurring a 29% overhead. Hence, the serialization of flits combined with the back-pressure produces performance overhead greater than RT. In NSFTR, this situation is aggravated further due to congestion caused by the injection of replicated flits.

**D. Hardware Implementation Overhead**

We implement RT and SRE in the Verilog RTL on top of a NoC router [3]. Since SRE introduces a set of reserve VCs to keep track of in-flight flits, it incurs a larger overhead of 8.13% compared to RT (4.4%). Energy overhead incurred due to the additional logic of RT and SRE are accounted for in the energy-delay product results in Section V.B.

**Error Detection Overhead**: To estimate the overhead for error detection, we synthesize the NoC router Verilog RTL with a minimum delay constraint (25%). By enforcing minimum delay using buffers, we ensure that the hold time of shadow flip-flops is not violated by the data from the previous pipeline stage. Further, we account for the additional registers required to store the status of in-flight flits. The NoC router incurs a small overhead of 2.4% in area and 0.18% in power.

**VI. CONCLUSION**

Voltage emergencies are poised to create a serious challenge for NoCs. We perform a cross-layer analysis of VEN and study its impact from technology scaling and application behavior. Subsequently, we propose DrNoC—two techniques to detect and recover from VEN induced errors. Our comparative analysis shows compelling advantages of DrNoC over NSFTR to tolerate faults in a NoC.

**REFERENCES**


[12] Pasricha, S., & Zou, Y.”NS-FTR: a fault tolerant routing scheme for networks on chip with permanent and runtime intermittent faults”. In Design Automation Conference (ASP-DAC), 2011 16th Asia and South Pacific (pp. 443-448). IEEE.


