CMP Network-on-Chip Overlaid With Multi-Band RF-Interconnect

M. Frank Chang[‡] Jason Cong Adam Kaplan Mishali Naik Glenn Reinman Eran Socher[‡]
Sai-Wang Tam[‡]

[‡] *UCLA Electrical Engineering Department*
{mfchang,socher, roccotam}@ee.ucla.edu

UCLA Computer Science Department
{cong,kaplan,mishali,reinman}@cs.ucla.edu

Abstract

In this paper, we explore the use of multi-band radio frequency interconnect (or RF-I) with signal propagation at the speed of light to provide shortcuts in a many core network-on-chip (NoC) mesh topology. We investigate the costs associated with this technology, and examine the latency and bandwidth benefits that it can provide. Assuming a 400mm^2 die, we demonstrate that in exchange for 0.13% of area overhead on the active layer, RF-I can provide an average 13% (max 18%) boost in application performance, corresponding to an average 22% (max 24%) reduction in packet latency. We observe that RF access points may become traffic bottlenecks when many packets try to use the RF at once, and conclude by proposing strategies that adapt RF-I utilization at runtime to actively combat this congestion.

1. Introduction

The age of nanometer design has brought power and thermal considerations into sharp focus, making them high-priority architectural design metrics. Additionally, long wires (such as those that communicate dependency information) have become problematic, as their delay and the power consumption of repeaters is increasing relative to that of the transistors that drive them [9]. Thus, high-frequency single-core processors have become less attractive in the nanometer age, as their performance gains are achieved at a tremendous energy expense. Processor manufacturers are increasingly relying on Chip Multi-processor (CMP) designs, where silicon resources are partitioned among a number of processor cores. These cores can be connected together with a network on-chip (NoC) interconnect that can also include the shared on-chip (but off-core) cache hierarchy.

In order to scale future CMPs to 100's or even 1000's of cores, sophisticated interconnect topologies will be essential in enabling low-latency application communication and efficient cache utilization. We find that RF interconnects have tremendous promise in providing higher bandwidth between such a large number of interacting components, as well as reducing the number of cycles required for cross-chip communication, via signal propagation at the speed of light. However, RF bandwidth comes at an area cost, and cannot completely replace conventional RC wired interconnect. Therefore we propose a two-layer hybrid NoC scheme called MORFIC (Mesh Overlaid with RF InterConnect), where the RC wires are analogous to city streets accommodating local traffic, and the RF is like a superhighway, connecting distant points on the chip. We consider the circuit challenges remaining in bringing RF technology into CMOS design. The contributions of this work are as follows:

- Using real-world designs and ITRS projections as well as a physical implementation in 90 nm IBM process, we investigate the costs associated with on-chip RF interconnect, and demonstrate a physical roadmap for this promising technology (Section 2).
- We present the MORFIC architecture (Section 3), and discuss the architectural decisions a designer must make when implementing a two-layer NoC topology (Section 4).
- We demonstrate the performance/area tradeoff of augmenting a mesh interconnect with various amounts of RF-shortcut, providing an average performance improvement of 13% (up to 18%) for an area cost of roughly 0.13% on the active silicon layer. This corresponds to an average 22% reduction in the average latency experienced by each packet in the network (Sections 5.1 and 5.3).
- We study the deadlock problem that can occur when routing using general shortcuts, and evaluate

the application performance impact of two types of deadlock solutions: a turn-model based route restriction as well as a progressive deadlock detection and recovery scheme (Section 5.2).

- We observe that RF access points can become bottlenecks when many packets try to access the RF at once, and show that this congestion can be alleviated by statically restricting RF shortcut usage (Sections 5.3.1 and 5.3.2).
- Finally, we propose strategies which dynamically detect congestion at RF-I shortcuts and throttle RF-I usage accordingly (Section 5.3.3), and conclude this work in Section 6.

2. On-Chip RF Interconnect

RF interconnect was proposed in [4] as a high aggregate bandwidth, low latency alternative to traditional interconnect. Its concept and benefits of enhancing it with FDMA and CDMA were also demonstrated, mainly for off-chip on-board applications [5][12]. Its benefits to CMP design and performance were not previously analyzed. On chip RF interconnect is a new interconnect concept proposed here for CMPs. It again offers the advantages of a very high aggregate bandwidth and low latency for direct across chip communication to improve the CMP processing speed. It also offers lower power consumption compared with traditional global interconnect, while using the standard digital CMOS technology without additions or modifications. Taking full advantage of CMOS transistors speed, RF-I performance benefits from the CMOS technology scaling trend. It offers even further performance benefits by statically or even dynamically allocating the aggregate chip bandwidth to different users by assigning the available RF bands. In this section we propose the concept of RF interconnect for CMPs, review prior work on RF-I, and compare this interconnect technology to prior proposed alternatives.

2.1. Limits of traditional on-chip interconnect

Traditional global interconnects based on repeater bus wires suffer from two main limitations when considering the future needs of CMPs. The first of these is poor latency scaling – the ITRS [16] projects that the repeated wire delay will remain fairly constant for future technology nodes and may even increase. Moreover, for a global interconnect across the chip, the required energy-per-bit of a repeated bus does not scale well either, since the capacitance and supply voltage scale poorly. However, the amount of on-chip interconnect grows rapidly with each technology generation, causing the total power consumption of the on-chip interconnect to rise at an alarming rate. The result is that traditional repeated bus based global interconnects are major power consumers with limited data rates that do not take full advantage of the available super-scaled transistor bandwidth. For example, in 90nm CMOS technology, the typical repeater signal is running at 4Gbit/s which requires it only occupy about 4GHz of bandwidth. As compared with the f_T (frequency of unity current gain) of 90nm CMOS transistors, which is about 120GHz, the traditional buffer utilizes less than one-tenth of the total available bandwidth.

2.2. RF Interconnect and its benefits

The concept of RF interconnect is based on transmission of *waves*, rather than voltage signaling. When using voltage signaling, the entire length of the wire has to be charged and discharged to signify either ‘1’ or ‘0’. In the RF approach, an electro-magnetic (EM) wave is continuously sent along the wire (treated as a transmission line). Data is modulated onto that carrier wave using amplitude and/or phase changes. A simple and popular modulation scheme for this application is binary-phase-shift-keying (BPSK) where the binary data changes the phase of the wave between 0^0 and 180^0.

Figure 1 demonstrates an example of a ten carrier RF-I. This design uses ten different carrier frequencies ranging from 20GHz up to 200GHz, where each carrier (or band) transmits a 10Gbit/s data stream. Therefore, the total aggregate data rate per wire in this example is 10Gbit/s per carrier × 10 carriers = 100Gbit/s per transmission line. In the frequency domain, BPSK data modulation at a rate of R, takes about R of bandwidth, but it requires a 2R carrier frequency spacing to decrease data interference between channels. As a result, a total available bandwidth of BW can be used to transmit an aggregate data rate of BW/2. In the transmitter, each data stream is first up-converted with individual carrier frequency. After that, these ten up-converted signals are then combined and coupled into the on-chip transmission line. In the receiver, each individual channel signal is down-converted by a selective mixer, and ten different data streams are recovered following their respective low-pass filters. The bottom of figure 1 shows the signal on the transmission line in the frequency domain, with data bandwidth centered on different carrier frequencies,

Table 1: CMOS switching speed scaling

RF CMOS vs. Tech Node (ITRS)	90nm	65nm	45nm	32nm	22nm	16nm
f_T (GHz)	120	170	240	320	400	490
f_{max} (GHz)	200	270	370	480	590	710
Max RF carrier frequency (GHz)	324 [10]	432	592	768	944	1136
Max Aggregate Data Rate with RF-I (Gb/s/wire)	160	216	296	384	472	568

utilizing the total available bandwidth much more efficiently.

RF-I available data rates are inherently limited by the switching speed of conventional CMOS circuits. Faster switching devices enable faster modulation of the signal and also increase the number of available channels that we can exploit. There are two metrics to describe how fast the CMOS can be switched: f_T, which is the frequency of unity current gain and f_{max}, which is the frequency of unity power gain and also referred to as the maximum oscillation frequency achievable using that CMOS technology. In mainstream 90nm CMOS, both f_T and f_{max} already exceed 100GHz for NMOS devices. ITRS predicts that in 22nm and 16nm CMOS, both f_T and f_{max} will be higher than 500GHz. Recently, we have demonstrated a 324GHz voltage controlled oscillator (VCO) in standard 90nm CMOS technology [10], breaking the assumed oscillation frequency barrier of f_{max}. A reasonable rule of thumb to estimate and project the aggregate data rate of RF-I in future technology nodes would be half the maximum carrier frequency possible in that technology. Using this rule in Table 1, we can project a maximum aggregate data rate as high as 568Gbit/s per wire in 16nm CMOS technology.

We have designed a single-band RF-I in 90nm CMOS technology, and have achieved a signal data rate of 5Gbit/s with the carrier frequency centered at 20GHz. Table 2 summarizes the area and power overhead for the Tx and Rx in our design at 90nm.

Table 3 demonstrates characteristics of our implementation of a multi-band RF-I at 90nm CMOS technology. A behavioral model simulation shows that 10GHz channel spacing is sufficient to carry 5Gbit/s data with a low BER. The suggested channels at the 90nm node are 10GHz, 20GHz, 30GHz, 40GHz, 50GHz and 60GHz. Therefore, the total aggregate data rate is 30Gb/s per wire, which is at least six times larger than the data rate of a single traditional repeater bus.

As shown above, FDMA (frequency division multiple access) can be used with RF-I to increase the data rate between two users. It can also be used to allow multiple users to connect to the same shared transmission line and communicate concurrently using different frequency bands. Each user has a transmitter, a receiver or both, each of them selecting a specific frequency band using its up-converting or down- converting mixers. Therefore, N channels can support up to 2N different users simultaneously communicating with each other. Multiple channels can even be assigned to a particular communicating pair to increase the amount of bandwidth available for communication.

2.3. RF-I scaling

Passive devices, such as inductors, consume the dominant portion of the transceiver area. Since the size of a passive device is inversely proportional to the operational frequency, as the frequency of the signal increases, the size of the passive device can be scaled down (Figure 2a). At 20GHz, the size of the inductor is approximately 50μm×50μm. However, due to frequency scaling, the size of the inductor at 400GHz can be as small as 12μm×12μm, about a 20x reduction in area. As long as the carrier frequency can increase at each new generation of technology, the transceiver area will also scale down. According to ITRS, the f_T of the NMOS transistor in 22nm CMOS technology will be around 400GHz. Switching as fast as 400GHz in future generations of CMOS will allow us to have a large number of high frequency channels for an RF-I. In each new technology generation, the number of channels available on a single transmission line can be expected to grow thanks to the faster transistors available (shown in Figure 2b). It is assumed that the average power consumption per transceiver channel is expected to stay constant at about 6mW. The logic behind the assumption is that although RF circuits at higher carrier frequencies require more power, this additional power is compensated by the power saved at the lower carrier frequencies due to higher f_T transistors available with scaling. In addition to increased number of channels, the modulation speed of each carrier would also increase, allowing a higher data rate per channel. As a result, the aggregate data rate is expected to increase by about 40% every technology node, as shown in Table 3. In addition, the cost of the data rate, in terms of area/Gbps and the energy consumption per transmitted bit are expected to scale down.

2.4. Comparison with other types of on-chip interconnect

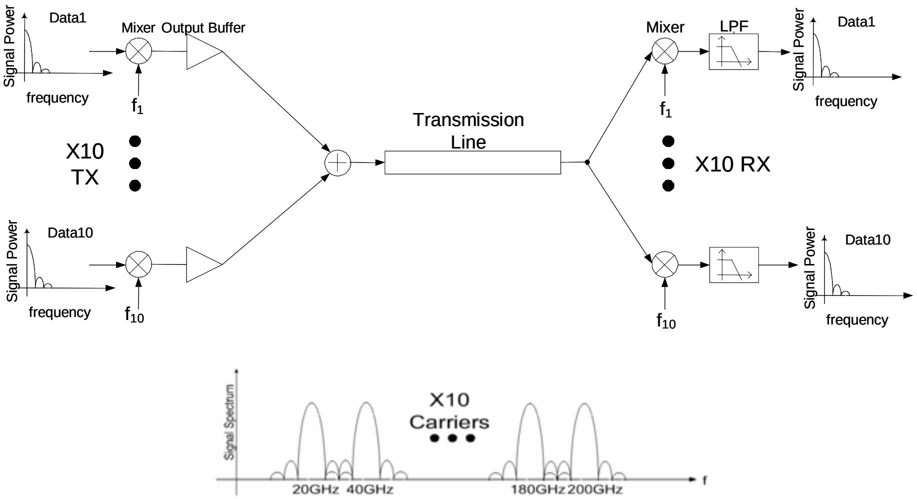


Figure 1: A ten carrier RF-Interconnect and corresponding waveform at the transmission line

We compare our performance estimations to that of the parallel bus based on ITRS [16] and to that of optical interconnects proposed in [11] and extrapolate to future technology nodes up to 22nm. We assume a usage of differential transmission line with 12µm pitch for the RF-I. The traditional bus that uses optimal repeated wires exhibits latency of 800ps for a distance of 2cm at 90nm technology, which gets even worse with scaling to almost 1500ps at 22nm. RF-I and optical interconnect maintain a fairly low and constant latency of about 200ps that is mostly limited by wave propagation time. The energy consumption of the bus is expected to improve from 21pJ/bit at 90nm to 13pJ/bit at 22nm, but RF-I and optical-I are expected to achieve an order of magnitude reduction in energy. As opposed to optical-I energy consumption which does not benefit from scaling, RF-I energy does (mainly due to higher modulation rates). The data rate density of the bus is slowly increasing with scaling from 2Gbps/µm at 90nm to 8Gbps/µm at 22nm but would require more buffers. In RF-I the increase is more dramatic (up to 12Gbps/µm at 22nm) due to addition of carrier frequencies and increase modulation rate, both contributed by scaling of transistors. Optical-I also expects an increase in data rate density due to technology, but uncorrelated to CMOS scaling and therefore smaller. RF-I has the advantage of using the standard digital CMOS technology, while optical-I requires integration with on-chip and off-chip non-CMOS devices adding to package complexity and cost. These devices are also highly temperature sensitive, which raises even more power related issues.

3. The MORFIC architecture

In this section we propose the MORFIC (Mesh Overlaid with RF InterConnect) architecture: a conventional mesh topology augmented with RF-I enabled shortcuts to reduce communication latency. As demonstrated in Section 2, RF interconnects can provide scalable, low-latency communication with conventional CMOS technology. RF-I is also extremely flexible, as different frequencies on different transmission lines can be allocated to the NoC. The flexibility and extreme low-latency of RF interconnect argues for the use of a shared pool of transmission lines that can physically span the NoC. Different points on the NoC can then access this shared waveguide pool for rapid access to other points on the NoC. Collectively, the RF-I can be thought of as a “super-highway,” sending packets long distances with very low latency. By contrast, the standard 2-D mesh links can be thought of as the “city streets” of the chip. RF access points (or “freeway onramps”) in the mesh allow packets to enter the faster RF-I.

The baseline topology we consider in this paper is shown in Figure 3a. It is comprised of a 10x10 mesh of 5-port routers, each with a local port attached to either a processor core, an L2 cache bank, or a DRAM interface (pictured as a circle, diamond, or plus respectively). This design uses 64 cores, 32 cache banks, and 4 memory interfaces, with a 4GHz system clock. The interconnect operates at 2GHz. Each router in the mesh (represented as a square) has a 5-cycle pipelined latency, and routes packets using an XY/YX scheme. In XY/YX routing, half of the packets are routed in the X dimension first, then along the Y axis to their destination. The other half are routed in the Y dimension first, then along the X axis. The baseline mesh links are 16 byte wide, single-cycle buses

Table 2: Power and area of single-carrier RF-I with 20GHz carrier and 5Gbit/s in 90nm CMOS

TX	Power (mW)	Active Area	Passive Area
Mixer	0.5	5um x 5um	50um x 50um
PA	1.5	10um x 10um	50um x 50um
Total TX	2	125um^2	5000um^2
RX	**Power (mW)**	**Active Area**	
Mixer	2	10um x 10um	
Baseband	2	20um x 20um	
Total RX	4	500um^2	

connecting each router to its immediate neighbors, as well as its local attached node. We have implemented full virtual channel support, and have given each buffer a capacity of 8 entries. We select a 2D mesh as our reference topology, as mesh networks allow for regular implementation in silicon, and are simple to lay out. Comparison against other topologies is beyond the scope of this work, but the techniques we describe hence could be employed on a number of designs of this scale.

We have chosen this overall topology to reduce long-distance communication bottlenecks on the chip. The largest messages being sent in the network are DRAM responses, which each carry 128-byte L2 cache blocks from a main memory interface to a fetching L2 bank. By surrounding the DRAM interfaces with L2 cache banks, we reduce the distance that the largest messages must travel, and reduce their spatial overlap with traffic between cores and L2 caches. This computation/storage spatial hierarchy, with a cluster of L2 cache banks at the center of the chip, surrounded by the processor cores they service, has been explored in other designs, namely Beckmann and Wood's CMP-SNUCA [2], which surrounded a mesh-interconnected bank-cluster with eight CPU cores.

Our shared L2 cache is a statically address-partitioned NUCA with a directory-based MSI coherence protocol. Our coherence protocol has been optimized to reduce message injection via silent evictions and reply-forwarding [6]. Furthermore, our protocol is robust enough to tolerate network reordering of all coherence messages, including silent evictions and coherence acknowledgements.

As we will demonstrate, the MORFIC architecture:

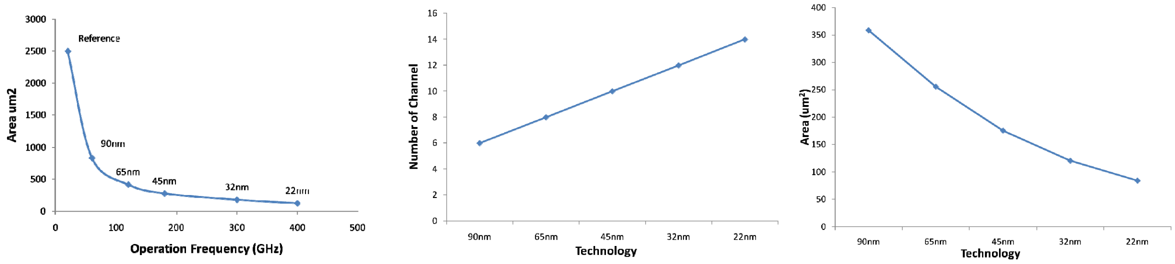
- Provides scalable, low-latency performance for a forward-thinking many-core mesh NoC topology.
- Avoids costly arbitration for RF-I frequencies across the mesh topology.
- Allows simultaneous communication on different frequency bands for improved bandwidth.

3.1. Related Work

Beckmann and Wood [1] introduced the use of transmission lines for mitigating the impact of the communication latency between L2 cache banks and the cache controllers. They have outlined CMP floorplans optimized for less complex circuitry, where the cache banks reside near the edges of the chip and cache controllers are located in the center of the chip. Transmission lines provide a low latency shortcut between two components distantly located from each other. However, in future CMPs with a large number of cores and cache banks on the die, it is essential to extend such schemes for improving the latency of both core-to-core as well as core-to-cache communication. And while transmission lines provide low-latency shortcuts in a mesh topology, they do not take advantage of frequency divided communication.

Ogras and Marculescu [13] explored the enhancement of a standard mesh network via addition of application-specific long-range links between pairs of frequently communicating routers. The goal of their work is to maximize the amount of traffic that could be injected into the network before saturation was reached. Using a profile of an application's network traffic, their algorithm searches through all possible shortcut permutations and then estimates the effect of these shortcuts on the critical traffic load. In order to avoid deadlock in routes constructed using these links, they employ a turn-model route restriction called South-Last (which we implement and evaluate in Section 5.2). Unlike the single-cycle shortcuts employed in the MORFIC architecture, Ogras and Marculescu implement their long-range links using a higher-latency point-to-point pipelined bus. The application-specific nature of these long-range links makes them unsuitable for use in a general-purpose architecture, and no algorithms are presented to adapt their use to changing communication conditions.

Kirman et al. [11] have employed optical technology to design a low-latency, high-bandwidth shared bus. While their design does take advantage of low-latency and high bandwidth via simultaneous transmission on different wavelengths, they examine optical interconnect to augment a bus topology instead of a more scalable mesh topology. Moreover, none of these studies have considered dynamically adapting shortcut utilization during an application run. Applications exhibit a wide range of communication patterns, and for future CMPs with large number of cores on chip overuse of shortcut access points can lead to substantial congestion, as we will demonstrate.

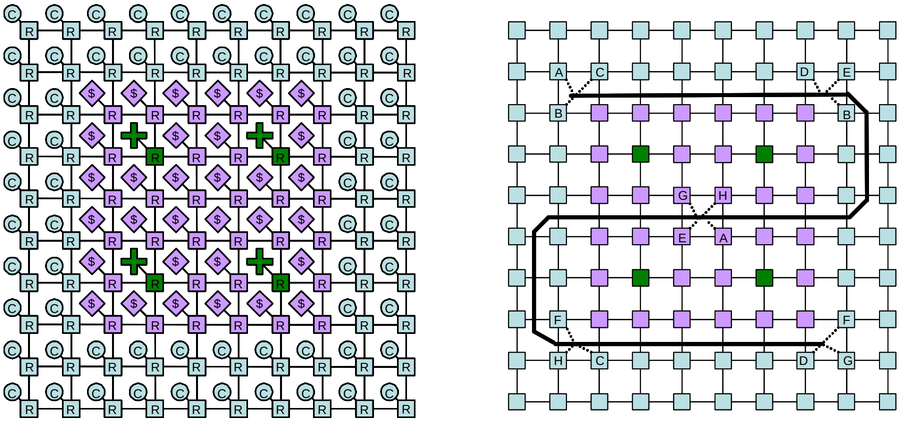


(a) (b) (c)

Figure 2: RF-I (a) inductor scalability, (b) available channels, (c) transceiver area per Gbps

Table 3: Scaling trend of RF-I

Technology	# of Carriers	Data rate per band (Gb/s)	Total data rate per wire (Gb/s)	Power (mW)	Energy per bit(pJ)	Area (Tx+Rx) mm^2	Area/Gbps (μm^2/Gbps)
90nm	6	5	30	36	1.2	0.0107	357
65nm	8	6	48	48	1	0.0112	233
45nm	10	7	70	60	0.85	0.0115	164
32nm	12	8	96	72	0.75	0.0119	124
22nm	14	10	140	84	0.6	0.0123	88



(a) (b)

Figure 3: Mesh topologies – baseline (a) and with RF shortcuts (b)

4. MORFIC architectural decisions

4.1. RF link placement

Additional hardware is required at each access point to interface with the RF transmission line (as described in Section 2). Therefore we must decide where to place the access points to the RF transmission line, using the 10x10 mesh topology in Figure 3a as a starting point.

The MORFIC architecture is realized via a rounded Z-shaped RF transmission line connecting three routers at each corner of the mesh to four routers in the center of the mesh, as demonstrated in Figure 3b. The Z-shaped waveguide is not just a single shared bus – it is capable of supporting multiple simultaneous point-to-point transmissions over each RF band. The Z-shaped topology is intended to illustrate the connection of the transmission line to the set of transmitters and receivers. The actual wire layout would use mainly Manhattan geometry with smooth corners.

The placement of the RF access points in the mesh was designed to minimize the number of cycles between distant endpoints of the mesh. We logically divided the mesh into five sectors – the four corner sectors for core accesses and the center sector for L2 accesses. Using this topology, every sector can reach every other sector through the RF transmission line. Each lettered router in Figure 3b is equipped with an additional sixth port, over which it can transmit and receive data to/from the RF-I.

4.2. Packet routing

One design concern is how to determine when a packet should make use of RF-enabled shortcuts – standard XY/YX routing is not sufficient in these cases. Instead, we add a routing table to each router in our topology. The table has one entry for each destination in the mesh (in our case 100 entries) and has three bits per entry. Each entry indicates what direction a packet should travel to get to the destination – one of either five or six possible directions depending on the radix of the router. For the six radix routers (i.e. the routers with RF-I access), one possible direction is to use the RF shortcut. These routers are statically configured based on the current topology of the mesh using an all-pairs shortest path algorithm.

It is possible that RF-I access points could become an NoC bottleneck if too many packets want to use the shortcuts. This is akin to too many cars trying to get on to a freeway at a single on-ramp. In such cases, we would like to have some packets use XY/YX routing instead of using RF-I – analogous to having some drivers use surface streets when highway conditions are congested. We have a single bit per packet which is used to determine whether the packet will use the routing table or XY/YX routing. This bit is set when the packet enters the network. We will explore different policies for setting this bit in Section 5.3.

4.3. The perils of deadlock

The RF shortcuts in our topology create cyclic dependencies in the mesh that can lead to deadlock. For example if router X wants to transmit a flit to router Y, but Y's incoming buffer for that port is full, X will have to wait. Eventually other routers waiting to transmit flits to X will also block, one of which could be Y. This condition of circular dependence may lead to a state of deadlock, where every router involved in the cycle is waiting for an output buffer to become free on an upstream router.

There are several techniques that can be employed to deal with deadlock conditions. The turn-model [8] completely avoids deadlock by making sure that the set of allowable turns made by packets in the network cannot form a cycle. Thus, by selecting a subset of legal turns which a packet can make along its route, deadlock can be avoided. We consider one turn-model approach called South-Last, which was used by Ogras and Marculescu in their Small World design [13]. South-Last imposes two restrictions on packets entering a router. If a packet is traveling south, it should continue traveling southward (either South-West, South-East, or directly South). Also, if a packet enters a router traveling west (entering on its east port), it cannot be routed such that it makes a U-turn, and travels back east. These restrictions apply to outbound long-range shortcut links as well as to links in the baseline mesh. With these types of restrictions in place, circular buffer dependences cannot occur. However, these same restrictions can potentially limit the achievable performance of RF shortcuts, by disallowing a packet to use them under certain conditions.

A less restrictive option is to allow turns that form a cycle, but to detect potential cases of deadlock and recover from them. This strategy of deadlock detection and recovery is based on theory presented by Duato and Pinkston [7], which states that deadlock-free routing can be achieved as long as a connected channel subset is deadlock-free. In other words, there is no need to restrict the possible turns made in the network, as long as we are able to detect potential deadlock conditions and react by routing packets on a reserved emergency channel, which itself can never deadlock.

Our deadlock detection scheme works as follows. If a source router S tries to transmit a flit of data to a receiving router T, and T's inbound queue is full on the port connecting S to T, then S must block and retransmit the flit later. In this case, S will use the same channel to transmit a waiting-list to T. The waiting-list is a bit-vector with one bit per router in the NoC mesh (in this case size 100). Every bit set in the waiting-list identifies a router that is waiting on the recipient. When S sends a list to T, at minimum the bit in the waiting-list corresponding to router S is set, so that T knows that S is waiting on T. S will also set bits in the waiting-list corresponding to any routers that are waiting on it (as it may have received some waiting lists as well). In this manner, each router accumulates a list of what other routers are waiting on it. In cases of circular dependences that lead to deadlock, a router will eventually detect that it is waiting on itself – it simply need detect when the bit corresponding to itself is set in its own waiting-list. This condition raises deadlock. Note that this is a conservative and imprecise detection mechanism, as we are really detecting circular buffer dependence, which is a necessary condition of deadlock.

A router sends a waiting-list whenever a message cannot be sent due to a full buffer on the receiving router. This waiting-list message does not interfere with other communication because the communication

Table 4: Simulation parameters

Core Parameters	
Number of Cores	64
Fetch/Issue/Retire Width	6/3/3
ROB/IQ Size	128 / 12
Branch Misprediction	6 cycles
Branch Predictor	Hybrid, 16k-entry
L1 Instruction Cache	8 KB, 4 way, 32 byte block size, 2 ports
L1 Data Cache	8 KB, 4 way, 32 byte block size, 1 port
L2 Cache Parameters	
Number of Banks	32
Each bank: 256KB, 8 way, 128 byte blocks, 1 port	

link would have been idle otherwise (as the receiving router buffer is full). When T's incoming queue becomes un-blocked, S will send T a one time waiting-list-clear message, which contains the same contents as the original waiting-list message. Using this, T will know that the routers corresponding to bits set in that message are no longer waiting on it. If some other router U is also trying to send to T, and has transmitted its own waiting-list to T, then bits common between S and U's waiting lists will naturally be set on the next attempted send from U (assuming the clog has not been relieved).

When deadlock occurs, all queued packets in the network become XY-routed packets which can no longer use shortcuts. These packets will be routed on an emergency virtual channel, a spare virtual channel which is only used when this condition occurs. As XY-routing does not allow turns which would form a cycle, this spare virtual channel will not deadlock, and by Duato and Pinkston's theory [7] the network will remain deadlock-free. As soon as all packets are converted to XY routed packets which can only use the spare VC, the deadlock condition is lowered. Only packets injected into the network after this point will have an opportunity to use shortcuts, unless and until deadlock occurs again.

We implemented both South-Last turn restrictions as well as the progressive deadlock detection and recovery scheme described above. In Section 5.2 we discuss the performance impact of these approaches.

4.4. Frequency band allocation

Another issue is how to allocate frequency bands in the shared transmission-line pool. One approach would be to dynamically assign frequency bands to communicating pairs on the fly – this would ideally provide maximal allocation flexibility and bandwidth utilization. It requires some arbitration mechanism to assign frequency bands to both the sender and receiver of data. The latency of this operation includes sending a communication request to the arbiter, the actual arbitration, sending the corresponding frequency to both sender and receiver, the actual communication on that frequency, and the signal to release the assigned frequency. Rather than add this latency, we consider a topology where the frequencies are distributed between communicating pairs in the mesh topology at a coarser granularity, amortizing the cost of frequency assignment over a larger number of cycles. Frequency-band assignment could potentially be done by the hardware or software (i.e. application or OS).

In this paper, we logically organize the shared waveguide as eight bidirectional shortcut links. Each lettered router in Figure 3b represents an end-point for these bidirectional links. For example, the router labeled A in the upper left sector of the mesh can transmit directly to the router labeled A in the center sector of the mesh, and vice versa, in a single clock cycle. However, neither router labeled A is able to transmit data directly to any other RF-enabled router with a different letter: the A in the upper left cannot send directly to E in the center via the RF-I. However, a packet can be sent from A in the upper left to A in the center via the RF-I, and then be routed west to E over a standard mesh link.

This distribution of shortcuts allows messages in a sector to easily hop to neighboring sectors directly via the RF transmission-line pool. For example, at the upper left sector, a message can either take its C-labeled shortcut router down to the lower-left sector, its B labeled router to the upper-right sector, or its A-labeled router to the center of the mesh. In the case of the four corner sectors, diagonal communication (i.e. from the upper left to the lower right sector) will take two trips on the RF transmission line (i.e. one to the center, and then one to the destination sector).

For this paper, we assume that the available transmission-line pool bandwidth is evenly allocated to all eight bidirectional shortcuts for the duration of program execution – therefore the frequency assignment is done only once for the entire application. However, future work will consider heterogeneous allocation of bandwidth based on run-time network congestion. This is a natural fit for this frequency allocation strategy: periodic coarse-grain assignment as a means of NoC adaptation.

5. Experimental Results

We have adapted the SESC [14] framework for this study, completely rewriting the on-chip network topology and L2 cache code. Our core and cache bank simulation parameters are summarized in Table 4.

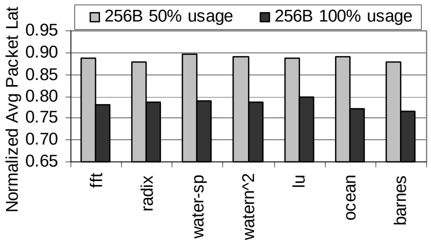


Figure 4: Latency reduction with 256B RF-I

A five-cycle fully pipelined router delay was assumed, with 1-cycle delays for each stage. The stages are routing, virtual channel arbitration, switch allocation, switch traversal, and output link traversal. Our simulator models network routers and their queues with a behavioral model of store-and-forward packet switching. A packet stalls within a router for one of two reasons: either the outbound link is occupied, or a buffer at the destination router is full.

We use seven shared-memory multithreaded applications from the Stanford SPLASH suite [15] to evaluate RF shortcuts: Barnes, FFT, LU, Ocean, Radix, Water-Nsquared, and Water-Spatial. These applications were configured to use 64 threads with their standard input set, and each of these threads was mapped to one of the MORFIC cores. We simulated each benchmark by fast-forwarding to its parallel section, and then running to completion.

5.1. Application performance benefits of RF Interconnect

An allocation of 256B of RF bandwidth, evenly divided between each of the eight point-to-point shortcuts described in Section 4.4, would match the baseline mesh bandwidth for each of these links. Put differently, between each endpoint of an RF shortcut, 16B of link bandwidth would be available each cycle, the same as between neighboring nodes on the baseline mesh. In a 32nm design, this RF provision would consume roughly 0.51 mm^2 on the active silicon layer. On a 400 mm^2 die, this would be an area overhead of 0.13%.

In Figure 5a, we demonstrate the performance gain achieved by enhancing our mesh with an overlay of 256B of RF bandwidth. In each column, the left bar (256B 50% usage) represents the following policy: half of the packets injected into the network use standard XY/YX routing, and the other half proceed along their shortest path, using the routing table described in Section 4.2. The right bar (256B 100% usage) represents a policy allowing all packets to proceed along their shortest path, requiring a routing table lookup at each intermediate router. Hence, we refer to this latter policy as opportunistic shortcut usage, as every packet is given the opportunity to exploit RF shortcuts for latency savings. These bars are normalized to the runtime of each application on the baseline mesh topology, with no RF-shortcuts and only XY/YX routing.

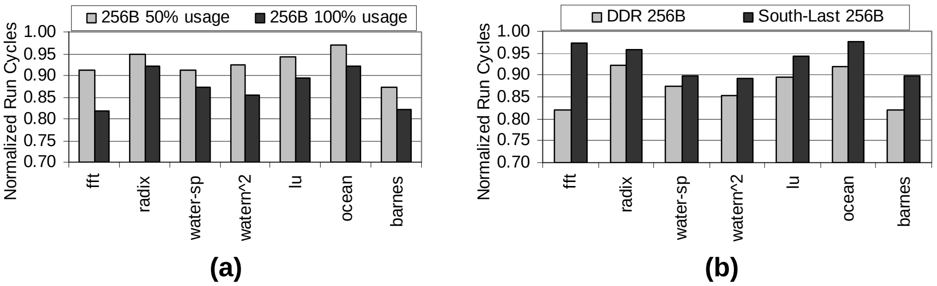
At an allocation of 256B RF bandwidth, higher performance is achieved by sending more packets along their shortest-path, as the congestion at the RF access points is outweighed by the latency-savings experienced for cross-chip traversal. The opportunistic use of RF shortcuts in the 100% usage case leads to an average performance gain of 13% on these benchmarks, with the highest gains experienced by FFT and Barnes at 18%.

These gains are consistent with our investigation of the latency and bandwidth sensitivity of these Splash-2 benchmarks on the baseline 10x10 mesh. We found that doubling the pipelined router latency from 5-cycles to 10-cycles had a drastic effect on these benchmarks (an average of 44% performance degradation), whereas halving link bandwidth from 16B to 8B only degraded performance by an average 7%. RF shortcuts reduce the number of cycles required to traverse long distances on chip, and this latency savings translates directly into increased application performance.

Corresponding to this performance increase, we notice a reduction in the average latency experienced by each packet en route from its source to its destination. Figure 4 shows the average packet latency for opportunistic as well as 50% usage of 256B of RF-interconnect, normalized to the average packet latency on the baseline mesh. The latency savings vary little by benchmark, and average 11% for 50% usage, and 22% for opportunistic usage. Barnes experiences the largest average latency savings, at 24%.

5.2. Performance impact of deadlock strategy

In Section 4.3 we described two strategies for dealing with the inevitability of deadlock in a shortcut-enhanced mesh network: the South-Last turn model [13] and a progressive deadlock-detection and recovery (DDR) scheme using an emergency virtual-channel [7]. In Figure 5b, we demonstrate how each of these deadlock strategies performs with an allocation of 256B of RF bandwidth. As the previous section indicated, opportunistic (100%) usage of RF-I leads to higher performance at 256B of allocation, therefore in this section we continue to use RF opportunistically. As in Figure 5a, these results are normalized to the



(a) (b)

Figure 5: (a) Performance with 256B RF-I bandwidth and (b) DDR vs South-Last approaches

performance of each application on the baseline mesh, with no RF shortcuts. (Note that the results of Section 5.1 used the DDR scheme, and that the 256B 100% usage results in Figure 5a are identical to the DDR 256B results presented in Figure 5b.)

As indicated by the results, deadlock-detection and recovery always outperforms the South-Last turn-model. At this bandwidth allocation, the South-Last rules are too restrictive, and certain routes which would have decreased packet latencies are disallowed due to their potential to cause deadlock. As a result, the average performance gain experienced using the South-Last policy is only 7%, around half of that experienced when using a DDR strategy. Although the DDR strategy allows deadlock to occur, the recovery is not time-consuming, as all packets in flight will switch to XY-routing. Additionally, deadlock is detected very infrequently (tens to hundreds of times per application, over many millions of cycles). However, DDR requires additional resources to achieve its performance gains, in the form of an extra buffer on each physical router channel, as well as the transmission of waiting-list and waiting-list-clear messages to blocked routers. For the rest of this paper, we will use the DDR strategy.

5.3. Static bandwidth allocation

5.3.1. Opportunistic shortcut usage. As demonstrated in Section 5.1, the opportunistic usage of shortcuts at 256B of allocation leads to a significant performance increase. However, such an RF allocation may be too costly for designers. In Figure 6, we explore several allocations of aggregate RF bandwidth, ranging from 16B to 256B, and normalize their runtime on each application to the baseline case (no RF shortcuts, represented by the horizontal line at 1). When less RF bandwidth is available, opportunistic shortcut usage presents a problem, as application performance can degrade by more than 400% (for Barnes) for small RF allocations. For instance, in the case of 16B of RF, each shortcut is only allocated a single-byte of bandwidth in each direction. In this case, packets naively routed on their shortest path will congest the shortcut access points, causing queues to fill up at shortcut entrances. This can result in massive network congestion. At this design point, the latency-saving potential of RF is dwarfed by its negative impact on congestion.

5.3.2. Statically restricted shortcut usage. As opportunistic usage of RF-I can create bottlenecks at small RF allocations, we attempt to alleviate this congestion by allowing some, but not all, packets to use the RF. As mentioned in Section 4.2, a single bit in each packet can be used to determine whether the packet will use its shortest path (which may include the use of RF-I shortcuts), or whether the packet will use XY/YX routing, avoiding RF entirely.

In Figure 7a, we show the performance obtained on an aggregate 32B of RF allocation, where each shortcut is given 2B in each direction. The bars indicate the runtime of each application (normalized to no RF interconnect), where 25%, 50%, 75%, and 100% of the packets entering the network are sent on their shortest path. Figure 7b presents these same configurations for 96B of RF allocation, where each shortcut is given 6B in each direction. At 32B of total RF bandwidth, performance increases as shortcuts are used less. However at 96B, this relationship is more complex. For some applications, such as Water-Nsquared, LU, and Barnes, performance improves as shortcuts are used more. This is due to the fact that the performance lost to shortcut bottlenecks is outweighed

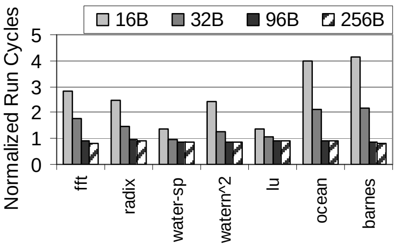
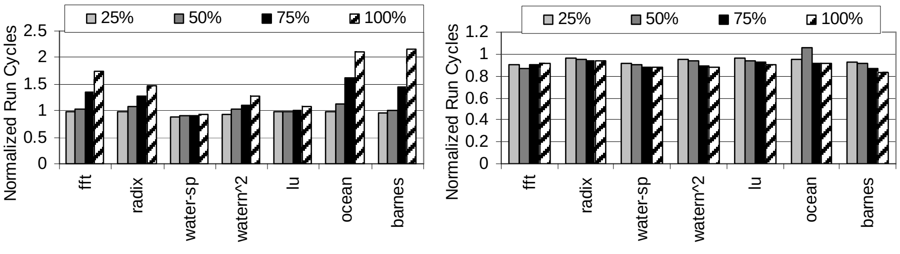
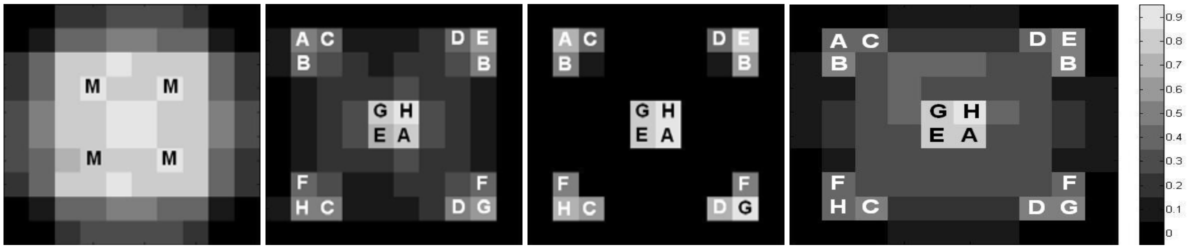


Figure 6: Performance obtained by varying RF-I bandwidth



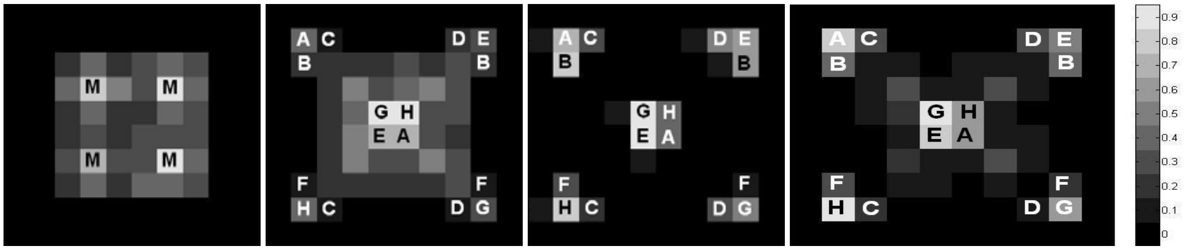
(a) (b)

Figure 7: Reduced RF-I utilization for (a) 32B bandwidth and (b) 96B bandwidth



(a) (b) (c) (d)

Figure 8: Number of flits sent, normalized to maximum across (a) no RF-I, (b) 256B RF-I 100% usage, (c) 32B RF-I 100% usage, and (d) 32B RF-I 25% usage



(a) (b) (c) (d)

Figure 9: Number of router stalls, normalized to maximum across (a) no RF-I, (b) 256B RF-I 100% usage, (c) 32B RF-I 100% usage, and (d) 32B RF-I 25% usage

by the performance gain from shortcut latency reduction. For FFT, the best performance is experienced when half of the packets use their shortest path, and the other half use XY/YX routing.

Figure 8 and Figure 9 are color-charts depicting the activity of each router in the 10x10 mesh, aligned as presented in Figure 3. In Figure 8a and Figure 9a, the routers attached to memory-interfaces are labeled M, and in Figure 8 and Figure 9 b,c, and d, the routers attached to RF-I access points are labeled as in Figure 3b. Figure 8 depicts the number of flits sent at each router, and Figure 9 depicts the number of cycles that a router must stall while waiting for an occupied outbound link. These values are reported for the benchmark Water-Nsquared, for no RF-I (in a), 256B RF-I used opportunistically (in b), 32B of RF-I used opportunistically (in c), and 32B of RF-I where 25% of the packets are routed along their shortest path. The square representing each router is shaded relative to the maximum value across each configuration (where a lighter shade represents more activity). Comparing the two figures, it is clear that when no RF-I is used, the routers attached to L2 cache banks and memory interfaces experience the most stalls, and send the most flits. When shortcuts are used opportunistically, they send the most flits, but also experience a great number of stalls. At 32B of opportunistic usage, where each shortcut is only allocated 2B in each direction, it is clear that stalls are concentrated around the shortcut access points. However, when shortcut usage is restricted to 25%, the concentration of traffic becomes more spread out, as does the pattern of stalled routers.

5.3.3. Dynamic restricted shortcut usage. Based on the results from the previous section, we note that no shortcut-usage restriction fits all the applications for a given amount of RF bandwidth. Some applications may experience better performance when sending more packets along their shortest path route, whereas others may experience more congestion, and require further RF restriction.

We have explored two simple strategies to try and locate the optimal shortcut restriction at runtime: one which searches for the optimal RF-I shortcut utilization one time at the beginning of the parallel section of each application, and another which continuously adapts shortcut utilization throughout application execution. The performance impact of each of these strategies is presented in an extended version of this work [3]. We find that the ability to adapt to changing network conditions leads to better application performance.

6. Summary

In this work, we have motivated the use of multi-band RF-interconnect as a low-latency alternative to traditional on-chip interconnect in CMP architectures. Starting with a physical implementation in 90 nm process, we have applied ITRS projections to show how RF-I will scale to future process technologies, and evaluate its potential to boost the performance of shared-memory multithreaded applications on a CMP. Assuming a 400mm^2 die in 32 nm process, we have demonstrated that in exchange for 0.13% of area overhead on the active layer, RF-I can provide an average 13% (max 18%) boost in application performance, corresponding to an average 22% (max 24%) reduction in packet latency. We have also evaluated two different approaches to deadlock, and found that deadlock detection and recovery outperforms a restrictive deadlock avoidance strategy on this topology. We have also noticed that RF access points may attract too much traffic if strict shortest-path routing is used, and have proposed to avoid these bottlenecks by detecting and reacting to network congestion.

Acknowledgments

This research was supported in part by NSF grant CCF-0133997, Semiconductor Research Corporation grant 2005-TJ-1317, and DARPA grant SA5430-79952. The authors acknowledge the support of the GSRC Focus Center, one of five research centers funded under the Focus Center Research Program. We wish to thank our reviewers for useful feedback.

References

[1] B. Beckmann and D. Wood, “TLC: Transmission Line Caches,” in *Proceedings of MICRO-36*, December 2003.

[2] B. Beckmann and D. Wood, “Managing Wire Delay in Large Chip-Multiprocessor Caches,” in *Proceedings of MICRO-37*, December 2004.

[3] M.F. Chang, J. Cong, A. Kaplan, M. Naik, G. Reinman, E. Socher, S.-W. Tam,“CMP Network-on-Chip Overlaid with Multi-Band RF-Interconnect,” UCLA Computer Science Department Technical Report UCLA/CSD-TR-07-0032, December 2007.

[4] M.F. Chang, V.P. Roychowdhury, L. Zhang, H. Shin, Y. Qian, “RF/wireless interconnect for inter- and intra-chip communications,” in *Proceedings of the IEEE*, vol 89. no 4, April 2001.

[5] M. F. Chang, I. Verbauwhede, C. Chien, Z. Xu, J. Kim, J. Ko, Q. Gu, and B. Lai, “Advanced RF/Baseband Interconnect Schemes for Inter- and Intra-ULSI communications,” in *IEEE Transactions on Electron Devices*, July 2005.

[6] D. Culler, J.P. Singh, A. Gupta, *Parallel Computer Architecture: A Hardware/Software Approach*, Morgan Kaufman Publishers Inc, San Francisco, CA, 1999.

[7] J. Duato and T.M. Pinkston, “A General Theory for Deadlock-Free Adaptive Routing Using a Mixed Set of Resources,” in *IEEE Transactions on Parallel and Distributed Systems*, vol. 12, no. 12, December 2001.

[8] C.J. Glass, L.M. Ni, “The Turn Model For Adaptive Routing,” in *Proceedings of ISCA-19*, May 1992.

[9] R. Ho, K.W. Mai, and M. Horowitz, “The Future of Wires,” in *Proceedings of the IEEE*, vol 89. no 4, April 2001.

[10] D. Huang, T. LaRocca, L. Samoska, A. Fong and M.C.F. Chang, ”A 324GHz CMOS Frequency Generator Using a Linear-Superposition Technique,” *Tech. Dig. ISSCC 2008*, February 2008.

[11] N. Kirman, M. Kirman, R.K. Dokania, J.F. Martinez, A.B. Apsel, M.A. Watkins, and D.H. Albonesi, “Leveraging Optical Technology in Future Bus-based Chip Multiprocessors,” In *Proceedings of MICRO-39*, December 2006.

[12] J. Ko, J. Kim, Z. Xu, Q. Gu, C. Chien, and M.F. Chang, “An RF/Baseband FDMA-Interconnect Transceiver for Reconfigurable Multiple Access Chip-to-Chip Communication,” in *2005 IEEE International Solid-State Circuits Conference (ISSCC) Digest of Technical Papers*, February 2005.

[13] U.Y. Ogras and R. Marculescu, “’It’s A Small World After All’: NoC Performance Optimization Via Long-Range Link Insertion,” in *IEEE Transactions on VLSI Systems*, vol. 14, no. 7, July 2006.

[14] J. Renau, B. Fraguela, J. Tuck, W. Liu, M. Prvulovic, L. Ceze, S. Sarangi, P. Sack, K. Strauss, and P. Montesinos, “SESC Simulator,” Jan. 2005, http://sesc.sourceforge.net.

[15] S.C. Woo, M. Ohara, E. Torrie, J.P. Singh, and A. Gupta, “The Splash-2 Programs: Characterization and Methodological Considerations,” in *Proceedings of ISCA-22*, June 1995.

[16] *International Technology Roadmap for Semiconductors: Semiconductor Industry Association*, 2006.