BTP Phase 1 Report

Wireless NoC (WiNoC) Architectures

Guide : Dr. Hemangee Kapoor

Apoorv Kumar

09010111

□

Department of
Computer Science & Engineering
Indian Institute of Technology - Guwahati

Where the ENIAC is equipped with 18000 vacuum tubes and weighs 30 tons , computers in future may have only 1000 vacuum tubes and weigh only 1.5 tons. - Popular Mechanics , 1944

Contents

Introduction

NoC is currently the many-core paradigm of the future, with many implementations already out in the wild showing promising prospects. The scaling issues however continue to bug the interconnect based architecture. In one of the most touted products, the Intel tera-flop processor for instance, sending data from one core to another might take as much as 75 cycles. As we reach a far more denser integration , with more elements on chip, the problem is only likely to aggravate.

In this report we look at some of the primary solutions that are being considered for replacing the wired interconnects. The ones include - RF Interconnects , optical/photonic interconnects and wireless interconnects. Out of these, wireless holds the greatest promise because of easy implementation, high level of reconfigurability, viability and available data-bandwidth.

1 Why NoCs / Cost Considerations.

1.1 Overview

Before we can start on the best/optimal architecture we have to specify on what grounds we would be measuring performance. Further we have to ensure that a proposed NcC architecture does improve upon alternative architectures on these grounds.

Under most widely accepted norms , we analyze the generic cost in terms of area and power.

NoCs help us overcome the common issues faced by currently used architectures. This chapter discusses the problems in brief.

1.2 Power

Power has become the most critical constraint in the design of many systems, from high-performance servers to em- bedded battery-operated devices. Recognizing the need to target increasing power consumption and design complexity in these systems, designers have turned to multi-core architectures such as chip multiprocessors (CMPs) and multiprocessor systems-on-a-chip (MPSoCs).

1.2.1 Network Power

We model both the processor and network using a graph *G = (N, L)* where N is the set of nodes and L is the set of links in G. This holds true for the ordinary multi-processors as well , and uni-cores just behave as a trivial subset. The only difference lies in how the fabric behaves. Whereas in unicores this power can be safely neglected , this power is included in the shared-bus power consumption for the ordinary multicores.

There are many frameworks for the estimation of this power , like LUNA and ORION.

Most commonly used “proxy-parameters” for the power consumption include link-utilization (represented by α). Dynamic network power, is a function of activity/utilization and energy costs of each of the key router components. Using link

utilization as an abstraction for network power , the level of activity at a network link is used as a measure the overall power consumption of that network router and link.[1]

1.2.2 Processor Power

Since we can model the processor cores using the same above graph, the same frameworks would work fine here as well. But in contrast to the network model , the data transfer over nodes is much cheaper and faster process. Again we can use resouce utilization as a proxy for power, similar to the way we abstract network power through link utilization in Section 1.2.1. We abstract the power consumption of a processor by the utilizations of individual resources. The summation of the energy costs of each component (functional units, register file, caches etc.) is captured by each respective utilization function.

In the case of the network fabric, the utilization of all of the components is approximately equal because message flowing in networks consume roughly the same amount of energy per hop. Estimates of individual network components' energy consumptions are thus not necessary because it is a relative power measure; constant factors can be eliminated. However, in the case of a processor pipeline, each instruction will not consume the same amount of energy. The component utilizations hence cannot be removed and abstracted as a single utilization. Relative estimates are thus required in order to obtain a relatively accurate estimation of processor power.

1.3 Area

With the failure of Dennard scaling and thus slowed supply voltage , scaling core count scaling may be in jeopardy, which would leave the community with no clear scaling path to exploit continued transistor count increases. In any case , preserving the transistors for the purpose of actual computing is a priority for any designer. For instance, by increasing the buffer size at each input channel from 2 to 3 words, the router area of a 4x4 NoC increases by 30% or more[2].To this end we like to keep the buffers (that add absolutely nothing to computations) to the minimum.Buffer consumes much of silicon and power and this is likely to reduce performance.

2 Performance parameters and QoS on NoCs.

2.1 Overview

In this section we discuss Quality of Service (QoS) for communications in Systems on Chip (SoC), as the minimum guarantees provided on the associated parameters. SoC inter-module communication traffic can be classified into four basic classes of service: signaling, real-time, RD/WR and block-transfer .

As we have already seen , latency, throughput and reliability are the most widely accepted parameters to be considered when talking about QoS. Communication traffic of the target SoC can be analyzed by means of analytic calculations and simulations, and QoS requirements (delay and throughput) for each service class can be accordingly derived. In this section we analyze how we can conform to a QoS.

2.2 Class based traffic

We have different types of traffic. Some are bound by strict time, delay and reliability constraints, while others are not of much importance and can be done at liesure. Thus in order to quantize QoS levels and check for conformity , we need to categorize the traffic into 4 broad groups.[3]

1. **Signaling** covers urgent messages and very short packets that are given the highest priority in the network to assure shortest latency. This service level is suitable for interrupts and control signals and alleviates the need for dedicating special, single-use wires for them.
2. **Real-Time** service level guarantees bandwidth and latency to real-time applications, such as streamed audio and video processing.
3. **Read/Write (RD/WR)** service level provides bus semantics and is hence designed to support short memory and register accesses.
4. **Block-Transfer** service level is used for transfers of long messages and large blocks of data, such as cache refill and DMA transfers.

2.3 Parameters

2.3.1 Latency

Latency is defined as the precursor time to the actual start of data transfer. This includes the time consumed for connection establishment and propogation delay.

Store-and-forward routing techniques come out as big culprits in assuring low latency. Besides incurring high buffer requirements , they are subject to huge queing delays, as a router waits for next router to have enough free buffer.

Circuit switching is not an option if we are to maintain a constrained latency. Though the service after a connection is made can be better than any other option, the wait while a dedicated path is established can be a long one. Further this leads to poor resource utilization. Since in order delivery is of great importance to us , we find a suitable replacement in virtual-circuit switching. The connection establishment is rather quick , and decent guarantees can be made once the connection is established.

Wormhole routing reduces latency and buffer requirements in the routers. It combines virtual cut-through with small unit size and is a currently accepted solution for providing latency gurantees. In wormhole switching, a packet is transmitted between the nodes in units of flits, the smallest units of a message on which flow control can be performed. The header flit(s) of a message contains all the necessary routing information and all the other flits contain the data elements. The flits of the message are transmitted through the network in a pipelined fashion. Since only the header flit(s) has the routing information, all the trailing flits follow the header flit(s) contiguously. [4]

2.3.2 Throughput

Throughput is defined as the maximum theoretical rate of data transfer over the network.

Though cirtuit switching can provide best throughput guarantees , it is avoided due to the high cost of establishing and managing circuit connections along with poor resource utilization. This is in contrast with the completely connection less data transer , which better utilizes the hardware , but is prohibitory considering the silicon spent on re-ordering buffers.

So we arrive at a trade-off , and choose a hybrid technique. Virtual cut through (as has been discussed earlier) solves the constraints of being plausible , efficient and reliable.

3 Problems with wired NoCs

3.1 Latency

Poor latency is one of the primary issues faced by wired networks. Even as the local wires scale well with improving technology, because they keep getting shorter, the global wires are major source of concern. The chip size doesn't go down with it's components, and so doesn't the distance between seperated members. The contribution of these wires to energy consumption and latency, previously considered negligible, have become relatively considerable because of the improvements in MOS technology.

With the set of components available and required to be connected increases exponentially, we are faced with the problem of packet being queued across multiple elements en-route to it's destination , what we call as a multihop network. These queuing delays, along with being tend to be quite random, contributing to inordinate jitter and slackening QoS.

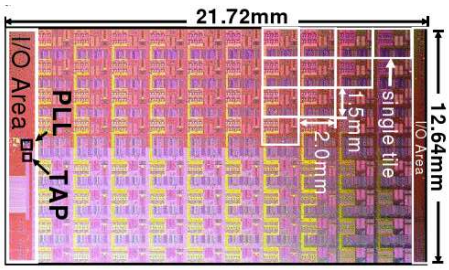


Figure 3.1:

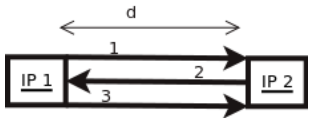
Courtesy: ISSCC 2007: An 80-Tile 1.28TFLOPS Network-on-Chip in 65nm CMOS (Sriram Vangal et al., Intel)

The TeraFlop prcoessor Figure 3.1 has the following specs:

- 65nm CMOS 80 tile NoC
- 10X8 2D mesh network-on- chip running @ 4GHz
- Bisection bandwidth 256GB/s
- 1 TFLOPS @ 1V about 98W
- Worst case latency - **75 clock cycles** - which can kill the performance expected from an 80 core chip

3.2 Throughput

Here is a small analysis of trying to send data over a global line using stop and wait (necessary in case we are using buffers , to ensure that buffer constraints are respected).

Figure 3.2: 

δ - Processing time

β - Packet size (the basic unit of measurement of buffer size)

τ - Time for the wire (acting as a capacitor to 'charge' and render the signal to other size)

n - bit-width of link

We assume that IP1 can instantaneously analyse the ACK and start transmitting the packet. This is plausible as it doesn't need to check it's buffers.

Thus, the theoretical minima of time taken to transmit a packet over the wires using BufferREQ-BufferACK method.

$t = \frac{(\tau+\delta)+(\tau)+(\beta\tau)}{n}$ [brackets correspond to events 1 , 2 and 3 in Fig. 3.2]

Due to ever increasing processing power and speed, δ is the least of our concerns. Thus, to decrease t , we need to increase n. But as proven in [5], the capacitance of wire (and thus τ) scales up almost linearly with n. This leaves us with no other method by which we can improve our bandwidth and latency, except somehow decrease τ, again implausible when we are constrained by wire material and count. Worse, for long links, need to add repeaters to keep the signal alive. These repeaters are more than just pieces of capacitance and the delay introduced by them kills the throughput.

3.3 Energy

The power consumed by (n parallel) wires is usually modelled as

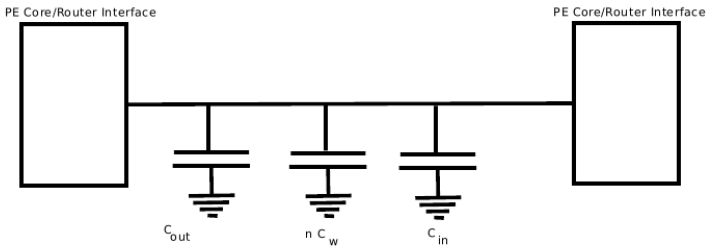


Figure 3.3:

$P_{wire}= \alpha \times C \times f \times V_{dd}^2$

Where:

P_{wire}- is the power consumed.

α- is the probability of switching

C- is the net capacitance $= C_{out} + nC_w + C_{in}$

- $C_{out}-$the intrinsic output capacitance of output port
- $C_{in}-$the intrinsic input capacitance of input port
- $nC_w-$the capacitance due to n parallel wires

f- is the operational frequency - and is constrained by the capcitor-charge-discharge delay introduced by C

$V_{dd}-$is the operational voltage.

Finally , the net power consumption by a link comes out to be:

$P_{link} = (P_{driver} + P_{repeater} + P_{wire}).N_{wires}$

What is apparent from the above equation is that we cannot scale the wire switching frequency indefinitely. On one hand, there is a direct linear factor of frequency in power, and secondly upon increasing the frequency we have to increase the $V_{dd}$to overcome the crosstalk noise, which has a quadratic factor in power.

There is plenty of literature available on optimizing link characteristics. In local links, for instance, we can decrease the worst case delay by temporal interleaving of data over wires. The odd wires carry data at odd clock ticks , and even ones at even. Similarly , for global links, that differ from local links in that they are usually wider and consume more area and power , we can curb the costs and energy consumption by using multiplexing data over fewer number of wires.[5] We do realize that in these optimizations, are hidden compromises in either energy or effective bandwidth.

3.4 Scalability

As the number of components increase virtue of Moore's law, the power consumed by the increased wire interconnects grows at an alarming rate. Not only the global interconnects are major consumers of power, they also deny us the 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 to 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.

4 Solution Paradigms

In order to alleviate the problems cited in the previous section, a lot of novel architectures have been recommended that outperform the wired network asymptotically.

4.1 RF Interconnects

4.1.1 Introduction

RF interconnects use radio frequency (mm-wave) over waveguides for transmitting data across a chip. Whereas the wired transmission relies on transferring a voltage over a set of transmission lines, RFIs employ EM waves sent over a media or wave guide.

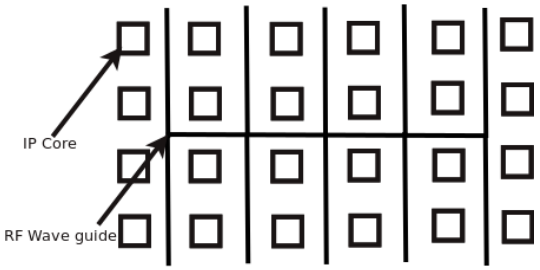


Figure 4.1:

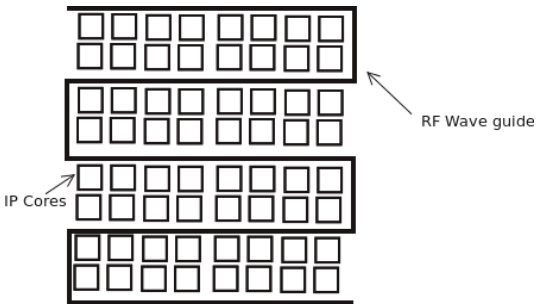


Figure 4.2:
Above two of the various possible implementations of RF-Interconnects.

The usual implementations of RFIs involve one (often several) wave guides that behave like a bus that nodes can write onto and read from. This, again raises issues of arbitration but these can be easily solved using tokens or equivalent arbitration mechanisms.

In RFIs, the transmission of data is achieved by modulating an electromagnetic (EM) wave along a wave guide (a generalized wire) that serves as a transmission line on-chip. In the dielectric, travels at speeds nearing that in vacuum, thus the latency is the smallest that is physically possible.

4.1.2 RFI components

4.1.2.1 Frequency generator

The most basic requirement to have RF interconnects is having an RF (mm-wave range) generator. In order to be dominant over the wired networks, this generation demands efficiency in terms of silicon area and power consumption. Phase-lock-loops (PLLs) have been used for the purpose for some time, but the biggest inhibition in it becoming main stream is it's voracious apetite for power.[6] tries to alleviate this problem by using sub-harmonic injection locking for concurrent generation of multiple mm-wave frequency carriers on-chip by locking them simultaneously to a single reference. They demonstrate the idea on 30GHz and 50GHz voltage controlled oscillators using a 10GHz source where each such oscillator consumes around 4mW of power.

4.1.2.2 Transmission lines

The second requirement for RF interconnects is having a transmission medium. At mm-wave frequencies, super-scaled CMOS processes are one of the most promising paradigms for fabricating on-chip transmission lines for guiding EM waves . These processes, again, can be implemented in a number of ways two of which are using a metal dielectric stack and co-planar waveguide. Setting dimensions present a fundamental tradeoff between performance metrics like loss , latency crosstalk and variation of loss with frequency. Higher loss limits the maximum distance to which a signal can be transmitted, which in turn limits the frequencies we can use for certain communication lines. In particular, higher frequencies with higher loss cannot be used for very large distance communication. Narrowing of transmission lines leads to an increase in resistive loss, although it ensures lower latency because of limited substrate in path. On the other hand, widening them increases substrate loss, and increases latency as the effective dielectric constant would be close to that of silicon, which is 11.7 (compared to 4 of the dielectric).

4.1.2.3 Encoding Scheme

The chips, unlike the RFIs, are all digital. Thus we need to interface these digital and analog paradigms using and ADC (analog to digital converter) and DAC (digital to analog). Further we need an encoding scheme that encodes digital data into analog

format, such as the very simplistic OOK (On-Off Keying) scheme or the more involved ASK (Amplitude Shift Keying), FSK (Frequency Shift Keying) and PSK (Phase Shift Keying).

OOK On-off keying modulation is the simplest modulation scheme among all types of signal modulation. In EM communications, on-off keying modulation is generally implemented in the base-band. The idea is to encode bits into presence of signal, for instance in Fig. 4.3, presence of signal marks a bit 1 and absence marks a bit 0. This is a specific case of a more general encoding called Amplitude Shift Keying (ASK).

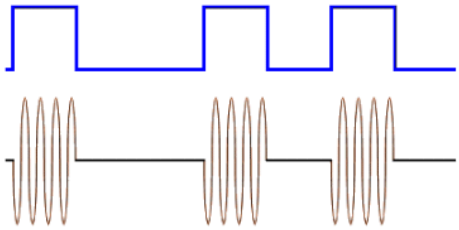


Figure 4.3:

FSK Frequency-shift keying (FSK) is a method of transmitting digital signals. The two binary states, logic 0 (low) and 1 (high), are each represented by an analog waveform. Shown inFig. 4.4 Logic 0 is represented by a wave at a specific (higher) frequency, and logic 1 is represented by a wave at a different (lower) frequency.

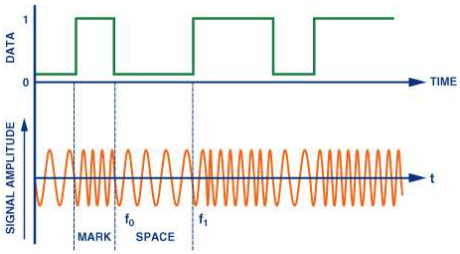


Figure 4.4:

PSK Phase-shift keying (PSK) is a method of digital communication in which the phase of a transmitted signal is varied to convey information. There are several methods that can be used to accomplish PSK.

The simplest PSK technique is called binary phase-shift keying (BPSK), and is depicted in Fig. 4.5. It uses two opposite signal phases (0 and 180 degrees). The digital signal is broken up timewise into individual bits (binary digits). The state of each bit is determined according to the state of the preceding bit. If the phase of the wave does not change, then the signal state stays the same (0 or 1). If the phase of the wave changes by 180 degrees – that is, if the phase reverses – then the signal state changes (from 0 to 1, or from 1 to 0).

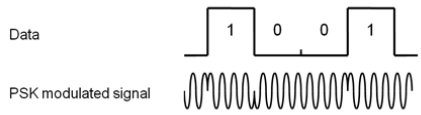


Figure 4.5:

4.1.3 RFI performance

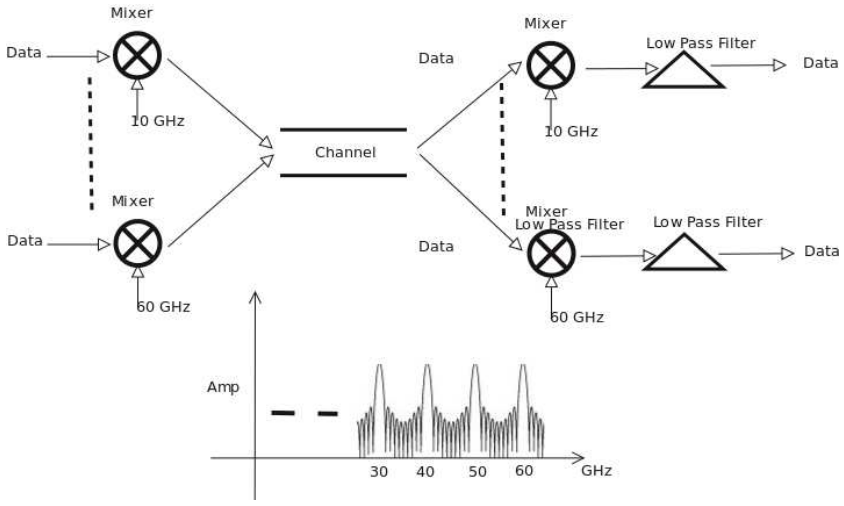


Figure 4.6:

RFI needs to justify the efforts by showing improvements in both performance and scalability over the incumbent wired technology that is the repeated wire, or its multi-bit extension.

In case of wires, the resistance and capacitance (RC) product of these wires introduces considerable delay in rise and fall of signals, which is intolerable in a mm-length wire setup. Since the data rate per wire is limited, in order to achieve high data rate, a wide bus is required. Scaling helps, allowing a higher density of wires, at the obvious cost of linear increase in the number of repeaters. Researchers in[7] show that the improving efficiency of repeaters (again, owing to the down-scale) is more than compensated by the increase in number of repeaters to ensure a decent bandwidth. At a projected 1 million plus repeaters, the energy hungry design becomes unwieldy.

In contrast, RFI performace is expected to increase with scaling. When using traditional voltage signaling, the entire length of the wire has to be charged and discharged to signify either '1' or '0', consuming much time and energy. In RF-I an electromagRFInetic carrier wave is continuously sent along the transmission line instead. Data is modulated onto that carrier wave using amplitude and/or phase changes.

As discussed previously, it is possible to improve bandwidth efficiency of RF-I by sending many simultaneous streams of data over a single transmission line. This is referred to as multi-band (or multi-carrier) RF-I[8]. In multi-band RF-I, there are N mixers on the transmitting (or Tx) side, where N is the number of senders sharing the transmission line (Fig. 4.6 uses N=6). Each mixer up-converts individual data streams into a specific channel (or frequency band). On the receiver (Rx) side, N additional mixers are employed to down-convert each signal back to the

original data, and N low-pass-filters (LPF) are used to isolate the data from residual high-frequency components. This method comes from standard wireless networks paradigm - FDMA. Leveraging the over 100 GHz in bandwidth available, we can modulate the net frequency-bandwidth into several high frequency carriers giving us unprecedented aggregate on-chip data-bandwidth[9].We can take a step further and introduce dynamic allocation of transmission frequencies. The distribution can be done on a plethora of grounds, including application bandwidth requirements, priority etc. This would ensure a high utilization of the huge raw aggregate data-bandwidth along with introducing a hook whereby we can control and ensure the QoS features of our chip.

Also the data rate is enhanced because of a couple of reasons. Firstly, the baseband data rate is limited by the high attenuation of high frequencies compared with DC due to wire resistance. At higher frequencies, required for higher data rates, we require equalization hardware to make up for the variations in losses from DC to the maximum frequency. Upon , modulating our original wave onto a carrier frequency the resulting wave experiences a relatively less change in attenuation even at high modulation rates (since the data bandwidth to carrier frequency ratio $\frac{\Delta f}{f_c}$ is much small than at baseband). This in turn simplifies the required equalization, saving power and area.

RF-I can allow signal transmission across a 400 mm 2 die in 0.3 ns via propagation at the effective speed of light [7] as apposed to less than or equal to 4 ns on a repeated bus.

4.2 Photonic Interconnects

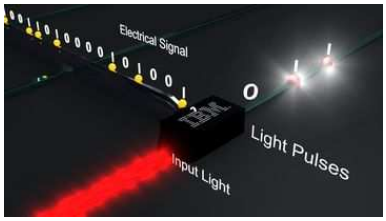


Figure 4.7:

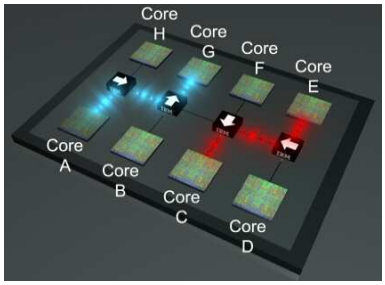


Figure 4.8:

Courtesy: IBM

While RF does provide low latency, it suffers from low bandwidth density, relatively large components and electromagnetic interference. Nanophotonics, tries to improve upon it by providing high bandwidth density, low latency, and distance-independent power consumption, which make it a promising candidate for future NoC designs.

4.2.1 Photonic Interconnect components

4.2.1.1 Laser source

As in the case of RFIs, a primary signal source is necessary that can be further modulated to use it to transfer data. In the case of photonic interconnects, this happens to be the laser source. As depicted in Fig. 4.7, the input light, coming from a laser source and input bits are translated into light pulses by the modulator. As is the case with majority of photonic signalling, it employs a classic example of OOK (discussed in sec. 4.1.2.3).

4.2.1.2 Waveguides

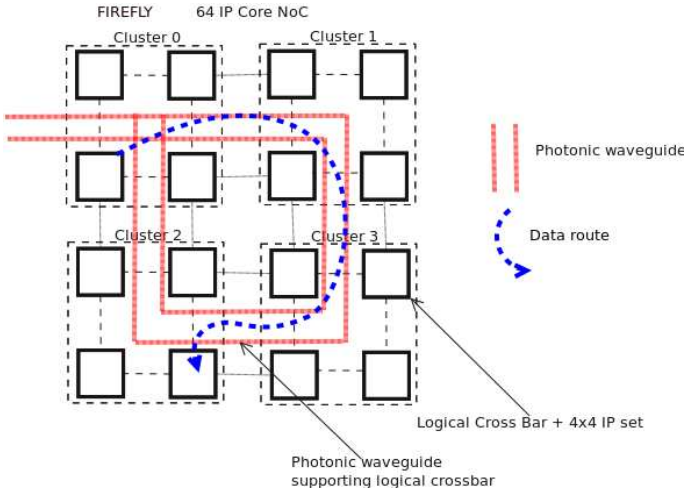


Figure 4.9:

Waveguides are required for for routing optical signals. Lasers of multiple wavelengths are fed from the laser source into a shared waveguide. Planar optical waveguides can be fabricated using Si as core and SiO_2 as cladding with transmission loss as low as 3.6dB/cm and good light confinement, allowing for sharp turns (radius of 2um) with minimal loss (0.013dB) [10] . With Dense Wavelength Division Multiplexing (DWDM) technique, lasers of different wavelengths can be transmitted within the same waveguide without in- terfering with each other. This allows for high bandwidth density and reduced layout complexity.

In [11], researchers propose a photonic interconnect design called Firefly. The Firefly topology (Fig. 4.9) uses clusters of $8x8$with each cluster further broken into IP sets

of $4x4$. The IP set might use photonic interconnect to send data to another IP set using one of the 64 wavelengths (generated by the laser source) employing DWDM we earlier cited . Firefly, in particular, creates a logical crossbar connecting all the 16 IP sets. The physical fabric for this logical network is the photonic waveguide compelmented by 64 wavelengths of light frequencies.

The Firefly, in turn, was inspired by the Corona architecture[12]. Corona exploits nanophotonics by using an all-optical crossbar topology. A 64 × 64 crossbar is implemented with multi-write-single-read optical buses. Each of the 64 buses or channels consists of 4 waveguides, each with 64 wavelengths and each channel is assigned to a different node in the network. Even though nanopho- tonics can transmit data at the speed of light, it also consumes considerable amount of energy in the form of static and dynamic power dissipation. Thus, Firefly uses nanophotonics only for long, intercluster links, while utilizing economical electrical signaling for local, intra-cluster links. Hence the total hardware and power consumption is reduced.

4.2.1.3 Ring Modulators

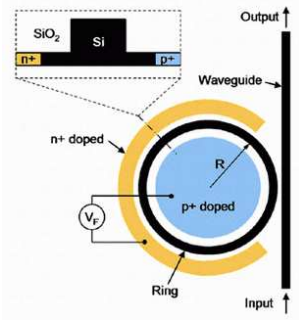


Figure 4.10: Courtesy: phys.org

Ring modulators are used for Electrical/Optical (E/O) signal conversion, resonant detectors for O/E signal conversion. A schematic representation of the component is shown in Fig. 4.10. V_F here controls the 'Output'. The resonant modulator modulates electrical signal onto a specific wavelength, which traverses the waveguide and is absorbed by the resonant detector of that specific wavelength. This modulation/detection process does not interfere with the lasers of other wavelengths.

4.2.2 Photonic Interconnect performance

As previously discussed, the RF suffers from an inherent lack of frequency-bandwidth density, essentially because of the low operational frequency-bandwidth. Photonic interconnects has all the benefits provided by RFIs, and improves upon its fallacies. Corona architecture[12] provides a gargantuan 10 terabits per second of

data-bandwidth , thanks to DWDM and the non-interfering characteristics of light waves.

4.3 Wireless Interconnects

However, nanophotonics has its own constraints. For example, unlike conventional electrical signaling, static power consumption constitutes a major portion of the total nanophotonic communication power.Nanophotonics also come with the additional energy cost for electrical to optical (E/O) and optical to electrical (O/E) signal conversions. Optical links face technological challenges such as design of efficient transmitter and receiver components, reliability of integrated light source and high manufacturing cost which prevents its commercial adoption. Thus, though the components of a complete photonic NoC, including dense waveguides, switches, optical modulators, and detectors, are now viable for integration on a single silicon chip ,these disadvantages hinder optical interconnections from becoming a feasible solution.

The latest in interconnects , wireless on-chip interconnects tries to assuage these issues and pave way for a low cost and high efficiency source of data-bandwidth. We dissect the paradigm in coming chapters. In [13], the authors show the supremacy of WiNoCs over the optical counterpart, with following improvements

Packet energy per unit Bandwidth (nJ/TBps) for various NoC architectures with increasing system of size

System Size	Flat Mesh	WiNoC - THz	WiNoC - SubTHz	RFNoC	Photonic NoC
128	560.0	21.0	31.3	34.1	34.1
256	721.7	13.3	19.6	25.8	29.6
512	1171	10.0	15.4	32.8	35.1

Area overhead ($\pm 3mm^2$) in mm^2 - 400 $mm^2$is the reference (chip area)

System Size	WiNoC - THz	WiNoC - SubTHz	RFNoC	Photonic NoC
128	20	55	30	50
256	45	90	60	120
512	90	140	110	410

5 Wireless NoC (WiNoC)

5.1 Overview

CMOS based wireless interconnection enables on-chip multi-hop communication by the embedding wireless channels. Fig. 5.1 shows a simplistic base case of wireless on-chip communication. One of the PE decides to send data to another. It forwards the data to the Digital to Analog Conversion unit (DAC) that is responsible to encode the digital data in analog format using one of the schemes discussed in sec. 4.1.2.3. Once the conversion is done, the analog data is forwarded to the transmitter that pushes the data onto the channel . Upon reception, the receiver transfers the data to Analog to Digital Conversion (ADC) unit, which in turn translates the analog signal into its constituent digital information. This digital data is conveyed to the recipient PE, thus completing the process of transmission.

The process of broadcast on the channel provides unprecendented flexibility to the setup and thus allows completely reconfigurable topologies supporting a wide range to routing strategies. Interestingly, a satisfactory implementation can be conceived using the current CMOS technology without the need of embedding a foreign technology. The wireless links serve as a high-bandwidth and low-latency replacement for the wired multihop connects.

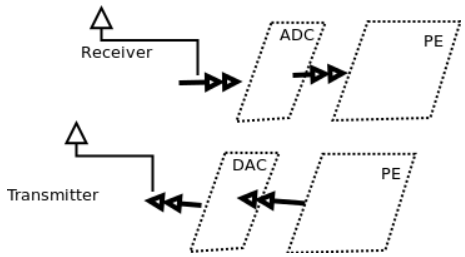


Figure 5.1:

Depending upon these factors the power involved in transmitting a bit (on average) can vary tremendously across chips. However, we can create a generic model.

$P_{wirelesslink} = (P_{transmitter} + P_{receiver} + P_{ADC} + P_{DAC})$

Please note that the baseband amplifiers , up/down conversion mixers etc. are included in the ADC and DAC black-boxes.

5.2 Internal Characteristics

5.2.1 Frequency

Firstly, we need to choose the bandwidth we want to operate on. This might vary from the tera Hz region to a few giga Hz. It is arguably the most important design decision, determining other aspects like design of wireless transceivers and peformance of physical layer. The WiNoCs primarily under the radar of current research can be classified into four broad groups according to the frequency in which they operate (depicted in Fig. 5.2).

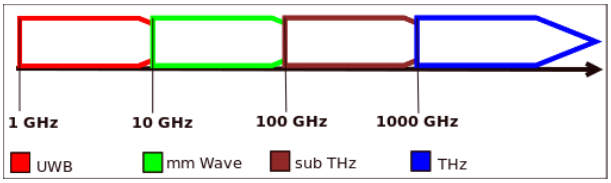


Figure 5.2:

5.2.1.1 Ultra Wide Band (UWB)

UWB lies at the lower end of the wireless spectrum that can be used on NoCs. This band has been well established in terms of feasibility of implementation by various groups, notably by [14], who utilize a carrier-free impluse radio-based UWB transciever to design a WiNoC. The transmitter is designed to generate the desired pulse with suitable driving strength so that signal can be efficiently radiated from an on-chip antenna. A CMOS integrated Gaussian monocycle pulse (GMP) generator creates an ultra short pulse that yeilds a very low power spectral density. The receiver constitutes of a wideband LNA, a correlator, an ADC, and synchronization circuits. It employs a 2.98-mm-long meander type dipole antenna that gives 1 mm data transmission range, whereas the transceiver can give a sustained data rate of 1.16 Gb/s on a channel with central frequency of 3.6 GHz.

5.2.1.2 mm Wave

As per the ITRS, for the 16nm CMOS technology, the cutoff frequency and unity maximum available power gain frequency are targetted between 600 GHz and 1 THz, respectively. This level of scaling assists in integrating the WiNoC units on chip as the units themselves are expected to scale proportionally. This also allows us to explore mm-wave as potential frequency candidate in WiNoCs. The on-chip antenna should provide us with the best power gain with minimal power and area requirements. [15] presents a metal zig-zag antenna as one of the most apt candidates for the purpose. [16] shows that a 0.38-mm antenna can achieve a 3 dB bandwidth

of 16 GHz with a central frequency of 57.5 GHz. [17] argues that the metal lines over the antenna doesn't affect the the gain factor, thus making it compatible with the current fabrication process.

5.2.1.3 sub-THz Wave

The minimum antenna size scales down with the wavelength. This makes sub-THz region all the more lucrative as it saves real estate.

[18] recommends that the antenna be put in a polyimide layer, deposited of the top silicon, such that most electromagnetic energy could be confined within this low-loss dielectric layer. It observes that our primary aim should be to minimize the radiation penetrating into the substrate since substrate loss is a major factor in determining chip performance. The low loss polyimide layer can be shown to constrain most of the signal within the layer, boosting the range of communication to a centimeter. Since it is possible to switch a CMOS transistor as fast as 500 GHz at 32 nm CMOS, this allows us to implement a large number of high frequency bands for the on-chip wireless network. Using a 32 nm CMOS process, empirically, there are expected be total of 16 available channels, from 100 GHz to 500 GHz, for the on-chip wireless network, and each channel can transmit at 10 to 20 Gbps, putting the raw aggregate data rate to upto 320 Gbps. Again, [18] predicts that only 1-2% of the power would be used in wireless communication with a simple ASK transceiver implementation.

The mm-wave frequency band can provide abundant bandwidth while not suffering from severe signal degradation, thanks to the short communication distances in the WiNoC. In [18], a noncoherent on–off keying (OOK) based transceiver is designed for an mm-wave WiNoC, OOK being selected as it allows relatively simple and low-power circuit implementation.

5.2.1.4 THz Wave

As observed before, increase in communication frequency is a win-win scenario. Antenna characteristics of carbon nanotubes (CNTs) in the THz/optical frequency range has been well investigated[19]. These antennas can achieve a bandwidth of around 500 GHz, whereas antennas operating in the millimeter wave range achieve bandwidths of tens of gigahertz. A higher frequency-bandwidth translates into a higher data rate. Certain specific characteristics of CNTs make them well suited for the purpose. Extremely low wavelength demands a very thin antenna structure for efficient transmission and reception. CNTs are naturally just a few nanometers across in diameter, what is virtually impossible to acheive using any other fabrication technique. Secondly, the virtually defect free CNTs show minimal surface loss and thus can detect weak signals very effectively.

5.2.2 Antennas

To ensure the high throughput and energy efficiency of the WiNoC, the transceiver circuitry has to provide a very wide bandwidth as well as low power consumption. In designing the on-chip mm-wave wireless transceiver, low power design considerations need to be taken into account both at the architecture and circuit levels.

5.2.2.1 Meander type dipole antenna

A meander type (linear) dipole antenna is one of the simplest kinds, made of two conductors separated by a small distance, with a center-fed driven element as shown in Fig. 5.3. The signal voltage is applied to the antenna at the center, between the two conductors. The characteristic waves produced by them is shown in Fig. 5.4.

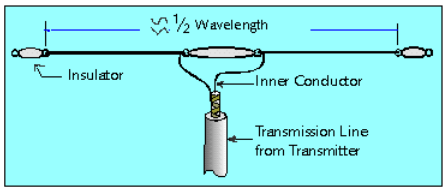


Figure 5.3: dipole antenna

These antennas are primarily used in the UWB based NoCs due to their simplicity and low hardware demands. Tab. 5.1 shows how the antenna characteristics are supposed to vary with the fabrication technology [20]. As apparent, the *energy/bit* scales proportionally with transistor size, whereas bandwidth scales inversely, making it a viable prospect for the future.

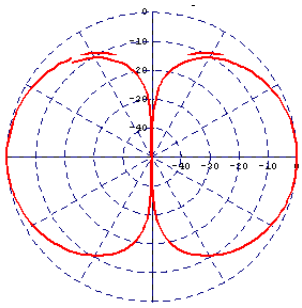


Figure 5.4:

Technology (nm)	90	65	45	32	22
Cut-off freq. (GHz)	105	170	280	400	550
Data rate per band ($Gbps$)	5.25	8.5	14	20	27.5
Dipole antenna length (mm)	8.28	5.12	3.11	2.17	1.58
Meander type dipole antenna area (mm^2)	0.738	0.459	0.279	0.194	0.14
Power (mW)	33	40	44	54	58
Energy per bit (pJ)	6	4.7	3.1	2.7	2.1

Table 5.1:

(courtesy: [21])

5.2.2.2 Zig-Zag antenna

A zig-zag antenna (Fig. 5.5) can be seen as a squashed helical antenna with planar geometry for ease in fabrication.As discussed previously, it has been often used for mm-wave communication to enhance the communication range over a centimeter. The viability in terms of implementation has been demonstrated in an early effort in[22] used for distributing clock signals. The signal frequency is 14.3 GHz, and the power incident to the antenna is 21 dBm. Zig-zag antenna is often implemented as a dipole antenna, with the zig-zag structure replacing the usual linear conductor.

5.2.2.3 CNT antenna

Due to a fundamentally different material and structures, CNT is quite different from the metal thin-wire antenna. The antenna performance such as gain can be adjusted by changing the conductivity of composite, while it is not possible for materials with fixed conductivity such as copper. As already discussed, CNTs are sleek and just a few nanometers across in diameter and they show minimal surface loss due to being defect-free. [23] reports that the bandwidth of a CNT composite single antenna beats that of copper antenna operating over the 24 to 34 GHz frequency range. It further shows the effectivity of CNT based antenna in the mm-wave region.[19] discusses its characteristics of low radiation resistance and efficiency in the THz range. It also recommends using a bundle of CNTs, claiming a 30 dB improvement in efficiency at 2 THz resonance frequency when using a rectangular geometry bundle. In [24], it has been demonstrated that semiconducting CNTs emit infrared photons when an electron-hole pair recombines across the band- gap. Thus, the CNT also acts as the signal generator, which reduces the transceiver complexity considerably.

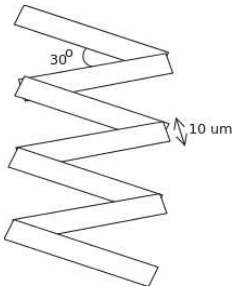


Figure 5.5:

Below is a summary of frequency-antenna combinations and their performance .

Frequency Range	Technology Node	Antenna Used	Modulation Scheme	Transmission Range	Data Rate	Energy/bit
UWB	180 nm	Meander type dipole	Pulse position modulation or biphase modulation	1 mm	1.16Gbps	Not available
mm-Wave	65 nm	ZigZag	Non-coherent OOK	20 mm	16Gbps	2.3pJ
Sub THz	32 nm	Not Available	ASK	10-20 mm	320Gbps	4.5pJ
THz	Not Applicable	Multi-walled CNT	Non-coherent OOK	23 mm	240Gbps	0.33pJ

(courtesy: [16])

5.2.3 Topology

5.2.3.1 Units

Node A common node on the NoC has three essential elements.

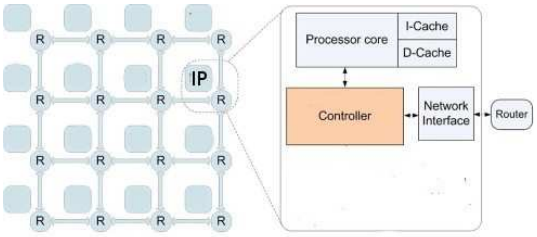


Figure 5.6:

- Processor - Consumes data and generates traffic.
- Router - Takes the routing decisions and maintains flow of packet data.
- Network Interface - Connects the router to IP core.

Link A link connects two routers. A router may have multiple bi-directional/uni-directional links, that may seperately correspond to data and control. Link is an abstract concept and can be established over any physical medium like wired, optical, wireless etc. Wired links cannot be reconfigured as they are hard-links. On the other hand, wireless links are flexible and can be established dynamically.

5.2.3.2 Qualifiers

Here we discuss certain desireable characteristics that a WiNoC may or may not posses.

Hybrid A purely wired network suffers from the issues that we have already discussed. But even having a completely wirless network is non optimal because not only does it bring in the inordinate overhead of excessive number of wireless transceivers but also actually run a performance deficit compared to wired network for short distances.

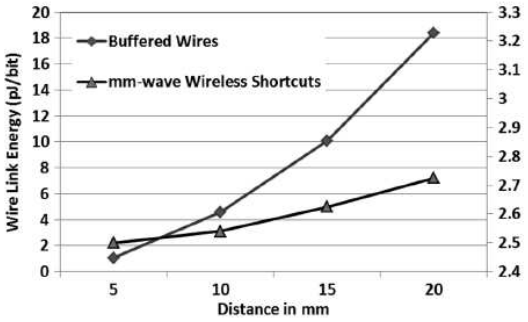


Figure 5.7: Courtesy:[16]

[16] calculates the 'tipping point' where the wireless shortcuts outdo their wired counterparts for mm-wave. Clearly, this happens only for distances greater than a certain value (here 7mm). So we would be better off using wired links for communicating to distances less than 7mm, and the opposite thereafter. This demands that we employ both wired and wireless links on our network and use the one that suits our purpose better. This ambivalence of paradigms is called a '**hybrid**' network.

The hybrid network, not only does better because it selects the best of two paradigms, but also because it can interleave the tasks between two distinct networks. For example, the wired network could be ear-marked for the high priority control packets, whereas the wireless network could be assigned the task of bulk data transfer. During the process of access contention, the nodes could contend for a channel and all the while the current channel 'owner' would transfer its data over the channel. By the time the current 'owner' is done , the next 'owner' would have been decided, which immediately gains control. This interleaving potentially reduces the contention overhead to zero.

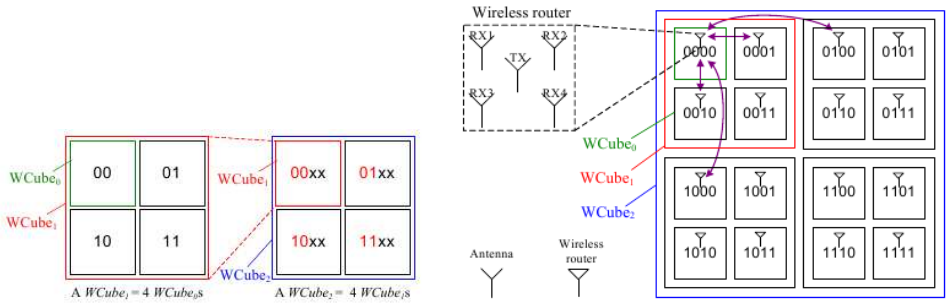
Because of these benefits, it is by default assumed that the presence of wireless network implies the presence of wired. So, we usually refer to the hybrid network as just "wireless" network.

Multichannel The high frequency paradigms at the high end UWB and beyond allows enough room to distribute the entire bandwidth into multiple channels of equal bandwidth seperated by guard bands. Such waves have characteristics that their signal has very short pulse duration and offer high data rate (C) with constant signal-to-noise ratio ($\frac{S}{N}$), achieved by increasing the bandwidth (B), following

Shannon's capacity equation ($C = B\,log_2(1 + S/N)$). This allows us to implement FDMA, which in turn helps carry multiple communications in parallel.

[21] proposes a Multiple Channel Wireless NoC (McWiNoC) using UWB. It disperses the RF nodes on-chip to form a "super-train" network. Utilizing the reconfigurability of wireless networks to configure the topology with variable communication range, it employs a channel arbitration scheme to grant multi-channel access and to resolve channel contention. [18] aruges about the viability of multiple non-overlapping channels in WiNoCs (see sec. 5.2.1.3).

Small World Small world networks are special networks that guarantees a very small path between any two nodes, even though the network is scaled up many orders of magnitude. Formally, the diameter of network (maximum path length between any two nodes) should not grow faster than $O(\,log(n)\,)$, where n is the number of nodes in the network.



An example of small world network is the WCube[18]. A WCube is recursively built 2D structure. A high-level WCube is composed of low level ones. The level of WCube $'n'$ $(n \geq 0)$ is mentioned in the subscript as $WCube_n$. $WCube_0$ is the most basic building block that is used to construct larger WCubes.It is represented by a single wireless router that is responsible for a cluster of m = 16 base routers of the baseline CMesh, thus comprising 4m nodes. An n-level WCube, incorporating the baseline $k - way$ CMesh (with a cluster of m base routers associated with a single wireless router), can support up to $2^{2n} \times m \times k$ nodes. Practical considerations force us to pin k at 4 and m at 16. This gives us the maximum number of nodes that can be supported as 2^{2n+6}. The diameter of this network comes out as $2n$, which ensures it's scalability even for a gargantuan size of network.

After the qualifiers that any topology could choose to conform to, we now see the various topologies that are widely used.

5.2.3.3 Bus topology

In the bus topology each node is connected to a single central wire. A signal from the source travels in both directions to all cores connected on the bus cable until it finds the intended recipient. The nodes that are not the recepient just ignore the data after

reading the packet or are pre-programmed by the controller to ignore it altogether. Alternatively, if the addresss matches the node address, the data is accepted. Since the bus topology consists of only one wire, it is rather inexpensive to implement when compared to other topologies. However, the low cost of implementing the technology is offset by the high cost of arbitration and management. Additionally, it can be a the single point of failure.

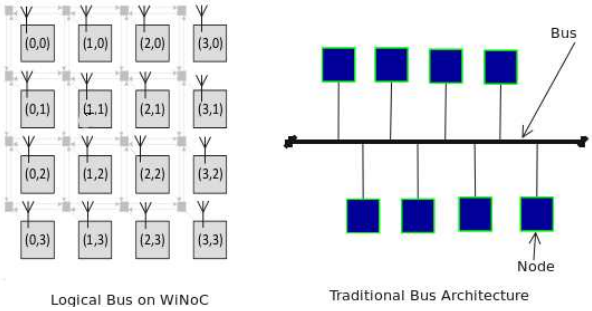


Figure 5.8:

In the context of wireless NoCs, the trivial implementation of a bus consists of nodes, each armed with a trasceiver pair and an antenna. The range of transmitter is high enough to contact any receiver on the chip. There is a single channel and all contend for it, with the selected node getting chance to transmit. It thus emulates a bus, providing a one hop transmission and a single channel.

5.2.3.4 Ring topology

A network topology that is set up in a circular fashion in which data travels around the ring in one/both direction(s) and each device on the on the path acts as a repeater to keep the signal strong as it travels. Each device incorporates a receiver for the incoming signal and a transmitter to send the data on to the next device in the ring.

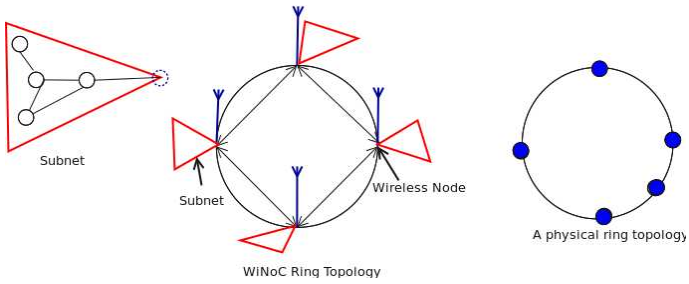


Figure 5.9:

In the wireless NoCs, ring structure is often found while implementing a hierarchical structure as depicted in the Fig. 5.9. A subnet has its own structure and works

using a wired network for local communication. For global data access, they use the wireless nodes that are arraged in a ring form and have access only to the immediately adjacent node.

5.2.3.5 Mesh topology

Mesh consists of nodes connected to its neighbors in a regular patter. The number of connections in a full mesh = $n(n-1)/2$.

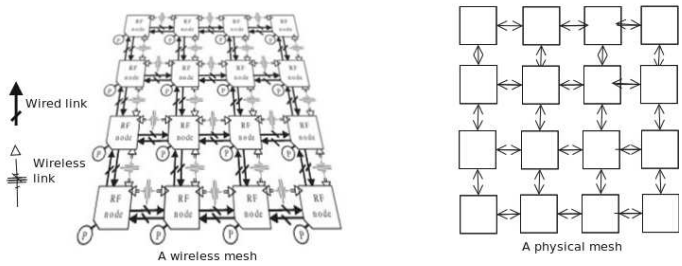


Figure 5.10:

Mesh is the most common topology found on the wired NoCs both because of it's effectiveness and ease of fabrication. In the wireless NoC paradigm, [16] discusses a noncoherent OOK based transceiver for implementing node to node mesh.

5.2.3.6 Star topology

In Star topology every node is connected to central node called hub or switch. The switch is the server and the nodes are the clients. The network does not necessarily have to resemble a star to be classified as a star network, but all of the nodes on the network must be connected to one central entity. All traffic that traverses the network passes through the central hub. The hub acts as a signal repeater. The star topology is considered the easiest topology to design and implement. An advantage of the star topology is the simplicity of adding additional nodes. The primary disadvantage of the star topology is that the hub represents a single point of failure.

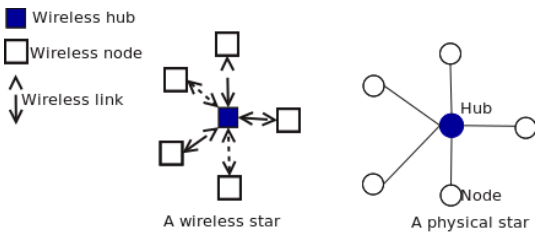


Figure 5.11:

5.2.3.7 Structure-hybrid topology

Sturcture-hybrid networks use a combination of any two or more topologies in such a way that the resulting network does not exhibit one of the standard topologies.It often involves using many heterogenous topologies in a hierarchical fashion interfaced with each other using wired or wireless links.

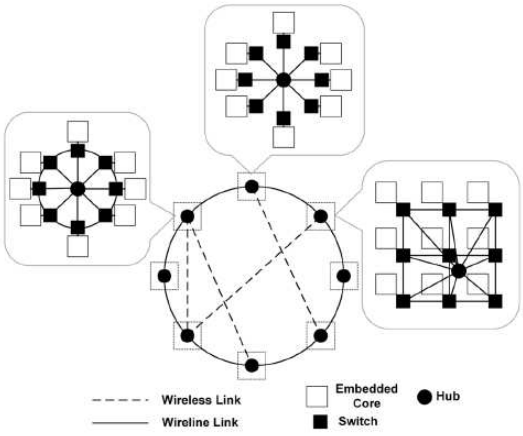


Figure 5.12:

As shown in [16], we reap the benefits of wireless links only if they are used at a certain distance. So the hybrid-structure networks try to connect local wired subnets using the low latency high bandwidth wireless links.

5.2.4 Media Access Control (MAC) protocols

Flow, vaguely is defined as a sequence of packets from source node to the destination node. In the wired NoC, flows contend for a specific router or link. But the scenario in WiNoCs is quite different owing to the single (handful of) channels available for communication. Also, since the system is distributed, synchronization is a difficult art to perfect. The sharing protocol needs to partition the bandwidth well across flows and nodes which are in transmissions range (called contention domains) of each other. Further we have to deal with issues like hidden-node and exposed-node problem. Contention can be broadly classified into two types:

Intra-flow contention is caused by nodes along the same flow contend among themselves for access to the wireless medium. A path could be seriously congested if the traffic source injects more packets into the traffic flow (especially for busty traffic) than the intermediate nodes can forward.

Inter-flow contention arises when multiple flows pass through the same region. The forwarding nodes of a flow encounter competition from not only their own flow but also other flows passing through them or their vicinity. The network will be

seriously congested if the source nodes inject more packets into the crossing flows than the nodes in these contention domains can forward.

5.2.4.1 Conflict based

CSMA/CA Carrier sense multiple access with collision avoidance (CSMA/CA) is a network multiple access method in which carrier sensing is used, but nodes attempt to avoid collisions by transmitting only when the channel is sensed to be "idle". However, this check doesn't ensure absence of collision or even detection (without use of explicit ECC codes, which is not a very likely prospect on a chip).

For removing the hidden terminal problem, one needs to implement a seperate RTS-CTS protocol which itself is too much of a temporal overhead, not to mention the additional circuit complexity that needs to be added. Exposed terminal problem is a still more difficult issue to solve, which if left that way, would eat into the effective bandwidth of the system.

Further more , the protocol fails to leverage the wired network lying underneath for improving the network performance. Thus, it is not a very strong contender as far as MAC protocols for on-chip networks are concerned.

SD-MAC Introduced in [14], it is a synchroized and distributed MAC protocol. It uses a binary countdown mechanism to resolve contention. The contending nodes pick up random number inside a range and forward it to the intended recipient. The receiver then operates upon the bits, and after a few rounds of shortlisting, arrives at the winner sender. It then proceeds to inform the sender that it may start the transmission. Thus, transmission is broken down into following four steps:

- **Initialize** : involves choosing the random number and associated operations.
- **Contention** : contend for the channel by conveying the randomly chosen number to recipient.
- **Access authorization** : see if the channel authorizes the request.
- If yes **Transmit.**
  - Else decrease size of range in which random number was chosen (this improves the probability of being selected the next time) and contend for the channel again.
  - The above step ensures fairness as the node contending for sufficient amount of time is guaranteed to get a turn after some time.

Interestingly, the fact that receiver selects the final sender completely alleviates the issues of hidden terminal and exposed terminal problem. Removal of hidden terminal is very important as it has a bearing on the net throughput. Similarly, removing exposed terminal issue improves the channel utilization.

SD-MAC utilizes two layers of communication - employing the wireless backbone for data transmission and wired network for control data. Synchronization is done through the faster medium - that is the wireless backbone. Competition for the next data transmission slot is interleaved with the current data transmission, this ensures that the temporal overhead of contention process is zero.

transmission slot (both on separate media - wired and wireless). INIP - sync + registration CCP - series of N slots

5.2.4.2 Conflict free

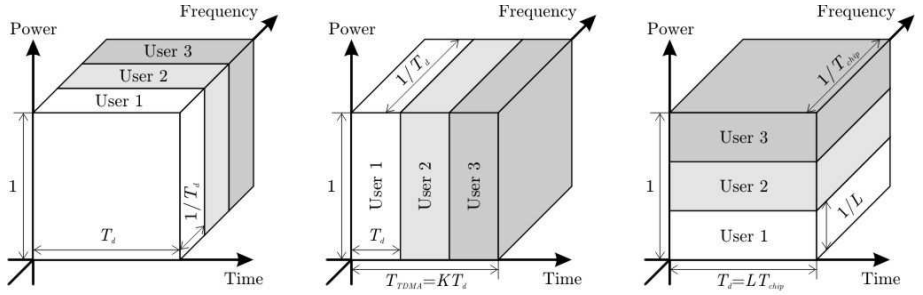


Figure 5.13: **FDMA** **TDMA** **CDMA**
Courtesy: Dr.-Ing. Erik Haas, German Aerospace Center

In the conflict free mechanisms, the nodes don't need to contend for effective data-bandwidth as it is pre-determined externally.

TDMA TDMA is a channel access method used for shared method for shared medium network. It can accommodate multiple users in a same frequency channel by dividing the frequency channel usage in time slots Fig. 5.13. In TDMA method the time axis is divided into a number of time slots, which are pre-assigned to different nodes. Each node has right to transmit freely during that slot assigned to it and all the system resources are devoted to it during that slot. Every such period is known as frame or cycle. In basic TDMA each node has exactly one slot in every frame. The more general TDMA schemes in which more slots are assigned to one node within a frame is referred to as generalized TDMA. For proper coordination of nodes in TDMA the nodes must be synchronized so that each node knows exactly when and for how long it can transmit. It is useful in the lower frequency regions like mm-wave, due to limitations in the available bandwidth and complexity of transceiver designs, having multiple non-overlapping channels is not an option.

There are various ways of achieving this coordination but few are viable on a NoC. One of those is the token based protocol.

[25] discusses the implementation of what is close to a token based TDMA. In this scheme, the particular wireless node possessing the token can broadcast flits into

the wireless medium. All other hubs will receive the flit as their antennas are tuned to the same frequency band. When the destination address matches the address of the receiving hub then the flit is accepted for further routing. It is routed either to a core in the subnet of that hub or to an adjacent hub. The token is released to the next wireless node after all flits belonging to a single packet at the current token-holding hub are transmitted. But the token passing protocol does not scale well with increasing number of wireless nodes.

However, the above discussed protocol differs from pure TDMA in one, it doesn't relinquish the token at the end of time slot , but at the end of packet transmission, second, the token passing delay eats into the time slot. Specifically, the token return period is bound to increase alongside the number of wireless nodes. This means that the time taken for acquiring the wireless medium by a node also increases. In order to solve these problems we need to look at other options.

CDMA CDMA utilizes orthogonal codes to allow multiple nodes to use the same channel at the same timeFig. 5.13. The idea is to have one from a set of orthogonal codes available at each of the nodes. While sending a packet, the sender node multiplies the data with the chip-code of the receiver. This encoded data thus created becomes a noise (that is automatically eliminated) to every node other than the receiver. On receiving the data, the receiver multiplies data with its chip-code and gets the original data back.

[26] uses a metal zig-zag antenna, OOK modulator/demodulator alongside CDMA implemented using 8 bit Walsh Codes (7 unique orthogonal codes). The ADC and the wireless ports add area overheads to the NoC switches with wireless capability. They report the total area overhead per wireless switch to be 1.79mm^2 resulting in a total overhead of 12.53 mm^2.

FDMA Using the FDMA channel access scheme multiple nodes can access a communication channel simultaneouslyFig. 5.13. In FDMA the whole frequency band is divided into the sub bands, and every sub band serves a single user, so every node has its own transmission frequency band and a receiver band. Guard bands are inserted between these frequency bands to prevent signal from one band to 'bleed' into another.

When operating sub-THz region, the net bandwidth is sufficient to be split into multiple bands. [18] partitions the entire bandwidth into 16 distinct channels that can be used simultaneously.

5.2.4.3 Priority based

Prioritized Fast Forwarding The concept has been discussed in [27].Fast-forwarding attempts to forward a packet once it is received . In this way, we can prevent source

node from injecting new packets before the previous packet of the same flow traverses out of its contention domain. Thus we propose a prioritized forwarding scheme, receiver priority , which can be easily incorporated into the SD-MAC protocol by assigning prioritized contention numbers for binary countdown. The basic idea is to assign higher priority to the downstream node of the same flow. We reserve the MSB as the receiver priority bit, '1' indicating tha t the current node just receives a packet; '0'

5.2.5 Routing protocols

5.2.5.1 XY Routing

XY routing is one of the most primitive and standard routing algorithms that is both shortest path (for wired mesh networks) and deadlock free. Its amazingly simple implementation means it consumes minimal hardware and computatational power. Due to this very property, it is often the protocol of choice in real life NoCs. However, it suffers from acute load-imbalance and latency issues becuase it doesn't take congestion into account. In wireless routing, XY is no longer the shortest path because of the non uni-dimensional (diagonal) paths available (as shown in Fig. 5.14).

5.2.5.2 Location Based Routing (LBR)

As the name says, LBR uses the source and destination nodes to decide the next node in route.[21] has developed LBR as an extension of GAR (Geographically Aided Routing) introducing lower latency and deadlock avoidance. It uses LBR in a multichannel environment using UWB-I. Basically it uses the idea that the range of a wireless transceiver can go beyond the one hop that wired links can connect. After the wireless nodes are placed on the chip and their trasmission range is determined, we get a McWiNoC with static yet configurable topology. GAR essentially uses the relative coordinates of nodes to decide the route, as does LBR.

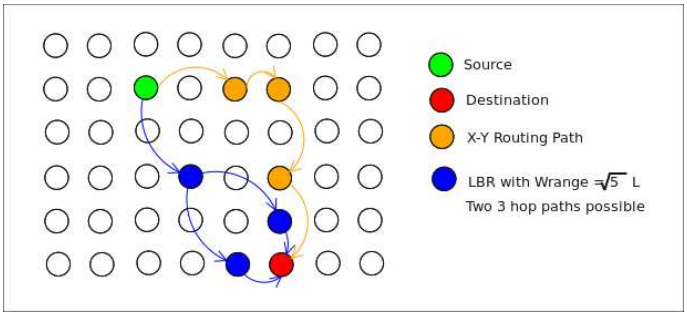


Figure 5.14:

5.2.5.3 Region Aided

[28] uses a cross layer design (similar to the OSI) for data transmission. The responsibilities of addressing and routing are vested in the network layer. It employs a hierarchical address scheme where "logical address" is calculated as $< CoreID, internalID >$ where $CoreID$ is ID of the IP whereas the internalID is address of the memory location or register inside core. In the adaptive region aided routing that the design uses, each RF node divides the chip plane from it's own perspective. Each region has region header and four border coordinates define a region. It uses region delimitation for routing which in turn can be split into 3 steps

- Region pre-identification - Best routing path is calculated based on metrics hop count, expected transmission time and path interference.
- Mutual overlapping elimination - If the nodes in the overlapping area belong to only one of the overlapped regions, simple overlapping occurs. Otherwise, if the nodes in the overlapping area belong to different overlapped regions, mutual overlapping occurs. Eliminating these overlapping regions makes sure the regions under consideration are mutually exclusive.
- Border Extension - The region border can be extended to reduce the hardware cost without hurting the routing decision.

5.2.5.4 Adaptive routing

Adaptive routing has been shown to outdo the static ones in most general purpose scenarios where the traffic is not well characterized and predictable.

The adaptive region aided routing by[28] has already beenLocation Based Routing (LBR) discussed.

[29] proposes an adaptive wormhole packet switching based routing. Packets are composed of 64-bit flits and each individual packet may be transmitted using wireless or wired or both. This decision is taken on the basis of location of source and destination nodes. The routing algorithm uses the wired links for nodes lying in the same subnet. The algorithm also takes into account the buffer status. Combining all these metrics, routing algo is adaptive with a parameter ∂ that can be altered as per traffic pattern and congestion that are calculated dynamically using these metrics.

5.2.5.5 Deadlock avoidance

[29] avoids a deadlock by using virtual circuits (VCs). There are two possibilities when using VCs

1. Single channel - only way is that routing algo takes care that deadlock is avoided.

2. Channels > 2 - reserve 2 channels , for deadlock avoidance and allocate the rest as shared channels to improve performance.

[21] on the other hand , a uses turn restricted LBR to prevent deadlocks.

5.3 Traffic scenarios

5.3.1 Burstiness

Burstiness is a measure of the granularity of the traffic. Bursty traffic tends to pour in all at once, whereas the uniform traffic is it's well behaved counterpart. Communication data traversing an interconnection network exhibits temporal variance. In other words, the level of burstiness or packets per unit time is not constant and varies from one cycle to the next.Uniformity in traffic makes predictions possible and optimizations effective. Thus ability to deal with bursty traffic is a desired trait in networks.

Consider two pieces of code.

```
F1():
    loop(1M times):
        compute()
    loop(1M times):
        message()
```

```
F2():
    loop(1M times):
        compute()
        message()
```

The traffic generated by F1() is much more bursty as it initially buffers all the computed data and then messages all at once. If the data is being sent through a heavily contended route, it is very likely to stall. In contrast, F2() interleaves it's computation with the message passing.

The factor of 'burstiness' tries to model the frequency and size of these bursts. Effectively, it tries to reflect the on-chip traffic self-similarity where traffic burst patterns repeat themselves over time. Burstiness modeling uses the Hurst parameter, $0.5 < H \leq 1$, which defines the level of self-similarity. The closer H is to 1 the higher the level of burstiness.

Brewer, in [30, 30]introduced the concept of "bandwidth matching". To avoid overrunning a slow receiver, the sender deliberately limits its injection rate. This is an example of an application's effort of trying to smooth the burstiness to achieve better throughput. As shown in the paper, the results can be quite dramatic, and the network's bandwidth saturates at a much higher level of traffic when bandwidth matching is employed.

5.3.2 Distribution

Traffic distribution tells about the probability that a specific node may send a data packet to some other node. This (for a simplistic scenario of temporally invariant probability) can be summarized using a traffic matrix.

$$\nabla = \begin{pmatrix} \nabla_{0,0} & \nabla_{0,1} & \cdots & \cdots & \nabla_{0,n-1} \\ \nabla_{1,0} & \nabla_{1,1} & \cdots & \cdots & \nabla_{1,n-1} \\ \vdots & & & & \vdots \\ \vdots & & & & \vdots \\ \nabla_{n-1,0} & \cdots & \cdots & \cdots & \nabla_{n-1,n-1} \end{pmatrix}$$

where $\nabla_{i,j}$is the probability of node i sending a packet to node j in each cycle. For an NoC $\nabla_{i,i} = 0 \,\forall\, i \in \{nodes\}$ i.e. the node doesn't send packets to itself. It is apparent that the probability of packets originating from $node_i$ is the sum of elements in row_i. This value $\alpha_i \leq 1$ and is called the **packet injection rate** for $node_i$. $\alpha_i$is a measure of the traffic generated by $node_i$.

5.3.2.1 Random

In random distribution of traffic, every node has a uniform probabibility of sending packets to any other node.

So , $\nabla_{i,j} = 1/n - 1 \,\forall\, i \neq j$

This is probably the most famous distribution and is used in some way or other in every model. The issue with it is that the uniformity tends to distribute the load pretty evenly. Thus, even routing algorithms or topologies with poor load balancing acumen perform well in this distribution providing misleading benchmark numbers. To avoid this, another paradigm of synthetic traffic is used called permutation traffic.

Permutation traffic is often used to stress a topology/routing algorithm. Its primary idea is to generate flows between pairs of nodes. So, for a node s, its destination d can be calculated using some given function π, i.e. $d = \pi(s)$. These are considered a better parameter to benchmark a topology/routing algorithm because they focus on node-node pair traffic which yields more realistic distributions than random traffic.

5.3.2.2 Bit complement

Bit complement is a permutation traffic distribution, with π acting on each bit of a binary source address.

Say the address of source node in binary format is $[s_0, s_1, ..., s_{i-1}]$.

Then the destination d = $[\neg s_0, \neg s_1, ..., \neg s_{i-1}]$.

In the matrix ∇, this is marked as $\nabla_{s,d} = PIR_s$. (PIR is the packet injection rate of s)

5.3.2.3 Bit reverse

Bit reverse , just like Bit complement is a permutation traffic distribution, with π acting on each bit of a binary source address.

Say the address of source node in binary format is $[s_0, s_1, ..., s_{i-1}]$.

Then the destination d = $[s_{i-1}, s_{i-2}, ..., s_0]$.

In the matrix ∇, this is marked as $\nabla_{s,d} = PIR_s$. (PIR is the packet injection rate of s)

5.3.2.4 Matrix transpose

Another permutation traffic distribution, where a node that has a position $s = (i, j)$ in the NoC grid sends data to $d = (j, i)$.

As with the previous cases the matrix ∇, this is marked as $\nabla_{s,d} = PIR_s$. (PIR is the packet injection rate of s)

5.3.3 Hop distance

Average hop distance is a characteristic of traffic distribution pattern and determines the average manhattan distance between any source-destination pair.

6 A wireless simulator built upon wired NoC simulator.

6.1 Current Noxim Structure

Noxim is a Network-on-Chip (NoC) simulator developed by the engineers Maurizio Palesi, Davide Patti and Fabrizio Fazzino at the Università degli Studi di Catania (Italy).

The simulator is developed in SystemC: a system description language based on C++. User provides the following input configuration parameters:

- network size
- buffer size
- packet size distribution
- routing algorithm
- selection strategy
- packet injection rate
- traffic time distribution
- traffic pattern
- hot-spot traffic distribution.

Using the given parameters, the simulator then evaluates and outputs the following metrics:

- throughput
- delay
- power consumption

These metrics are available both at node and global level.

6.1.1 Units

6.1.1.1 Processor

The processor is the unit responsible for creating traffic. It takes in the parameters of PIR and traffic pattern and accordingly generates the packets. In systemC this

is implemented as a module **NoximProcessingElement**. It is directly connected with the router. The packets that reach the destination router are finally drained into the processor.

6.1.1.2 Router

The router is responsible for routing packets through the network following the parameters provided by the user. Particularly, the routing algorithm and buffer size are potent factors in deciding the network performance. 'Selection strategy' is used for fine-tuning the decisions when multiple options are available for routing the packet. It has been implemented as the module **NoximRouter**.

6.1.1.3 Tile

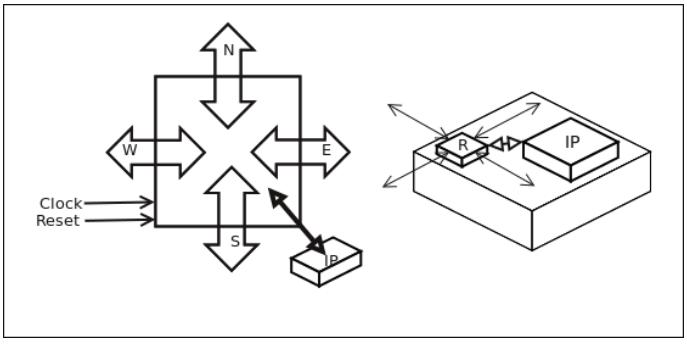


Figure 6.1:

A tile consists of the processor and router. The entire network is essentially tiles connected by links.

6.1.1.4 Links

Links are provided within the SystemC framework.

In Noxim, each of the interfaces has the following links going out/in.

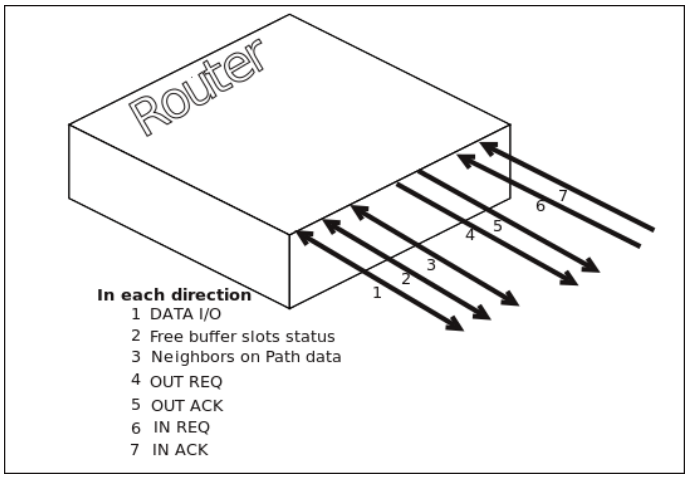


Figure 6.2:

1. Data I/O -
   a) the actual transfer to flits takes place through this line.
   b) Width = 1 flit size.
2. Free buffer slots status - t
   a) his information is used to optimize and choose between multiple routing options.
   b) Width = $\log_2(maxBufferSize)$
3. Neighbors on path -
   a) again used to choose between two routing path candidates
4. OUT-REQ
   a) used by the node to request a buffer slot in an adjacent node in order to begin transmission/forwarding.
   b) Width = 1
5. OUT-ACK
   a) used to notify the availability of free buffer an adjacent node soliciting a buffer slot.
   b) Width = 1
6. IN-REQ
   a) used by the node to receive a request of an adjacent node for a buffer slot in order to begin transmission/forwarding.
   b) Width = 1

7. IN-ACK

   a) used by node to receive the notification issued by the adjacent node regarding the availability of free buffer slot in it.

   b) Width = 1

6.1.1.5 NoC

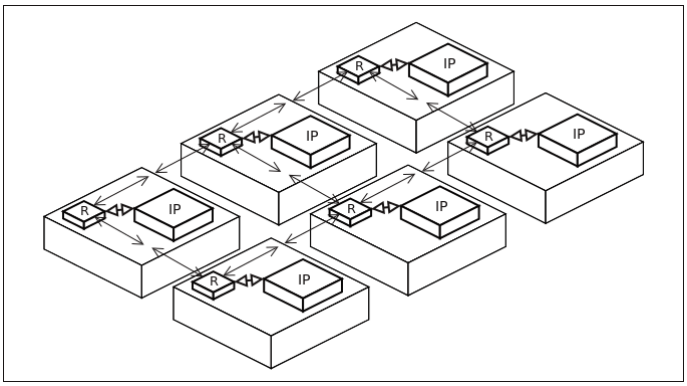


Figure 6.3:

Combining the links and the tiles, Noxim creates a mesh NoC as depicted in Fig. 6.3 - a 2x3 NoC.

6.1.2 Topology

Noxim, as it is, supports only a mesh topology.

6.1.3 Traffic scenarios

Noxim provides a wide range of traffic scenarios -

- Random
- Transpose
- Bit reversal
- Butterfly
- Shuffle

6.2 Adding a wireless layer

We currently lack a decent open source simulator to simulate wireless NoCs. There are a bunch of simulators that can take in the parameters characteristic of wireless links and apply it to wired networks but the results are not likely to be satisfactory if one wants to get a deeper picture, and not just a high level result (which again might not be good enough).

In order to add a wireless layer to the Noxim simulator, we need to make some sweeping changes in the architecture. More so because the wireless model needs to take the fine grained asynchronous and analog nature of wireless communication. Right now I am developing a hard-wired 4x4 mesh simulator employing 4 cluster based wireless links as a proof of concept. Once the results are proven to be satisfactory, the idea can be generalized to get a NxN mesh simulator with arbitrary number of wireless nodes.

To start modelling the wireless layer the first entity we need to model is the channel in which the wireless nodes will operate.

6.2.1 Wireless channel design

Following is a SystemC model of the wireless router

```
SC_MODULE(NoximWirelessChannel) {
    sc_in<bool> reset;
    // Constructor
    SC_CTOR(NoximWirelessChannel) { channel_data = NULL; }
public:
    sc_mutex channelAccess;
    NoximFlit* channel_data;
    sc_event pushIntoChannel;
};
```

channelAccess is a mutex (model provided by the SystemC library) that any node needs to lock before accessing the channel. This is an idealistic model where we assume that all the nodes are somehow synchronized and getting a lock is possbile in the distributed environment.

Once the channel is locked by a wireless node for access, the node proceeds to write the data (flit) it needs to send on the media by making **channel_data** point to the data.

Finally, following the asynchronous model, when data is written onto the media, all other nodes should become aware of it. This is achieved by notifying the event **pushIntoChannel**. In the model, all the wireless nodes are asynchronously sensitive to this event. Upon being notified that some data has been written onto the

channel, the nodes read the data, and accept or reject it depending upon who the intended recipient is (this is done using the destination address on the flit).

6.2.2 Wireless node design

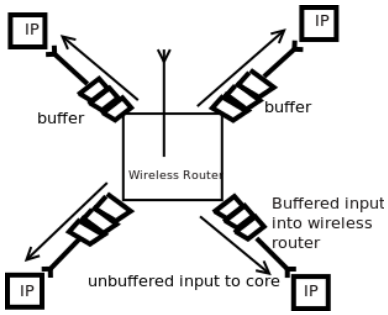


Figure 6.4:

Enunciating the structure (see the SystemC code that follows):

The buffer array **channel_buffer** stores the data coming in from the IPs. This is necessary because we are trying to interface a module that is being contended for with a traffic generator. On the other hand, we don't need buffers on the reverse link because the IPs are not being contended for. The wireless router can try to reach the IPs and will always find them free. In addition to these, the **wireless_buffer** holds the packets received through wireless network. The packets are then delivered to the intended IPs. Again the buffer is necessary as the nodes using wired links might not be able to keep up with the wireless link.

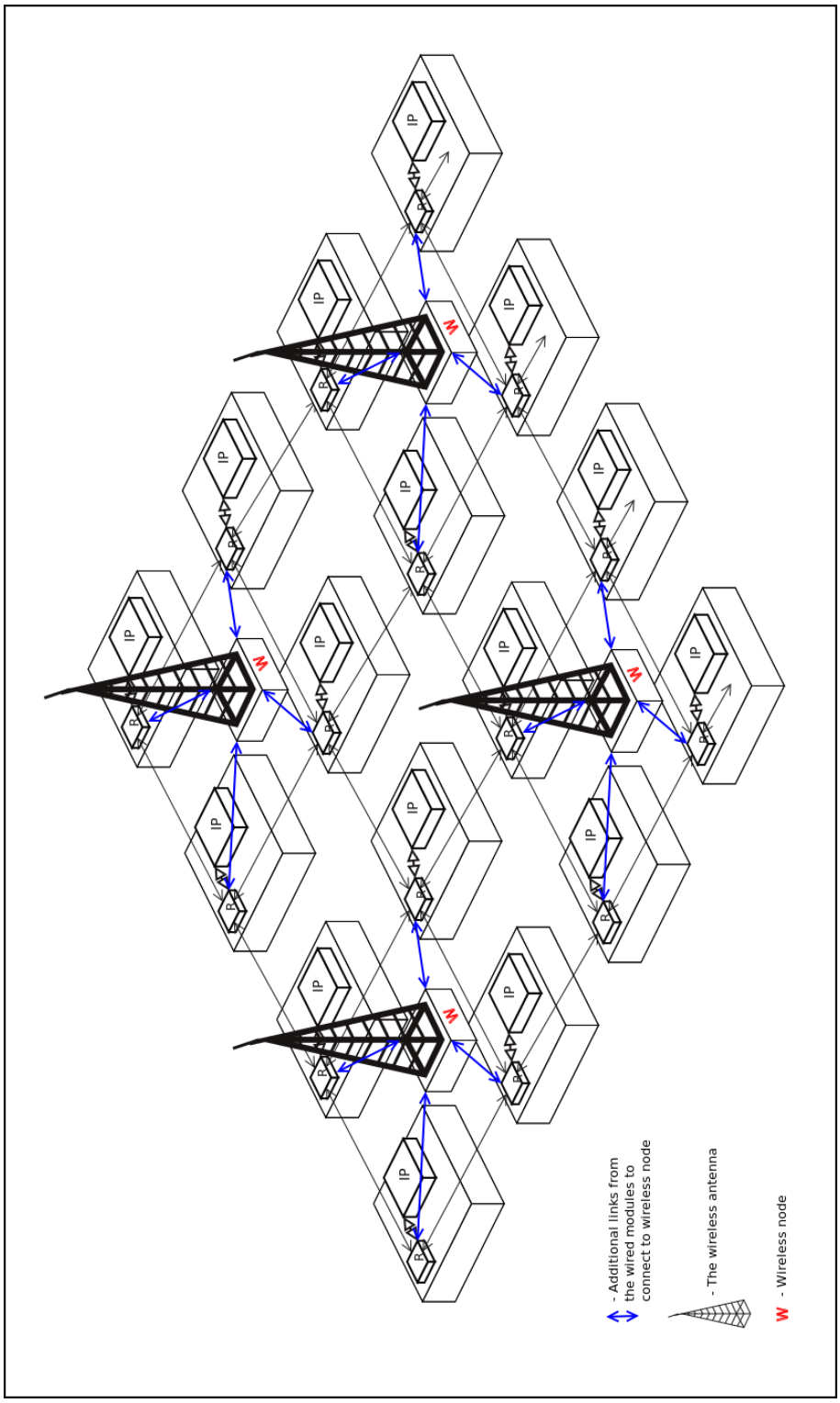
The I/O ports in this module are very similar to that of a wired router using the same set of clock,reset and REQ-ACK ports.

The constructor sets the sensitivity of router. The wired processes are sensitive to clock (clock triggered) while the wireless reception is triggered by someone writing a piece of data on the channel.

```
SC_MODULE(NoximWirelessRouter) {
/*internal functions have been removed to maintain clarity*/
// I/O Ports
// ====================================
    // The input clock for the router
    sc_in_clk clock;
    // The reset signal for the router
    sc_in<bool> reset;
    // The input channels
    sc_in<NoximFlit> flit_rx[WDIRS_Y][WDIRS_X];
    // The requests associated with the input channels
    sc_in<bool> req_rx[WDIRS_Y][WDIRS_X];
    // The outgoing ack signals associated with the input channels
    sc_out<bool> ack_rx[WDIRS_Y][WDIRS_X];
    // The output channels
    sc_out<NoximFlit> flit_tx[WDIRS_Y][WDIRS_X];
    // The requests associated with the output channels
    sc_out<bool> req_tx[WDIRS_Y][WDIRS_X];
    // The outgoing ack signals associated with the output channels
    sc_in<bool> ack_tx[WDIRS_Y][WDIRS_X];
// ====================================
// ====================================
// Registers
    // buffer for wireless reception
    NoximBuffer wireless_buffer;
    // Buffer for each input channel
    NoximBuffer channel_buffer[WDIRS_Y][WDIRS_X];
// ====================================
    sc_event signalInChannel;
    NoximWirelessChannel * channel;
// Constructor
// ====================================
    SC_CTOR(NoximWirelessRouter) {
// ——— Wired reception process
        SC_METHOD(rxWiredProcess);
            sensitive << reset;
            sensitive << clock.pos();
// ——— Wired transmission process
        SC_METHOD(txWiredProcess);
            sensitive << reset;
            sensitive << clock.pos();
// ——— Wireless reception process
        SC_METHOD(rxWirelessProcess);
            sensitive << channel->pushIntoChannel;
    }
};
```

6.2.3 Final view of the merged wired and wireless layer

Figure 6.5:



7 Conclusion

The wireless NoC paradigm has been analyzed in this report. Firstly the issues with the currently available wired networks were discussed. After that, the different solution paradigms were enumerated and compared, leading to the conclusion that WiNoCs hold the most promise for the future. Further we discussed the internal characteristics of WiNoCs like antennas, the operational frequency, topology , the MAC protocols and the routing protocols. We ended the section by discussing the external parameter - traffic and it's effects on the performance of WiNoC. In the last section we analyzed the contribution I am trying to make and the expected future work to be done for the successful completion of my BTP thesis.

Acknowledgments

I would like to thank Dr. Hemangee Kapoor for her continued support and guidance.

Bibliography

[1] N. Eisley and L.-S. Peh, "High-level power analysis for on-chip networks," in *Proceedings of the 2004 international conference on Compilers, architecture, and synthesis for embedded systems*, CASES '04, (New York, NY, USA), pp. 104–115, ACM, 2004.

[2] U. Y. Ogras, J. Hu, and R. Marculescu, "Key research problems in noc design: a holistic perspective," in *Proceedings of the 3rd IEEE/ACM/IFIP international conference on Hardware/software codesign and system synthesis*, CODES+ISSS '05, (New York, NY, USA), pp. 69–74, ACM, 2005.

[3] E. Bolotin, I. Cidon, R. Ginosar, and A. Kolodny, "Qnoc: Qos architecture and design process for network on chip," *J. Syst. Archit.*, vol. 50, pp. 105–128, Feb. 2004.

[4] P. Mohapatra, "Wormhole routing techniques for directly connected multicomputer systems," *ACM Comput. Surv.*, vol. 30, pp. 374–410, Sept. 1998.

[5] M. Kim, D. Kim, and G. Sobelman, "Network-on-chip link analysis under power and performance constraints," in *Circuits and Systems, 2006. ISCAS 2006. Proceedings. 2006 IEEE International Symposium on*, p. 4 pp., may 2006.

[6] S.-W. Tam, E. Socher, A. Wong, Y. Wang, L. D. Vu, and M.-C. Chang, "Simultaneous sub-harmonic injection-locked mm-wave frequency generators for multi-band communications in cmos," in *Radio Frequency Integrated Circuits Symposium, 2008. RFIC 2008. IEEE*, pp. 131 –134, 17 2008-april 17 2008.

[7] M.-C. F. Chang, E. Socher, S.-W. Tam, J. Cong, and G. Reinman, "Rf interconnects for communications on-chip," in *Proceedings of the 2008 international symposium on Physical design*, ISPD '08, (New York, NY, USA), pp. 78–83, ACM, 2008.

[8] M.-C. F. Chang, J. Cong, A. Kaplan, C. Liu, M. Naik, J. Premkumar, G. Reinman, E. Socher, and S.-W. Tam, "Power reduction of cmp communication networks via rf-interconnects," in *Proceedings of the 41st annual IEEE/ACM International Symposium on Microarchitecture*, MICRO 41, (Washington, DC, USA), pp. 376–387, IEEE Computer Society, 2008.

[9] J. Ko, J. Kim, Z. Xu, Q. Gu, C. Chien, and M. Chang, "An rf/baseband fdma-interconnect transceiver for reconfigurable multiple access chip-to-chip communication," in *Solid-State Circuits Conference, 2005. Digest of Technical Papers. ISSCC. 2005 IEEE International*, pp. 338 –602 Vol. 1, feb. 2005.

[10] Y. Vlasov and S. McNab, "Losses in single-mode silicon-on-insulator strip waveguides and bends," *Opt. Express*, vol. 12, pp. 1622–1631, Apr 2004.

[11] Y. Pan, P. Kumar, J. Kim, G. Memik, Y. Zhang, and A. Choudhary, "Firefly: illuminating future network-on-chip with nanophotonics," in *Proceedings of the 36th annual international symposium on Computer architecture*, ISCA '09, (New York, NY, USA), pp. 429–440, ACM, 2009.

[12] D. Vantrease, R. Schreiber, M. Monchiero, M. McLaren, N. Jouppi, M. Fiorentino, A. Davis, N. Binkert, R. Beausoleil, and J. Ahn, "Corona: System implications of emerging nanophotonic technology," in *Computer Architecture, 2008. ISCA '08. 35th International Symposium on*, pp. 153 –164, june 2008.

[13] S. Deb, K. Chang, A. Ganguly, and P. P. Pande, "Comparative performance evaluation of wireless and optical noc architectures.," in *SoCC* (T. BÃŒchner, R. Sridhar, A. Marshall, and N. Schuhmann, eds.), pp. 487–492, IEEE, 2010.

[14] D. Zhao and Y. Wang, "Sd-mac: Design and synthesis of a hardware-efficient collision-free qos-aware mac protocol for wireless network-on-chip," *Computers, IEEE Transactions on*, vol. 57, pp. 1230 –1245, sept. 2008.

[15] J.-J. Lin, H.-T. Wu, Y. Su, L. Gao, A. Sugavanam, J. Brewer, and K. O, "Communication using antennas fabricated in silicon integrated circuits," *Solid-State Circuits, IEEE Journal of*, vol. 42, pp. 1678 –1687, aug. 2007.

[16] S. Deb, A. Ganguly, P. Pande, B. Belzer, and D. Heo, "Wireless noc as interconnection backbone for multicore chips: Promises and challenges," *Emerging and Selected Topics in Circuits and Systems, IEEE Journal on*, vol. 2, pp. 228 –239, june 2012.

[17] K. K. O, K. Kim, B. Floyd, J. Mehta, H. Yoon, C.-M. Hung, D. Bravo, T. Dickson, X. Guo, R. Li, N. Trichy, J. Caserta, W. Bomstad, J. Branch, D.-J. Yang, J. Bohorquez, J. Chen, E.-Y. Seok, L. Gao, A. Sugavanam, J.-J. Lin, S. Yu, C. Cao, M.-H. Hwang, Y.-R. Ding, S.-H. Hwang, H. Wu, N. Zhang, and J. E. Brewer, "The feasibility of on-chip interconnection using antennas," in *Proceedings of the 2005 IEEE/ACM International conference on Computer-aided design*, ICCAD '05, (Washington, DC, USA), pp. 979–984, IEEE Computer Society, 2005.

[18] S.-B. Lee, S.-W. Tam, I. Pefkianakis, S. Lu, M. F. Chang, C. Guo, G. Reinman, C. Peng, M. Naik, L. Zhang, and J. Cong, "A scalable micro wireless interconnect structure for cmps," in *Proceedings of the 15th annual international conference on Mobile computing and networking*, MobiCom '09, (New York, NY, USA), pp. 217–228, ACM, 2009.

[19] Y. Huang, W.-Y. Yin, and Q. H. Liu, "Performance prediction of carbon nanotube bundle dipole antennas," *Nanotechnology, IEEE Transactions on*, vol. 7, pp. 331 –337, may 2008.

[20] D. Zhao and R. Wu, "Overlaid mesh topology design and deadlock free routing in wireless network-on-chip," *Networks-on-Chip, International Symposium on*, vol. 0, pp. 27–34, 2012.

[21] D. Zhao, Y. Wang, J. Li, and T. Kikkawa, "Design of multi-channel wireless noc to improve on-chip communication capacity!," in *Networks on Chip (NoCS), 2011 Fifth IEEE/ACM International Symposium on*, pp. 177 –184, may 2011.

[22] J. Branch, X. Guo, L. Gao, A. Sugavanam, J.-J. Lin, and K. O, "Wireless communication in a flip-chip package using integrated antennas on silicon substrates," *Electron Device Letters, IEEE*, vol. 26, pp. 115 – 117, feb. 2005.

[23] A. Mehdipour, I. Rosca, A. Sebak, C. Trueman, and S. Hoa, "Carbon nanotube composites for wideband millimeter-wave antenna applications," *Antennas and Propagation, IEEE Transactions on*, vol. 59, pp. 3572 –3578, oct. 2011.

[24] J. A. Misewich, R. Martel, P. Avouris, J. C. Tsang, S. Heinze, and J. Tersoff, "Electrically induced optical emission from a carbon nanotube fet," *Science*, vol. 300, no. 5620, pp. 783–786, 2003.

[25] S. Deb and K. . P. P. . B. B. . H. D. Ganguly, Amlan ; Chang, "Enhancing performance of network-on-chip architectures with millimeter-wave wireless interconnects.," in *Application-specific Systems Architectures and Processors (ASAP), 2010 21st IEEE International Conference on*, pp. 73 – 80, IEEE, 2010.

[26] A. Vidapalapati, V. Vijayakumaran, A. Ganguly, and A. Kwasinski, "Noc architectures with adaptive code division multiple access based wireless links," in *Circuits and Systems (ISCAS), 2012 IEEE International Symposium on*, pp. 636 –639, may 2012.

[27] Y. Wang and D. Zhao, "Distributed flow control and buffer management for wireless network-on-chip," in *Circuits and Systems, 2009. ISCAS 2009. IEEE International Symposium on*, pp. 1353 –1356, may 2009.

[28] Y. Wang and D. Zhao, "Design and implementation of routing scheme for wireless network-on-chip," in *Circuits and Systems, 2007. ISCAS 2007. IEEE International Symposium on*, pp. 1357 –1360, may 2007.

[29] C. Wang, W.-H. Hu, and N. Bagherzadeh, "A wireless network-on-chip design for multicore platforms," *16th Euromicro Conference on Parallel, Distributed and Network-Based Processing (PDP 2008)*, vol. 0, pp. 409–416, 2011.

[30] E. A. Brewer and B. C. Kuszmaul, "How to get good performance from the cm-5 data network„" in *Proceedings of the 1994 International Parallel Processing Symposium, Cancun, Mexico, April 1994.*