Approaching the Theoretical Limits of a Mesh NoC with a 16-Node Chip Prototype in 45nm SOI[*]

Sunghyun Park, Tushar Krishna, Chia-Hsin Owen Chen, Bhavya Daya,
Anantha P. Chandrakasan, Li-Shiuan Peh
Massachusetts Institute of Technology, Cambridge, MA

ABSTRACT

In this paper, we present a case study of our chip prototype of a 16-node 4x4 mesh NoC fabricated in 45nm SOI CMOS that aims to simultaneously optimize energy-latency-throughput for unicasts, multicasts and broadcasts. We first define and analyze the theoretical limits of a mesh NoC in latency, throughput and energy, then describe how we approach these limits through a combination of microarchitecture and circuit techniques. Our 1.1V 1GHz NoC chip achieves 1-cycle router-and-link latency at each hop and energy-efficient router-level multicast support, delivering 892Gb/s (87.1% of the theoretical bandwidth limit) at 531.4mW for a mixed traffic of unicasts and broadcasts. Through this fabrication, we derive insights that help guide our research, and we believe, will also be useful to the NoC and multicore research community.

Categories and Subject Descriptors

B.4 [Hardware]: Input/Output and Data Communications

General Terms

Design, Performance, Measurement

Keywords

Network-on-Chip, Theoretical Mesh Limits, Virtual Bypassing, Multicast Optimization, Low-Swing Signaling, Chip Prototype

1. INTRODUCTION

Moore's law scaling and diminishing performance returns of complex uniprocessor chips have led to the advent of multicore processors with increasing core counts. Their scalability relies highly on the on-chip communication fabric connecting the cores. An ideal communication fabric would incur only metal-wire delay and energy between the source and destination core. However, there is insufficient wiring for dedicated global point-to-point wires between all cores [8], and hence, packet-switched Networks-on-Chip (NoCs) with routers that multiplex wires across traffic flows are becoming the de-facto communication fabric in multicore chips [5].

These routers, however, can impose considerable overhead. Latency wise, each router can take several pipeline stages to perform the control decisions necessary to regulate the sharing of wires across multiple flows. Inefficiency in the control also frequently leads to poor link utilization on NoCs. Buffers queues have been used to improve flow control and link utilization, but come with overhead in energy consumption. Conventional wisdom is that NoC design involves trading off latency, bandwidth and energy.

In this paper, we describe our design of a NoC mesh chip that aims to simultaneously approach the theoretical latency, bandwidth and energy limits of a mesh, for all kinds of traffic (unicasts, multicasts and broadcasts). We first derive such theoretical limits of a mesh NoC for unicasts and broadcasts. This analysis closely guided us in our design which leverages virtual bypassing to approach the theoretical latency limit of a single cycle per hop for unicasts, multicasts and broadcasts. This, coupled with the speed benefits of low-swing signaling, enabled us to swiftly reuse buffers and approach theoretical throughput without trading off energy or latency. Finally, low-swing signaling applied to the datapath helps us towards the theoretical energy limit.

Contributions. In this paper, we make the following contributions:

- We present a mesh NoC chip prototype that shows 48-55% latency benefits, 2.1-2.2x throughput improvements and 31-38% energy savings as compared with an equivalent textbook baseline NoC described in Section 3.1. To the best of our knowledge, this is the first mesh NoC chip with multicast support.
- We define the theoretical mesh limits for unicasts and broadcasts, in terms of latency, throughput and energy. We also characterize several prior chip prototypes' performance relative to these limits.
- We present lessons learnt from our prototyping experience:
  - Virtual bypassing can enable 1GHz single-cycle router pipelines and 32% buffering energy savings with negligible area overhead (5% only). It comes at the expense of a 21% increased critical path, though this timing overhead can be masked in multicore processors where cores limit the clock frequency rather than routers. More critically, virtual bypassing does not address non-data-dependent power.
  - Low-swing signaling can substantially reduce datapath energy (3.2x less energy in 1mm links compared to a full-swing datapath) as well as realize high frequency single-cycle traversal per hop (5.4GHz with a 64bits 5×5 crossbar and 1mm links), but comes with increased process variation vulnerability and area overhead.
  - System-level NoC power modeling tools like ORION 2.0 [12] can be way off in absolute accuracy (∼5x of measured chip power) but maintain relative accuracy. RTL-based post-layout power simulations (post-layout) are much closer to measured power numbers, but post-layout timing simulations are still off.

The rest of the paper is organized as follows: Section 2 defines our baseline router, derives the theoretical limits of a mesh NoC, and characterizes prior chips performance relative to these limits. Section 3 describes our fabricated NoC prototype, while Section 4 details measurement results. Finally, we conclude in Section 5.

[*]The authors acknowledge the support of the Gigascale Systems Research Center and Interconnect Focus Center, research centers funded under the Focus Center Research Program (FCRP), a Semiconductor Research Corporation entity, and DARPA under Ubiquitous High-Performance Computing.

DAC 2012, June 3-7, 2012, San Francisco, California, USA.
Copyright 2012 ACM ACM 978-1-4503-1199-1/12/06 ...$10.00.

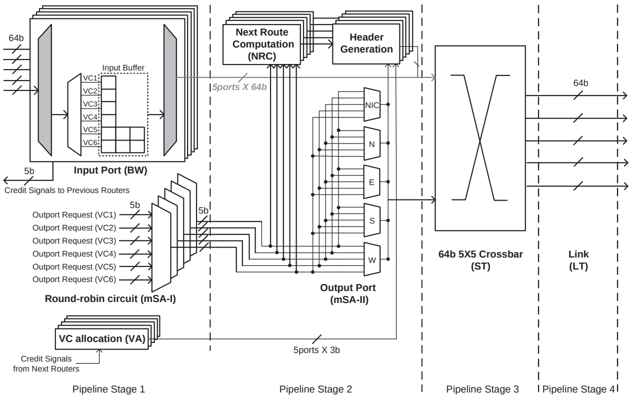


Figure 1: Baseline router microarchitecture.

2. BACKGROUND AND RELATED WORK

2.1 Baseline Mesh NoC

The mesh [6] is the most popular NoC topology for a general-purpose multicore processor, as it is scalable, is easy to layout, and offers path diversity [7, 10, 11, 21, 23]. Each core in a multicore processor communicates with other cores by sending and receiving messages through a network interface controller (NIC) that connects the core to a router (hence the network). Before a message is injected into the network, it is first segmented into packets that are then divided into fixed-length flits, short for flow-control units. A packet consists of a head flit that contains the destination address, body flits, and a tail flit that indicates the end of a packet. If the amount of information the packet carries is little, single-flit packets are also possible, *i.e.* where a flit is both the head and tail flit. Because only the head flit carries the destination information, all flits of a packet must follow the same route through the network.

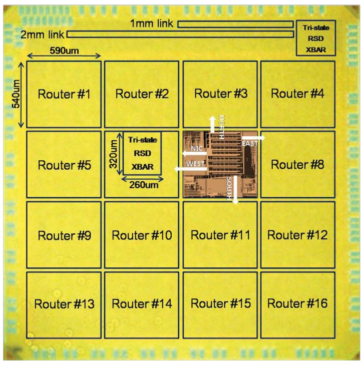
Figure 1 shows the microarchitecture of an input-buffered virtual channel router. Before an incoming flit is forwarded to the next router, it needs to go through several actions in order: buffer write (BW), route computation (NRC) (only for head flits), switch allocation (SA), virtual channel allocation (VA) (only for head flits), buffer read (BR), switch traversal (ST), and link traversal (LT). Out of all these actions, only ST and LT actually move the flits toward the destination. Thus, we consider all other actions as overhead. We will refer to this as the baseline router throughout the paper.

2.2 Latency, Throughput and Energy Limits

A mesh topology by itself imposes theoretical limits on latency, throughput and energy (*i.e.* minimum latency and energy, and maximum throughput). We derive these theoretical bounds of a $k \times k$ mesh NoC for two traffic types, unicast and broadcast traffic, as shown in Table 1. Specifically, each NIC injects flits into the network according to a Bernoulli process of rate R, to a random, uniformly distributed destination for unicasts, and from a random, uniformly distributed source to all nodes for broadcasts. All derived bounds are for a complete action: from initiation at the source NIC, till the flit is received at all destination NIC(s). More details on the derivation of the bounds is shown in Appendix A.

2.3 Related Work

There have been few chip prototypes with mesh NoCs as the communication fabric between processor cores or nodes, as listed in Table 2. Other NoCs, *e.g.* KAIST [2], Spidergon [4], Pleiades [24], are targeted for heterogeneous topologies and architectures, making it difficult to characterize them against the theoretical mesh limits. The prototypes range from full multicore processors to standalone NoCs. Of these, three chips were selected for comparison, that differ significantly with respect to targeted design goals and optimizations: Intel Teraflops which is the precursor of the Intel IA-32 NoC, Tilera TILE64 which is the successor of the MIT RAW, and SWIFT, a NoC with low-swing signaling. Each processor is described further in Appendix B.



Flit Size	64 bits
Request Packet Size	1 flit *(coherence requests and acknowledges)*
Response Packet Size	5 flits *(cache data)*
Router Microarchitecture	10 X 64b latches per port (6VCs over 2MCs)
Bypass Router-and-link Latency	1 cycle
Operating Frequency	1GHz
Power Supply Voltage	1.1V and 0.8V
Technology	45nm SOI CMOS

Figure 2: Die photo and overview of our fabricated 4×4 mesh NoC.

We calculated zero-load latency and channel load of these networks for both unicast-only and broadcast-only traffic. Zero-load latency is calculated by multiplying the average hop-count by the number of pipeline stages to traverse a hop, with serialization latency added on to model pipelining of all flits. We computed channel load based on an flit injection rate per core of R, following the methodology of [6]. The results are shown in the Table 2. We can see that our proposed router optimizes for broadcast (multicast) traffic and has much lower zero-load latency and channel load compared to all other networks.

TILE64 attempts to optimize for all three metrics, by utilizing independent simple networks for different message types. The simple router design, with no virtual channels, improves unicast zero-load latency but broadcast traffic latency is poor as its lack of multicast support forces the source NIC to duplicate $k^2 - 1$ copies of a broadcast flit and send a copy to every destination NIC. This increases channel load by $k^2 - 1$ times, causing contention at all routers along the shared route, making it impossible to meet the single-cycle per hop. TILE64's static partitioning of traffic across 5 networks may also lead to poor throughput when exercised with realistic uniform traffic. Similar effect on broadcast latency and channel load is observed for the Teraflops and SWIFT NoCs as none of these chip prototypes have multicast support. The SWIFT NoC with a single-cycle pipeline for unicasts performs better on zero-load latency, albeit at a lower operating frequency. The TeraFLOPS NoC has poor zero-load latency in terms of cycles due to a 5-stage pipeline, which is aggravated with broadcasts.

In the rest of this paper, we will describe how we designed a NoC chip specifically to approach the theoretical limits.

3. PROPOSED NOC CHIP DESIGN

This section describes the design of our chip prototype. Figure 2 shows our fabricated 16-node 4x4 NoC. The network is packet-switched, and all routers are connected to network interface circuits (NICs) to generate and receive packets. Each router has 5 I/O ports: North, East, South, West and NIC. Each input port has two message classes (MCs), request and response, to avoid message-level deadlocks in cache-coherent multicores.

3.1 Overview of Proposed Router Pipeline

Our design essentially evolves the original textbook router pipeline (Fig. 1) into a strawman 4-stage router pipeline tailored for multicasts so multicasts/broadcasts do not require multiple unicast packets to be injected. Next, we add features pushing latency towards the theoretical limit of a single cycle per hop, throughput towards the theoretical limit of maximum channel load, and energy towards the theoretical limit of just datapath traversal.

In the first pipeline stage, (1) flits entering the router are first buffered (BW). (2) Each input port chooses one output port request (mSA-I) out of the requests from all VCs at that input port with a round-robin logic that guarantees fair and starvation-free arbitration. Since multicast flits can request multiple output ports, the request is a 5b vector. (3) The next router VC is selected (VA) for each neighbor from a free VC queue at each output port. These

Table 1: Theoretical Limits of a $k \times k$ mesh NoC for unicast and broadcast traffic.

Metric	**Unicasts** **(one-to-one multicasts)**	**Broadcasts** **(one-to-all multicasts)**
Average Hop Count ($H_{average}$)	$2(k+1)/3$	$(3k-1)/2$, for k even $(k-1)(3k+1)/2k$, for k odd
Channel Load on each bisection link ($L_{bisection}$)	$k \times R/4$	$k^2 \times R/4$
Channel Load on each ejection link ($L_{ejection}$)	R	$k^2 \times R$
Theoretical Latency Limit given by $H_{average}$	$2(k+1)/3$	$(3k-1)/2$, for k even $(k-1)(3k+1)/2k$, for k odd
Theoretical Throughput Limit given by max$\{L_{bisection}, L_{ejection}\}$	R, for $k <= 4$ $k \times R/4$, for $k > 4$	$k^2 \times R$
Theoretical Energy Limit E_{xbar}: energy of crossbar traversal E_{link}: energy of link traversal	$2(k+1)/3 \times E_{xbar}$ + E_{xbar} + $2(k+1)/3 \times E_{link}$	$k^2 \times E_{xbar}$ + $(k^2-1) \times E_{link}$

Table 2: Comparison of mesh NoC chip prototypes

	Intel Teraflops [10]	Tilera TILE64 [23]	SWIFT [14]	This work	
	8×10, 65nm	5 8×8, 90nm	2×2, 90nm	4×4, 45nm SOI	
Clock frequency	5GHz	750MHz	225MHz	1GHz	
Power supply	1.1-1.2V	1.0V	1.2V	1.1V	
Power consumption	97W	15-22W	116.5mW	427.3mW	
Latency Metrics	Modeled as 8×8 networks				4×4 network
Delay per hop	1ns	1.3ns	8.9-17.8ns	1-3ns	
Zero-load latency (cycles)	30 (unicast) 120.5 (broadcast)	9 (unicast) 77.5 (broadcast)	12 (unicast) 86 (broadcast)	6 (unicast) 11.5 (broadcast)	3.3 (unicast) 5.5 (broadcast)
Throughput Metrics	Modeled as 8×8 networks				4×4 network
Channel width	39b	5×32b	64b	64b	64b
Bisection bandwidth	1560Gb/s	937.5Gb/s	112.5Gb/s	512Gb/s	256Gb/s
Channel load (R:injection rate/core)	64R (unicast) 4096R (broadcast)	64R (unicast) 4096R (broadcast)	64R (unicast) 4096R (broadcast)	64R (unicast) 64R (broadcast)	16R (unicast) 16R (broadcast)

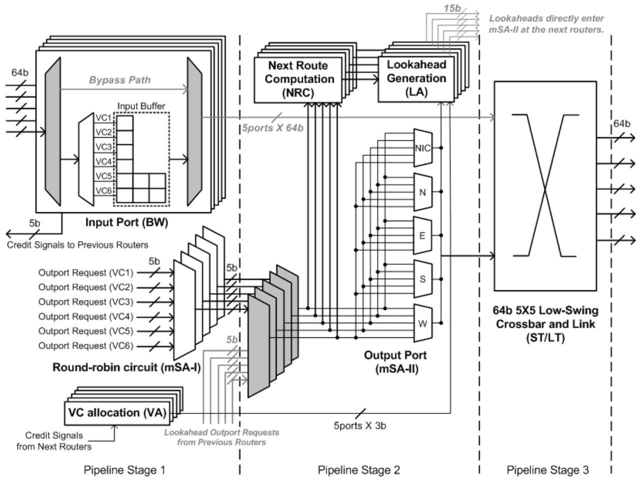


Figure 3: Proposed router microarchitecture and pipeline.

3 operations are executed in parallel without decreasing operating frequency as they are not dependent on each other. In the second stage, output port requests for the next routers are computed (NRC) for the winners of mSA-I, and concurrently, a matrix arbiter at each output port grants the crossbar ports to the input port requests (mSA-II). Multicast requests get granted multiple output ports. In the third stage flits physically traverse the crossbar (ST) and reach the next router through the link (LT) in the fourth stage.

At this point, our strawman router can simultaneously send a broadcast packet to all 16 nodes of a NoC. The baseline textbook router (Fig. 1), on the other hand, needs to generate multiple unicasts at each cycle to implement the broadcast packet and such unicasts takes 4 cycles per hop. The proposed design (Fig. 3) will completely be described through the following subsections.

3.2 Towards Theoretical Latency Limits

We push our strawman towards the limit by adding two key features: (1) virtual bypassing [15–17] to remove/hide delays due to buffering and arbitration and (2) low-swing circuits on the datapath to achieve single cycle ST+LT without lowering clock frequency.

Single-stage pipeline with lookaheads. In stage 2 of the strawman, we add and generate 15b lookahead signals from the results of NRC and mSA-II, and send them to the next router. The lookaheads try to pre-allocate the crossbar ahead of the actual flit, thus hiding mSA-II from the router delay. The lookahead takes priority over requests from buffered flits at the next router, and directly enters mSA-II. If the lookahead wins an output port, this pre-allocation allows the following flit to bypass the first two pipeline stages and go into the third stage directly, reducing the router pipeline depth from 4 to 2. Active pre-allocation by lookaheads enables incoming flits to bypass routers at all loads, in contrast to a naive approach of bypassing only at low-loads when the input queues are empty.

Single-cycle ST+LT with low-swing circuits. We apply a low-swing signaling technique, which can reduce the charging / discharging delay and dynamic energy when driving capacitive parasitics [20], to the highly-capacitive datapath. As will be described later in Section 3.4, the proposed low-swing circuits obtain higher current driving ability (or lower linear drive resistance) even at small V_{ds} than the reduced-swing signaling generated by simply lowering supply voltage, and hence, our low-swing datapath enables single-cycle ST+LT at higher clock frequency. Such single-cycle ST+LT can operate at up to 5.4GHz with 1mm 0.15um-width 0.30um-space links as demonstrated with measurement results in Section 4.3.

These two optimizations achieve a single-cycle-per-hop delay for unicasts and multicasts, exactly matching the theoretical latency limits. The caveat is that in case of contention for the same output port from multiple lookaheads, one of them will have to be buffered and then forced to go through the 3-stage pipeline. In addition, critical path delay is stretched, which will be analyzed in Section 4.

3.3 Towards Theoretical Throughput Limits

We take two steps towards the throughput limit for both unicasts and broadcasts (1) multicast support inside routers, and (2) single-cycle hop latency for fast buffer reuse.

Multicast support inside routers. We design a router that can replicate flits, allowing one multicast/broadcast flit to be sent from the source NIC, and get routed to all other routers in the network via a tree. This allows a broadcast flit to share bandwidth till it does not require an explicit forking into different directions. This dramatically reduces contention compared to the baseline design where multiple flits would have be sent as unicasts which are guaranteed to create contention at along the shared routes. We use a dimension ordered XY-tree in our design as it is deadlock free, and simplifies the routing algorithm. The ability to replicate flits in the router is implemented in the form of our broadcast-optimized

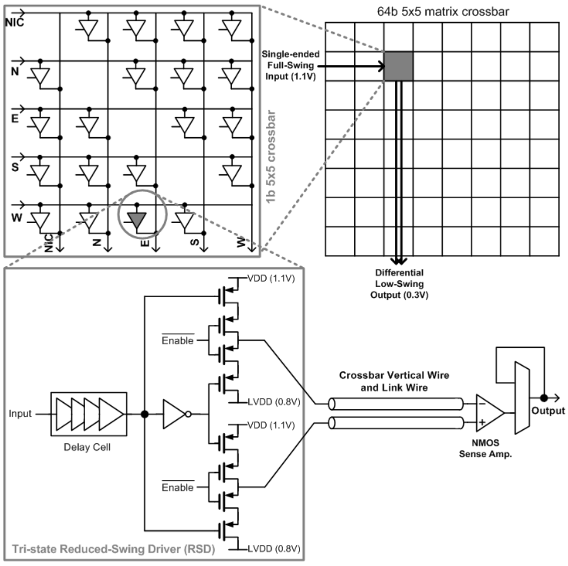

NIC
N
E
S
W
1b 5x5 crossbar
64b 5x5 matrix crossbar
Single-ended Full-Swing Input (1.1V)
Differential Low-Swing Output (0.3V)
VDD (1.1V)
Enable
LVDD (0.8V)
VDD (1.1V)
Enable
LVDD (0.8V)
Input
Delay Cell
Crossbar Vertical Wire and Link Wire
NMOS Sense Amp.
Output
Tri-state Reduced-Swing Driver (RSD)

Figure 4: Proposed low-swing crossbar and link circuits.

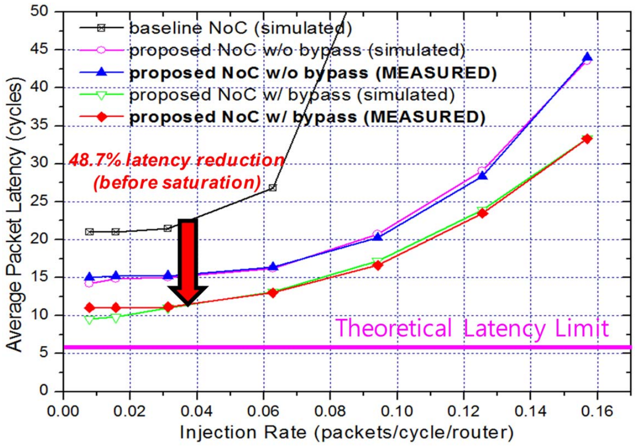
crossbar and mSA-II (switch allocation for multiple output ports).

Single-cycle-per-hop latency. The number of buffers/VCs required at every input port to sustain a particular throughput depends upon the buffer/VC turnaround time, *i.e.* the number of cycles for which the buffer/VC is occupied. This is where our optimizations for latency in Section 3.2 come in handy here since they reduce the pipeline depth, thus reducing buffer turnaround time, thereby increasing throughput given the same number of buffers. For our single-cycle pipeline, the turnaround time for buffers/VCs is 3: one cycle for ST+LT to the downstream router, one cycle for the free VC/buffer signal to return from the downtsream router (if the flit successfully bypassed), and one cycle for it to be processed and ready to be used for a new flit. We thus choose 4 VCs in the request message class, each 1-flit deep (since requests packets in our design are 1-flit wide) to satisfy VC turnaround time and sustain high throughput for broadcasts. We chose 2 VCs in our response message class, each 3-flit deep, for the 5-flit response packets. This number was chosen to be less than the turnaround time to shorten the critical path, and reduce the total buffers (which increase power consumption). We thus chose a total of 6 VCs per port, with a total of 10 buffers.

3.4 Towards Theoretical Energy Limits

Section 2 reveals a significant energy gap between the baseline router energy and the theoretical energy limit (which is just clocking and datapath energy, E_{xbar} and E_{link}). Such a gap is due to buffering energy (E_{buff}), arbitration logic energy (E_{arb}) and silicon leakage energy (E_{lkg}). Conventionally, these energy overheads are traded off against latency and throughput as follows: (1) Fewer buffers reduce E_{buff} and E_{lkg}, but stretch latency due to contention and lower throughput. (2) Simple routers like wormhole routers reduce E_{arb} and E_{lkg}, and increase operating frequency f, but these come at the expense of poorer latency and throughput.

Our proposed NoC first includes multicast support so even broadcasts and multicasts can approach the theoretical energy limit. Then, it incorporates two new features that permits different tradeoffs of latency, throughput and energy. First, our multicast virtual bypassing reduces E_{buff}, while improving both latency and throughput. The hidden cost lies in increased E_{arb} and decreased f. As shown in Section 4.1, the savings in E_{buff} outweigh the E_{arb} overheads, and operating frequency can still be in GHz. Second, our chip employs low-swing signaling to reduce dynamic energy in the datapath (E_{xbar} and E_{link}) which is unavoidable and part of the theoretical energy limit. Low-swing signaling provides an opportunity to break the conventional trade-offs that achieve dynamic energy savings at the cost of latency and throughput penalties; In fact, low-swing optimizes both energy and latency. Its downsides lie in its area overheads and reduced process variation immunity.



Baseline	Fabricated chip (bypass-disable)	Fabricated chip (bypass-enable)
557Gb/s, **54.4%** of theoretical limit	859Gb/s, **83.9%** of theoretical limit	892Gb/s, **87.1%** of theoretical limit

Maximum throughput comparison ***(received)***

Figure 5: Throughput-latency performance evaluation with mixed traffic at 1GHz.

Figure 4 shows the circuit implementation of the low-swing crossbar directly connected to links with tri-state reduced-swing drivers (RSDs). This crossbar enables low-swing signaling in the datapath (crossbar vertical wires and links). The tri-state RSD disconnects horizontal and vertical wires and only drives the corresponding vertical wire and link, thereby providing energy-efficient multicasting capability. With an additional supply voltage (LVDD), the 4-PMOS stacked RSD design generates more reliable low-swing signaling in the presence of wire capacitance and resistance variation than equalized interconnects [9, 13, 18] where low-swing signaling is obtained by wire channel attenuation. A delay cell aligns an input signal (which drives only a 1b crossbar) to an enable signal (which drives all of 64 1bit crossbars). It reduces mismatch between charging and discharging time, thus decreasing inter-symbol interference (ISI). The 64bits links are designed with 0.15um-width 0.30um-space fully shielded differential wires, to eliminate noise coupling of crosstalk effects and supply voltage variation.

4. EVALUATION

In this section, we first evaluate the measured energy-latency-throughput of our fabricated NoC against that of the baseline mesh and theoretical limits defined in Section 2. Armed with our chip measurements, we then delve into three specific case studies on virtual bypassing, low-swing signaling and power modeling and estimation to dissect our design choices.

4.1 Energy-Latency-Throughput Performance

We measured average packet latency of our NoC as a function of packet injection rate, with two different traffic patterns: mixed traffic (50% broadcast request, 25% unicast request and 25% unicast response messages) and broadcast-only traffic (100% broadcast request messages), at 1GHz operating frequency. For brevity, Figure 5 only shows the results for mixed traffic along with the baseline performance and theoretical mesh limits. Here, we chose a more aggressive baseline that has single-cycle ST+LT instead of separate ST and LT stages shown in Fig. 1. Since even the full-swing baseline can support single-cycle ST+LT at 1GHz, this baseline is a fairer model of an equivalent unicast full-swing NoC. Except for the the single-cycle ST+LT, the baseline used in this section is identical as that described in Section 2.1. The theoretical latency limits (cycles/packet) include two extra cycles for NIC-to-router and router-to-NIC traversals which are indispensable since traffic injects and ejects through the NICs. Theoretical throughput

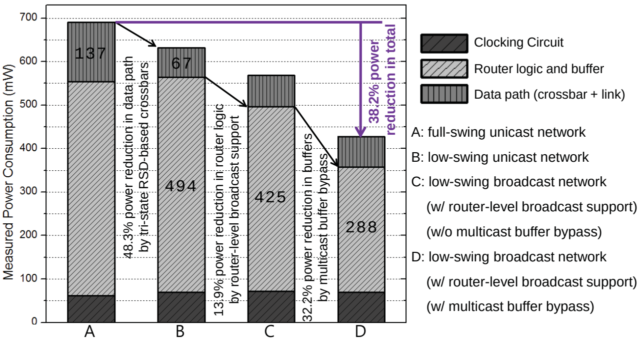


Figure 6: Measured power reduction at 653Gb/s at 1GHz.

limits are calculated based on received flits, then converted into Gb/s to factor in the 1GHz clock frequency and 64-bit flit size (16×64b×1/1GHz=1024Gb/s). Simulation results were obtained from pre-layout synthesis with sufficient simulation cycles (10^4 cycles) to make scan-chain warmup (128 cycles) negligible.

For latency, our design enables 48.7% (mixed traffic) and 55.1% (broadcast-only) reductions before the network saturates[1] as compared to the baseline. The low-load latency gap from the theoretical latency limit is 5.7 (6.3) cycles for mixed (broadcast) traffic, *i.e.* only 1.03 (1.14) cycles of contention latency per hop for mixed (broadcast) traffic. This can be further improved to 0.04 (0.05) cycles of contention latency per hop (obtained through RTL simulations) by removing the artifact in our chip whereby all NICS had identical pseudo-random generators that caused contention which lowers the amount of bypassing even at low injection rates.

Throughput wise, the fabricated NoC approaches the theoretical limits: 87% (mixed traffic) and 91% (broadcast-only) of the theoretical throughput limits. In addition, our NoC design has 2.1x (mixed traffic) and 2.2x (broadcast-only) higher saturation throughput than the baseline. In other words, the proposed NoC can obtain the same throughput as the baseline with fewer buffers or VCs. The throughput gap between the theoretical mesh and the fabricated chip is due to imperfect arbitration (like all prior chips, we use separable allocators, mSA-I and mSA-II, to lower complexity) and routing (XY routing can lead to imbalance in load).

Figure 6 shows the measured power reduction at 653Gb/s broadcast delivery at 1GHz at room temperature. The low-swing signaling enables 48.3% power reduction in the datapath. In addition, the single-cycle multicast capability and virtual bypassing result in 13.9% and 32.2% power reduction in router logics and buffers, respectively. Overall, our chip prototype achieves 38.2% power reduction compared to the baseline. To compare against the theoretical power limit, we performed a post-layout power simulation of a router in the middle of the mesh to further breakdown data-dependent power from non-data-dependent components like clocking. We then calculate the theoretical power limit to comprise just clocking and a full-swing datapath: 5.6mW/router, at close to zero-load injection rate (3/255). Compared to our NoC power consumption at the same low injection rate (13.2mW/router), our overhead comes largely from VC bookkeeping state (1.9mW/router) and buffers (2.0mW/router), whereas the allocators (0.7mW/router) and additional lookahead signals (0.2mW/router) contribute little additional power. The data-dependent power (*e.g.* buffers, allocators) is due to our identical PRBS generators at NICs that limited bypassing at low loads and can be removed by virtual bypassing, but the non-data-dependent power (*e.g.* VC state) will remain. Also, since our chip consumes nontrivial leakage power (76.7mW measured, 18% of overall chip power consumption at 653Gb/s), power gating will help to further close the gap, at the expense of a decrease in operating frequency.

Pre-layout simulations	
Baseline router design	549ns
Our virtual bypassed router design	593ns (1.08x overhead)
Post-layout simulations	
Baseline router design	658ns
Our virtual bypassed router design	793ns (1.21x overhead)
Measured critical path	
Our virtual bypassed router design	961ns (1/1.04GHz)

Table 3: Critical path analysis results.

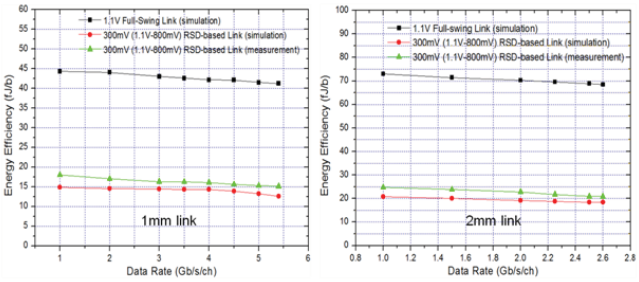


Figure 7: Measured energy efficiency of the proposed low-swing circuit on pseudo-random binary sequence data.

4.2 Virtual bypassing

Virtual bypassing of buffering to achieve single-cycle routers has been proposed in various forms [3, 15–17] in research papers. The aggressive folding of multiple pipeline stages into a single cycle naturally raises the question of whether that comes at the expense of router frequency f. While our chip is the first prototype to demonstrate a single-cycle virtual bypassed router at GHz frequency, it begs the question of how much f is affected. To quantify the timing overhead, we performed critical path analysis on pre- and post-layout netlists of the baseline and our design. Table 3 shows such estimates along with the actual measured timing.

The critical paths of both the baseline and the proposed router occur in the second pipeline stage where mSA-II is performed. The overhead of lookaheads lengthens the critical path by 8% in pre-layout simulations and 20% in post-layout simulations. It should be pointed out though that if the operating frequency is limited by the core rather than the NoC router, which is typically the case, this 20% critical path overhead can be hidden. In the Intel 48 core chip, nominal operation is 1GHz core and 2GHz router frequencies, allowing any network overhead to be masked [11].

Also notable is the fact that while the critical path of the post-layout simulation is 793ns, the maximum frequency of our chip prototype is 1.04GHz (*i.e.* the actual critical path is 961ns). This is mainly due to nonideal factors (*e.g.* a contaminated clock, supply voltage fluctuation, unexpected temperature variations, and *etc.*) whose effects cannot be exactly predicted in design phase.

4.3 Low-Swing Signaling

Low-swing signaling has demonstrated substantial energy gains in domains such as off-chip interconnects and SRAMs. However, in NoCs, there are few chip prototypes employing low-swing signaling [2, 14]. So a deep understanding of its trade-offs and its applicability to NoCs can be useful. To investigate such effects with longer links (necessary in a multicore processor as cores are much larger than routers), and at higher data rates than the network clock frequency (which is limited by synthesized router logic), an identical low-swing crossbar with longer link wires (1mm and 2mm) is separately implemented as shown in Figure 2.

Energy savings and 1-cycle ST+LT. The measured energy efficiency (Fig. 7) shows that the 300mV-swing tri-state RSD consumes up to 3.2x less energy as compared to a equivalent full-swing repeater. Experimental results also demonstrates that the tri-state RSD-based crossbar supports single-cycle ST+LT at up to 5.4GHz and 2.6GHz clock frequency with 1mm and 2mm links, respectively. The tri-state RSDs enables a reduction in the total amount of charge and delay required for data transitions, thereby resulting in these energy and latency benefits.

[1] To enable precise comparisons, we define the saturation point as the injection rate at which NoC latency reaches 3 times the average no-load latency; most multi-threaded applications run within this range.

Synthesized full-swing crossbar	26,840um^2
Proposed low-swing crossbar	83,200um^2 (3.1x overhead)
Router with the full-swing crossbar	227,230um^2
Router with the low-swing crossbar	318,600um^2 (1.4x overhead)

Table 4: Area comparison with full-swing signaling.

Area overheads. Table 4 shows the area overhead of our 5×5 64bits low-swing crossbar against an equivalent full-swing crossbar. The low-swing crossbar has a high area overhead (3.1x) compared to a synthesized full-swing crossbar, as the proposed RSDs employ differential signaling while the full-swing crossbar uses single-ended signaling. In addition, since our low-swing crossbar was carefully laid out due to noise coupling issues, such restricted placement and wiring of tri-state RSDs exacerbate the area overhead. However, at the router level, the relative area overhead goes down to 1.4x, and naturally, it will again diminish when compared against an entire tile with a core, cache and router.

Process variation effects. The critical drawback of low-swing signaling is reduced noise margin. In our circuit, the primary noise source is a sense amplifier offset caused by process variation. While low-swing signaling enables more dynamic energy savings as voltage swing decreases, the process variation effect worsens. Based on 1000-run Monte-Carlo Spice simulations, we chose 300mV-swing for above 3-σ reliability, but the voltage swing can be further decreased by offset compensation circuit techniques [1, 19, 22] at the cost of design complexity. Appendix C delves into further evaluation of our low-swing datapath.

4.4 Power Modeling and Estimation

Architectural power models such as ORION have been extensively adopted by researchers for early-stage evaluation of research ideas, while RTL-based energy estimates have also been widely used. With our chip, we can now study the gap between silicon-proven energy and different levels of energy modeling.

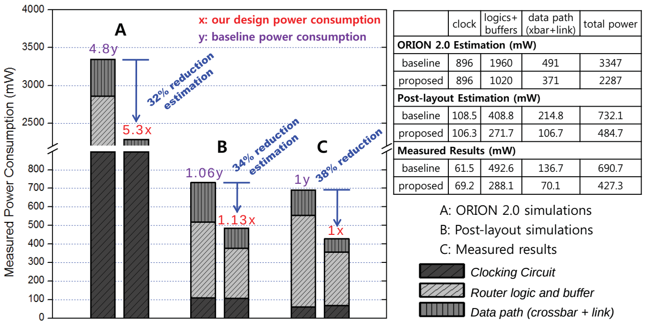
We compare our chip power measurements with two power estimates obtained from ORION 2.0 and post-layout netlists. The experiments (or simulations) are conducted with 1.1V supply voltage, 1GHz clock frequency, 653Gb/s throughput at room temperature. Figure 8 summarizes the results.

ORION 2.0 substantially over-estimates power (4.8-5.3x of measured chip power), but its estimate of relative power reduction between the baseline and our design (32% reduction) is not far from the measurements (38% reduction). This is because the transistor sizes assumed in ORION are much larger than the actual sizes in the chip. Thus, while ORION can be used for comparison of various system-level optimizations or early-stage design space exploration, its estimates should not be the basis of absolute power budgets.

On the other hand, the post-layout simulation gives us fairly accurate power estimates (6-13% deviation from measurements). Specifically, it slightly under-estimates the power of buffers and arbitration logic but over-estimates clocking and datapath power. Relative power reduction (34%) also matches well with measurements (38%). However, such accurate estimates come at the cost of tremendous simulation time overheads (several days for an entire NoC simulation) because the post-layout simulation calculates its estimates at the transistor-level along with parasitic effects. Moreover, since the post-layout estimation requires complete extracted netlists, it is difficult to apply to early-stage NoC evaluation.

5. CONCLUSION

This chip prototype offered us insights that may help guide future research. While virtual bypassing can effectively skip buffering and arbitration energy, there is a need for architectural techniques that tackle the non-data-dependent energy components as well without trading off latency and throughput. Similarly, though sophisticated off-chip signaling techniques have been shown to deliver substantial energy savings when applied to NoC interconnects, circuit or system-level solutions to their increased vulnerability to process variations need to be developed to ensure viability in future technology nodes. Finally, accurate timing and power models for early-stage NoC design are still sorely needed.



	clock	logics+ buffers	data path (xbar+link)	total power
ORION 2.0 Estimation (mW)				
baseline	896	1960	491	3347
proposed	896	1020	371	2287
Post-layout Estimation (mW)				
baseline	108.5	408.8	214.8	732.1
proposed	106.3	271.7	106.7	484.7
Measured Results (mW)				
baseline	61.5	492.6	136.7	690.7
proposed	69.2	288.1	70.1	427.3

Figure 8: Comparison of power estimates with measurements.

6. REFERENCES

[1] I. Arsovski and R. Wistort. Self-referenced sense amplifier for across-chipvariation immune sensing in high-performance content-addressable memories. In *IEEE Custom Integrated Circuits Conf.*, pages 453–456, 2006.

[2] S. Bell et al. A 118.4 gb/s multi-casting network-on-chip with hierarchical star-ring combined topology for real-time object recognition. *IEEE Journal of Solid-State Circuits*, 45:1399–1409, 2010.

[3] C.-H. O. Chen et al. Physical vs. virtual express topologies with low-swing links for future many-core nocs. In *Int'l Symp. on Networks-on-Chip*, May 2010.

[4] M. Coppola et al. Spidergon: a novel on-chip communication network. In *Int'l Symp. on System-on-Chip*, page 15, 2004.

[5] W. J. Dally and B. Towles. Route packets not wires: On-chip interconnection networks. In *DAC*, June 2001.

[6] W. J. Dally and B. Towles. *Principles and Practices of Interconnection Networks*. Morgan Kaufmann Publishers, 2004.

[7] P. Gratz et al. On-chip interconnection networks of the trips chip. *IEEE Micro*, 27(5):41–50, 2007.

[8] S. Heo and K. Asanovic. Replacing global wires with an on-chip network: A power analysis. In *Int'l Symp. on Low Power Elect. and Design*, pages 369–374, 2005.

[9] R. Ho et al. High-speed and low-energy capacitive-driven on-chip wires. In *Int'l Solid-State Circuits Conf.*, pages 412–413, 2007.

[10] Y. Hoskote et al. A 5-ghz mesh interconect for a teraflops processor. *IEEE Micro*, 27(5):51–61, 2007.

[11] J. Howard et al. A 48-core ia-32 message-passing processor with dvfs in 45nm cmos. In *Int'l Solid-State Circuits Conf.*, pages 108–109, 2010.

[12] A. Kahng et al. Orion 2.0: A fast and accurate noc power and area model for early-stage design space exploration. In *Proc. Design, Automation and Test in Europe*, pages 423–428, 2009.

[13] B. Kim and V. Stojanovic. A 4gb/s/ch 356fj/b 10mm equalized on-chip interconnect with nonlinear charge-injecting transmit filter and transimpedance receiver in 90nm cmos. In *Int'l Solid-State Circuits Conf.*, pages 66–67, 2009.

[14] T. Krishna et al. Swift: A swing-reduced interconnect for a token-based network-on-chip in 90nm cmos. In *Int'l Conf. on Computer Design*, pages 439–446, 2010.

[15] T. Krishna et al. Towards the ideal on-chip fabric for 1-to-many and many-to-1 communication. In *MICRO*, Dec 2011.

[16] A. Kumar et al. Express virtual channels: Towards the ideal interconnection fabric. In *Int'l Symp. on Computer Architecture*, June 2007.

[17] A. Kumar et al. Token flow control. In *MICRO*, Nov 2008.

[18] E. Mensink et al. A 0.28pj/b 2gb/s/ch transceiver in 90nm cmos for 10mm on-chip interconnects. In *Int'l Solid-State Circuits Conf.*, pages 314–315, 2000.

[19] M. Qazi et al. A 512kb 8t sram macro operating down to 0.57v with an ac-coupled sense amplifier and embedded data-retention-voltage sensor in 45nm soi cmos. In *Int'l Solid-State Circuits Conf.*, pages 350–351, 2010.

[20] J. M. Rabaey et al. *Digital Integrated Circuits: A design perspective*. Prentice Hall, 2nd Edition, 1998.

[21] M. B. Taylor et al. The raw microprocessor: A computational fabric for software circuits and general-purpose programs. *IEEE Micro*, 22(2):25–35, 2002.

[22] N. Verma and A. P. Chandrakasan. A high-density 45nm sram using small-signal non-strobed regenerative sensing. In *Int'l Solid-State Circuits Conf.*, pages 380–381, 2008.

[23] D. Wentzlaff et al. On-chip interconnection architecture of the tile processor. *IEEE Micro*, 27(5):15–31, 2007.

[24] H. Zhang et al. A 1 v heterogeneous reconfigurable processor ic for baseband wireless applications. In *Int'l Solid-State Circuits Conf.*, pages 68–69, 2000.

APPENDIX

A. DERIVATION OF THEORETICAL MESH LIMITS

First, we detail assumptions made in our analysis of the theoretical latency, energy and throughput limits, and explain our derivation process.

Assumptions:

1. Perfect routing: A router would route all packets with minimal hop-counts, balancing injected packets (termed channel load in our analysis) across multiple routes perfectly, thus keeping the load on all links optimally balanced.
2. Perfect flow control: A router maintains maximum utilization of the links; in other words, a link is never left idle when there is traffic routed across it.
3. Perfect router microarchitecture: All flits only incur the delay and energy of the datapath (ST and LT), that is, the router arbitrates between competing flits, performs crossbar and link traversal all in a single cycle and do not expend extraneous energy for buffering and control.

Assumption (1) and (2) are conventionally assumed in theoretical analysis of NoCs [6] while we further add assumption (3) as that is the minimum energy-delay per hop with synchronous NoCs.

Based on these assumptions, we derived the theoretical limits for unicast and broadcast traffic.

For unicasts, we analyze the theoretical limits for latency and throughput using the same technique as in [6]. We then derive the energy limit by multiplying hop count with crossbar and link energy costs.

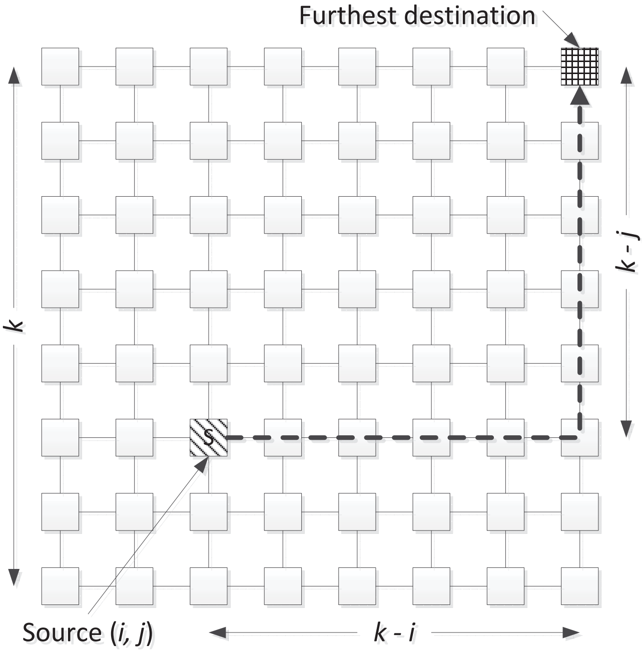


Figure 9: Latency calculation for broadcast traffic.

For broadcast traffic, to the best of our knowledge, no prior theoretical analysis exists. Here, we define the time till a flit is received by all destination NICs as equivalent to when this flit is received by the furthest NIC relative to the source NIC (Fig. 9). Hence, we derived the theoretical latency limit for received packets by averaging the hop delay from each source NIC to its furthest destination NIC. We obtained the theoretical throughput limit by analyzing the channel load across the ejection links and bisection links [6], and observed that the maximum throughput for broadcast traffic is limited by the ejection links. This differs from unicast traffic where throughput is always limited by the bisection links. As for the theoretical energy limit, intuitively, due to the nature of broadcasting, a broadcast flit needs to visit all k^2 routers in the network and traverse k^2 crossbars/links connecting them. Therefore, the energy limit grows quadratically with the number of routers in the network.

B. BACKGROUND ON PRIOR MESH CHIP PROTOTYPES

Here, we describe in detail these three other chips and corresponding NoC architecture.

Tilera TILE64 is a multiprocessor consisting of 64 tiles interconnected by five 2D mesh networks, where each tile contains a CPU, cache and a router, fabricated on the TSMC 90nm process and running at a speed of 700 to 866 MHz [23]. Four of the five networks are dynamically routed, each servicing a different type of traffic: user dynamic network (UDN) for user-level messages, I/O dynamic network (IDN) for I/O traffic, memory dynamic network (MDN) for traffic to/from the memory controllers, and tile dynamic network (TDN) for cache-to-cache transfers. The dynamic networks are packetized, wormhole routed, with a one cycle pipeline for straight-through traffic and two cycles for turning traffic. The static network is software scheduled, and has a single-cycle pipeline.

Intel Teraflops had a more complex NoC architecture, but the cores are much simpler than a standard RISC processor. Since simpler cores are more area- and energy-efficient than larger ones, more functional units can be supported within a single chip's area and power budget. Teraflops is a demonstration of the possibility of including an on chip interconnect, operating at 5 GHz, and achieving performance in excess of teraflops while maintaining a power usage of less than 100W [10]. Teraflops NoC has a five-port, two-lane, five-pipeline-stage router with a double pumped crossbar used to interconnect the tiles in a 2D mesh network. Each input port is connected to two 16 entry deep FIFO buffers, one for each lane. A single crossbar for both lanes is double pumped in the fourth pipeline stage using dual-edge triggered flip-flops, allowing the switch to transfer data at both edges of the clock signal.

SWIFT is a 2x2 standalone NoC research chip demonstrating the practicality of implementing token flow control [17] and low voltage swing crossbars and links. The bufferless traversal of flits through a reduce-swing datapath is demonstrated to perform at 400 MHz and obtain latency and power reductions of approximately 40 percent each [14]. The token flow control microarchitecture preallocates buffers and links in the network by using tokens. Many flits are then able to bypass buffering, improving link utilization and reducing the buffer turnaround time. Dual voltage supply differential reduced-swing drivers and sense-amplifier receivers sustain the low-swing signaling necessary to reduce the dynamic power consumption.

C. LOW-SWING CIRCUIT EVALUATION

Here, we present additional measured and simulated results of our low-swing signaling circuits.

Figure 10 shows energy efficiency and link failure probability of the 1mm 5Gb/s tri-state RSD as a function of voltage swing level. The normalized probability was calculated from 1000 Monte-Carlo Spice simulations. These results explicitly reveal the low-swing signaling energy gain trade-off against process variation vulnerability, as discussed earlier in Section 4.3.

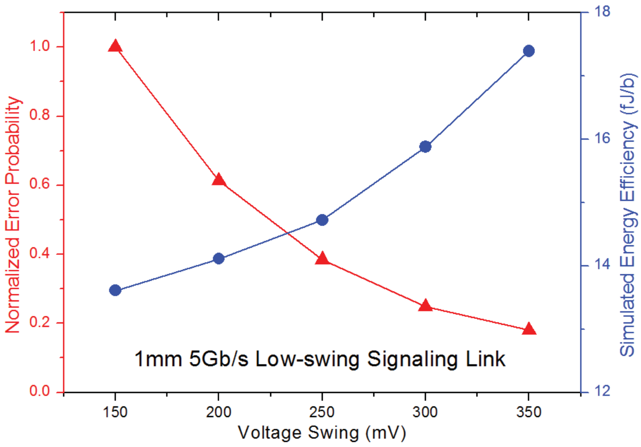


Figure 10: Low-swing signaling trade-off between reliability and energy efficiency.

We also measured power consumption of the 1b 5×5 tri-state RSD-based crossbar connected with 1mm link wires, with various multicast counts: a unicast, 2-multicast, 3-multicast and broadcast. Figure 11 show such results. As described in Section 3.4, the proposed low-swing crossbar drives only the corresponding vertical wires and link wires according to the multicast counts, and hence, it can provide energy-efficient multicasting capability (*i.e.* linearly increasing power consumption as a function of multicast counts).

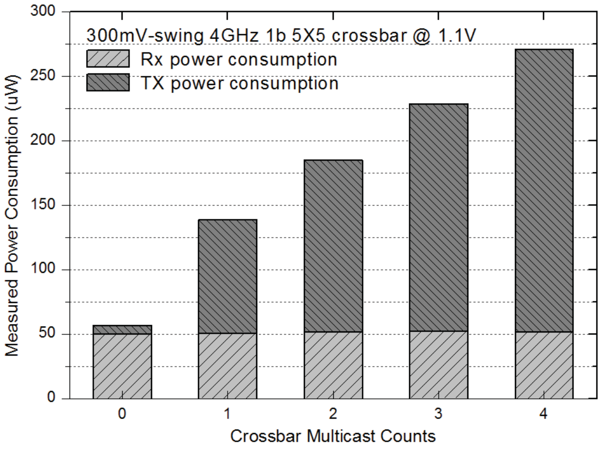


Figure 11: Measured dynamic power of the tri-state RSD-based crossbar.

We present another interesting trade-off between repeated and directly-transmitted (*i.e.* repeaterless) low-swing signaling. Figure 12 shows the 2.5Gb/s simulated vertical eye values with wire resistance variation at two 2mm-LT configurations: 1mm-repeated tri-state RSD and 2mm-repeaterless tri-state RSD. The results show that the 1mm-repeated low-swing link has a larger noise margin but takes an additional cycle and 28% more energy than the 2mm-repeaterless low-swing link.

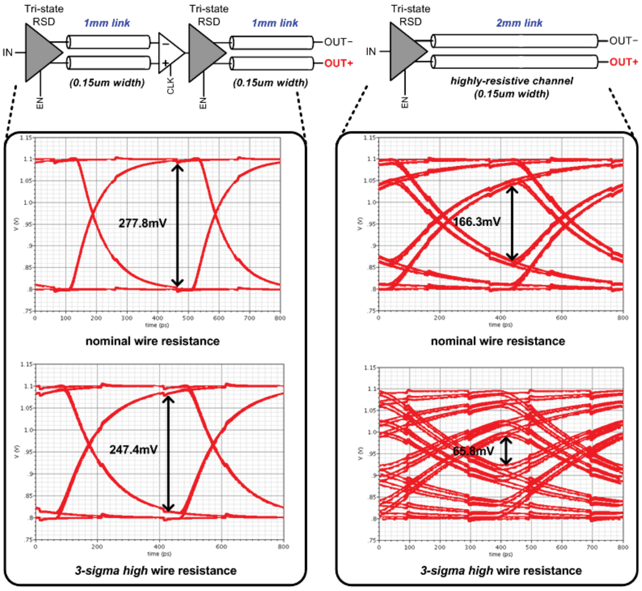
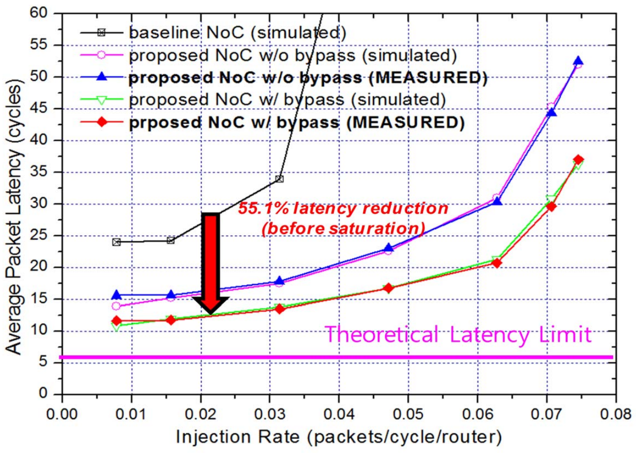


Figure 12: Reduced noise margin comparison of repeated and directly-transmitted low-swing signaling for 2mm-LT.

D. NETWORK PERFORMANCE WITH BROADCAST-ONLY TRAFFIC

As a case study, we evaluated the broadcast-only traffic performance of the fabricated NoC. Figure 13 shows the measured network performance and comparison with the simulated baseline performance. Compared to the mixed traffic performance (Fig. 5), the proposed NoC achieves more latency reduction and throughput improvement. In general, performance benefits of the proposed NoC get larger as network traffic becomes more broadcast-intensive (*i.e.* cache coherence protocols incorporate more broadcast messages with increasing core counts).



Baseline	Fabricated chip (bypass-disable)	Fabricated chip (bypass-enable)
528Gb/s, **51.6%** of theoretical limit	836Gb/s, **81.7%** of theoretical limit	932Gb/s, **91.1%** of theoretical limit

Maximum throughput comparison ***(received)***

Figure 13: Throughput-latency performance evaluation with broadcast-only traffic at 1GHz.