Vertical Link On/Off Regulations for Inductive-Coupling Based Wireless 3-D NoCs

Hao ZHANG†, Nonmember, Hiroki MATSUTANI†, Member, Yasuhiro TAKE†, Nonmember, Tadahiro KURODA†, and Hideharu AMANO†, Members

SUMMARY We propose low-power techniques for wireless three-dimensional Network-on-Chips (wireless 3-D NoCs), in which the connections among routers on the same chip are wired while the routers on different chips are connected wirelessly using inductive-coupling. The proposed low-power techniques stop the clock and power supplies to the transmitter of the wireless vertical links only when their utilizations are higher than the threshold. Meanwhile, the whole wireless vertical link will be shut down when the utilization is lower than the threshold in order to reduce the power consumption of wireless 3-D NoCs. This paper uses an on-demand method, in which the dormant data transmitter or the whole vertical link will be activated as long as a flit comes. Full-system many-core simulations using power parameters derived from a real chip implementation show that the proposed low-power techniques reduce the power consumption by 23.4%–29.3%, while the performance overhead is less than 2.4%.

key words: 3-D network on chip (3-D NoC), wireless, inductive-coupling, low power, on/off vertical link

1. Introduction

Three-dimensional IC (3-D IC) as well as System-in-Package (SiP) is an emerging research technology where multiple dies are stacked in a single package. It enables us to build custom systems by stacking necessary hardware components, such as processor chips, memory chips, and accelerator chips. Such flexibility of 3-D ICs is attractive compared to the conventional System-on-a-Chip (SoC) approach, in which all hardware components are integrated on a single chip. That is, hardware components of a 3-D IC can be customized by changing chips in a package according to the application requirements; for example, if the cache capacity of a 3-D IC system is not enough, we can replace the cache chip with a larger one.

Various interconnection techniques have been developed to connect multiple chips in a 3-D IC package: wire-bonding, micro-bump [1], [2], wireless (e.g., capacitive- and inductive-coupling) [3]–[6] between stacked dies, and through-silicon via (TSV) [3], [7] between stacked wafers. While many recent studies on 3-D IC architectures focus on the TSV that offers the largest interconnect density, inductive-coupling that can connect more than three known-good-dies without wired connections is another promising candidate. The inductive-coupling provides a large degree of flexibility to build the target 3-D ICs, such as adding, removing, and swapping the chips in a package after the chip fabrication, like building blocks.

Inductive-coupling is versatile and has been applied to various 3-D IC embedded systems, such as multi-core processors [8] and dynamically reconfigurable processors [9]. In addition, it has been applied to Network-on-Chips (NoCs) to extend the conventional 2-D NoC to 3-D [10]. A real wireless 3-D NoC chip, called Cube-0, has been developed in order to validate the concept. We will illustrate the Cube-0 system in Sect. 3.

Although the inductive-coupling itself is energy-efficient (e.g., 0.14pJ per bit [5]), inductors continuously consume a certain amount of power, regardless of packet transfers. That is, inductors waste significant power, especially when the utilization of vertical links (i.e., inductors) is low, which is a typical use case of 3-D ICs, in which most communications are within a chip while the communications between chips are not frequent. Although our wireless communication part of Cube-0 prototype can be applied to various applications, our first targets are embedded systems. Not like the systems with powerful processors and a large mount of cache, the power consumption of wireless links will occupy a significant portion of a total power and should be minimized for embedded applications.

To address this issue, in this paper, we propose dynamic on/off link control methods for wireless 3-D NoCs that stop the clock and power supplies to inductors according to the utilization. The proposed on/off link control methods can be used to optimize the Cube-0 system that employs a uni-directional ring for the vertical network. The details of Cube-0 are introduced in Sect. 3.

We also extend it to a bi-directional ring with full-duplex vertical links by duplicating the Cube-0 uni-directional ring. In addition, we propose a bi-directional ring with half-duplex links, which can significantly shorten the communication latency compared to the original bi-directional ring while the hardware amount for inductors is almost the same as the original one. We apply the on/off link control methods to these three networks and compare them with the original Cube-0 network in terms of the communication latency, power consumption, and hardware amount. The proposed methods are evaluated with a full-system Chip-Multi Processor (CMP) simulator using real parallel benchmark programs to confirm the efficiency of our proposal.

The rest of this paper is organized as follows. Sec-

Manuscript revised June 7, 2013.
†The authors are with the Graduate School of Science and Technology, Keio University, Yokohama-shi, 223–8522 Japan.
a) E-mail: cube@am.ics.keio.ac.jp
DOI: 10.1587/transinf.E96.D.2753

Copyright © 2013 The Institute of Electronics, Information and Communication Engineers
This paper surveys existing power optimization techniques for interconnection networks. Section 3 illustrates the prototype wireless 3-D NoC called Cube-0. Section 4 proposes the on/off link control methods for the uni- and bi-directional ring networks. Section 5 shows our evaluation results and Sect. 6 concludes this paper.

2. Related Work

In order to reduce the increasing power consumed in interconnection networks, various system level power management techniques have been proposed. Soteriou and Peh [11] explored the design space of on/off interconnection network based on a dynamic power management technique where network links can be turn on/off in response to bursts and dips in traffic with a distributes fashion. Kim et al. [12] proposed dynamic link shutdown (DLS) for cluster interconnection. It shuts off the power of down links when their utilization is below a certain threshold level, and a subset of highly used links can keep connectivity in the network. An adaptive routing strategy that intelligently uses a subset of links for communication was proposed, thereby facilitating DLS for minimizing energy consumption. Lee et al. [13] presented a variable frequency link for a power-aware interconnection network using the clock boosting mechanism, and applied a dynamic frequency scaling (DFS) to a 2-D NoC in order to judiciously adjust link frequency based on link utilization. However, since it is difficult to control the frequency and supply voltage of the transceiver and receiver used in the inductive coupling, traditional techniques around DVFS are difficult to be used.

Power gating techniques have been applied to on-chip routers [14], [15] so that standby power consumption can be reduced. In [16], PMOS power switches controlled by an ultra-cut-off (UCO) technique are inserted on each NoC unit to maintain minimum leakage in standby mode. In [17], more finer approach to divide and control the power domain is applied. Unlike DVSF, the power on/off techniques can be applicable for inductive coupling.

Although the techniques for sleep/wake-up is completely different between power gating and driver/receiver for inductive coupling, both circuit modules can be sleep mode for saving the power. Not like traditional 2-D NoCs or TSV based 3-D NoCs, in wireless 3-D NoCs, each chip is connected only with inductive coupling links and it is difficult to provide dedicated detective circuit for the power control using conventional technique. Thus, once a receiver is in the sleep mode, it cannot check the status of the corresponding transceiver until it is woken up by the signal from another link in the same chip. On the other hand, although it is possible to provide a dedicated network using inductive-coupling technique, unlike TSVs, since a certain power is required in the transmitter and receiver, various overheads will larger than the power saving by the power management network.

To the best of our knowledge, dynamic power management for inductive wireless 3-D NoCs which can address this problem has not been presented in other publications. Note that, the proposed methods can be applied to other types of wireless 3-D NoCs, such as capacitive-coupling based 3-D NoCs.

3. Cube-0: The First Trial of Wireless 3-D NoC

A prototype wireless 3-D chip, called Cube-0, which has on-chip routers and inductive-coupling data transceivers, was implemented on a 2.5 × 2.5 65nm CMOS chip. The purpose of this test chip is to find the implementation problems and explore the flexibility of the wireless stacking that enables us to add, remove, and swap the chips using a vertical ring network, but not for reducing power consumption.

3.1 Wireless 3-D NoC Implementation

In Cube-0, each chip has one pair of data transceivers and one pair of credit transceivers in pre-specified locations for enabling the vertical communication with other chips. As shown in Fig. 1, the chip is divided into four parts: a communication controller (1), point-to-point vertical downlink (2) and uplink (3) and a vertical shared bus (4).

The communication controller consists of two cores and two routers. The core has a packet generator and a packet receiver with a 45-bit packet counter. One router is connected to the downlink data transceiver and another for the uplink. Although these two routers are connected via bi-directional wired intra-chip link, only the top and bottom chips use one uni-directional intra-chip link to form an inter-plane vertical ring. Although a half part of Cube-0 is occupied by the vertical shared bus interface, we focus on the point-to-point vertical links for ring network in this paper. The details about the vertical bus is described in the paper [10].

Figure 2 shows the side view of four chips in a package. To place the transceiver and receiver of the neighboring two chips at the same location, each chip is placed with a certain gap. This space is also used for bonding of power and ground wires. On the right side of Fig. 2, Cube-0 chip architecture (top view) to show the details of chip organization.

The vertical links (downlinks and uplinks) consist of two pairs of Transmitter (Tx) and Receiver (Rx) channels. One Tx/Rx pair is used for 35-bit flit transfer, while another
pair for 2-bit credit back in the opposite direction for flow control. The wireless data are transferred serially using the doubled communication rate of a 4GHz local clock (LClk) shared by neighboring two chips. Thus, a Tx channel can transmit a 35-bit flit in each 200MHz system clock (SClk).

Vertical Bubble flow control [10] is also proposed for ring topology with virtual cut-through (VCT) switching to avoid deadlock. Each chip has a pair of data transceivers for uplink and downlink to form a ring. The data transceiver for uplink receives data from a neighboring lower chip and transmits the data to the neighboring upper chip. The downlink is used for the opposite direction of the uplink. It does not require any VCs but requires a single buffer with capacity of at least two packets for each input port. By limiting the packet injection not to consume all the buffer resources in a ring, packets on the ring continuously move without any deadlocks.

3.2 Vertical Link Implementation

The vertical links are composed of metal inductors which are capable of sending 8Gbps data. The serial clock frequency is up to 4GHz with double communication rate. Figure 3 shows a simple equivalent circuit of an inductive-coupling vertical link between stacked test chips. An inductor is implemented as a square coil with metal in common CMOS layout. The data modulated by a driver are transferred between two coils placed at exactly the same position of two stacked dies, and it is received at the other tier by the receiver.

A complete vertical link includes data transmitter (DTx), data receiver (DRx), credit transmitter (CTx), credit receiver (CRx), and LClk transceiver. The transmitters and receivers are configurable. There are two metal inductors at the two ends (i.e., neighboring chips) of vertical link respectively. When the vertical link works normally, a metal inductor should be configured to transmitter, the other one should be configured to receiver. In Fig. 4, we can see one end of vertical link is made up with metal inductor, Tx/Rx configuration module (Tx/Rx CM), and clock counter.

3.3 Power Consumption of Cube-0 Vertical Link and Problems

First, the power consumption of the target system (the vertical link shown in Sect. 3) is analyzed. SPICE simulation is used for evaluation, since it is difficult to analyze power in real chip. However, the total power of the simulation was almost the same value from a real chip. The power consumption of vertical link, on-chip router and communication controller are shown in Table 1.

In Table 1, “Active” represents the case when data is transferred through the wireless channel, while “Idle” shows the case there is no data on the channel.

<table>
<thead>
<tr>
<th>Component</th>
<th>Active</th>
<th>Idle</th>
<th>Sleep</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data Transmitter</td>
<td>4600</td>
<td>2500</td>
<td>0</td>
</tr>
<tr>
<td>Data Receiver</td>
<td>3500</td>
<td>9.2</td>
<td>0</td>
</tr>
<tr>
<td>Router</td>
<td>74.5</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Communication Controller</td>
<td>700</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

Compared with traditional works [18]–[20] using power gating, we can clearly find that the power consumption of vertical link of wireless interconnection is large. For example, in our second prototype Cube-1 [21], a MIPS compatible CPU and an energy efficient accelerator CMA [22] are connected with a inductive coupling ring network. Since the power consumption of the CPU and CMA is about 15mW and 20mW, respectively, about 32% of total system power is occupied by the wireless NoCs. This is due to the big gap between operating frequencies for routers and vertical links (200MHz and 4GHz respectively). Moreover, vertical links continuously consume a certain amount of power,
regardless of packet transfers. There is big waste to energy when the utilization of vertical links is low, which is a typical use case of 3-D ICs that the most communications are within a chip while the communication between chips are not frequent. For example, Fig. 5 shows ratio of idle time to application time in vertical links used in a full-system 3-D CMP simulator which will be introduced in Sect. 5.1. For the vertical links, 88.9% of the application execution time are idle in average.

“Sleep” mode is proposed in this paper. The transceiver is in the sleep mode by cutting of the bias voltage supply. Unlike the power gating for digital circuits, it does not require energy overhead except the power used in the controller. Although it requires a certain time for wake-up, it is within a clock cycle for system clock (200MHz) of the digital part. Also, compared with the power gating in digital circuits, there is little overhead for sleep down and wake-up of the vertical link.

According to the above analysis, it is essential to use sleep mode aggressively for reducing the power consumption of wireless link. However, the problem is that once the link is in the sleep mode completely, no data can be transferred since it is difficult to provide power control network between different chips. In the rest of paper, we propose methods for power control of wireless inductive coupling links.

4. Vertical Link Sleep Control Methods

First, the following three network configurations are introduced for wireless 3-D NoCs. Then, vertical link sleep control methods are proposed for these networks.

- Uni-directional ring network with simplex vertical links (URSL) is employed in Cube-0 which is the simplest case of our discussions in this paper. It is a single ring network where all data signals can flow in only one direction (counterclockwise or clockwise) through vertical links, as shown in Fig. 6 (a). That is, the configuration of inductors are fixed as a transmitter or receiver. This network has the lowest power consumption because only one pair of transceiver is used. However, the performance is worse than following two networks. The main reason is that data packets are transferred in one direction only, resulting in non-minimum packet transfers.

- Bi-directional ring network with full-duplex vertical links (BRFL) is an extension of URSL. It allows communication between neighboring chips in both direction simultaneously. The metal inductors are also pre-configured as a transmitter or receiver (i.e., they are not reconfigured at run-time). Thus, twice numbers of inductors used in the URSL are needed to form two uni-directional ring networks, as shown in Fig. 6 (b). However, twice the number of inductors undoubtedly will double the power consumption. Luckily, non-minimum routing is not introduced in this network. It has the highest performance among these three networks.

- Bi-directional ring network with half-duplex vertical links (BRHL) communicates for both directions, but only one direction at a time. That is, once one end begins receiving a signal from a transmitter, it must stay at the receiver mode while receiving the signal. Then, it can move to the transmitter mode. In Cube-0, the metal inductors can be reconfigured dynamically. It indicates that the vertical links can change their transmission direction like the red links, as shown in Fig. 6 (c). In this case, although non-minimum routing is not introduced because of its duplex property, transceivers need some clocks to be configured to opposite direction. That is, BRHL has worse performance than BRFL, but better performance than URSL. In addition, the BRHL consumes more power than URSL due to dynamic reconfiguration power consumption. However, it still consumes less power than BRFL, because dynamic reconfiguration power consumption is quite small compared with transceiver itself.

For these three networks, we propose sleep control methods that shut down the vertical links depending on their utilizations.
4.1 On/off Algorithm for Vertical Links

In Cube-0, no special method to save the power is applied. That is, a vertical link is naturally in the idle state except when data is transferred through it. As shown in Table 1, idle state still consumes a certain power to be saved. In this paper, this method is referred as a baseline in the evaluation.

Since it just stops the transceiver or receiver for the inductive coupling link, the state transition between sleep mode and active mode does not require energy overhead except the power consumption in the controller. It is like the clock gating used in digital circuits, and not related to power gating which requires energy overhead both for shutting down and activation. Moreover, the latency overhead of state transition between sleep mode and active mode is one clock cycle, same as the case between idle mode and active mode. As a result, we use sleep mode instead of idle mode which is employed in Cube-0. Note that, the analog part of inductive coupling link works at 4GHz while the digital part works at 200MHz clock. Although it takes a few clocks for stable working of the analog part, it can be available within one clock cycle of the digital part as the clock speed is much lower than that of analog part. All evaluations in Sect. 5 assume one clock cycle to activate inductive coupling links.

Two sleep modes are proposed here. Table 1 shows that the large part of power is consumed by the transmitter. So, if the transmitter for data (DTx) is shut down, a large part of power consumption can be saved. This mode is called partial sleep method (PSM). It only shuts down the data transmitter (DTx). Since other parts are running, the vertical link can be woken-up when the sender activates DTx in one clock cycle and sends the packets.

Another mode is the full sleep method (FSM). In the mode, the whole vertical link in the FSM including DTx, DRx, CTx, CRx, and LClk transceiver. Although the total analog part of wireless inductive link is shut down, the communication cannot be done until transceivers in the both chips are woken-up.

The sleep decision is based on the utilization of target vertical link and whether there are enough reachable paths to send wake up signals. We set a reasonable threshold utilization value first, then compare the link utilization within sample window (time slot) with the threshold when flit transmission completes. If the utilization is higher than the threshold, the state of the target vertical link is set from active to PSM. Otherwise, if there are paths to wake up the whole vertical, we set the state to FSM. If not, PSM is used again to only shut down the DTx. The decision to turn on a vertical link is basically the converse of the sleep mechanism and can be done as on-demand because of the property of fast on/off link transition. Such on/off control flows are summarized in Algorithm 1. The on/off regulation can be applied to above three ring networks. The details will be described in Sects. 4.3, 4.4, and 4.5.

Algorithm 1 Dynamic Vertical Link Shutdown Algorithm

```plaintext
1: if current_state == active & & idata arrives then
2:    continue to sample;
3: else if current_state == active & & idata arrives then
4:    compute the utilization Ul; compare with threshold Ul;
5:    if Ul >= Ul then
6:      shut down DTx; current_state ← partial sleep;
7:    else
8:      shut down the whole vertical link; current_state ← full sleep;
9:    end if
10: else if current_state == partial sleep & & idata arrives then
11:    current_state ← active; wake up the DTx;
12: else if current_state == full sleep & & idata arrives then
13:    if DRx current_state == sleep then
14:      send wakeup signal;
15:    else
16:      current_state ← active;
17:    end if
18: end if
```

4.2 Sleep Control Logic and Vertical Link Utilization

Three hardware modules are added for the proposed scheme: vertical link monitor, arbitration logic, and PSM/FSM manager. Figure 7 shows the diagram of the sleep control logic. Note that wakeup control logic is not required, because the on-demand method is used in this paper. As long as flit comes, the dormant DTx or the whole vertical link will be active without any judgment.

The vertical link monitor checks the state of the vertical link every cycle and computes the utilization of the vertical links between the neighboring chips depending on the size of sample window. Arbitration logic decides which vertical link (vertical links between the same neighboring chips are called neighbor links) can be shut down. If two request signals (from router_i and router_i±1 which are on layer i as shown in Fig. 7) are received, the arbitration logic decides which vertical link should be shut down depending on their utilizations. The vertical link which has lower utilization is prior to the one has higher utilization. If only one router sends the request, the arbitration logic gives decision signal to PSM/FSM manager directly. The arbitration logic is designed to avoid deadlock which leads neighbor vertical
links sleep forever. The reason will be explained in Sect. 4.3. PSM/FSM manager decides which policy to use, and sends the sleep signal. PSM or FSM will be used on the target links.

Formula (1) shows the vertical link utilization $U_{\text{vertical}}$:

$$U_{\text{vertical}} = \sum_{t = i}^{i + w} f(t)$$

where

$$f(t) = \begin{cases} 0, & \text{if no coming flit in cycle } t \\ 1, & \text{if flit comes in cycle } t \end{cases}$$

where $i$ is the start cycle of sampling, and $w$ is the size of sample window in terms of system clock (SClk).

4.3 Sleep Control Method for URSL

The PSM is advantageous when the vertical link utilization is high. For instance, in Fig. 8, when the red link utilization is higher than threshold, the sleep mechanism only shuts down the DTx immediately as long as no flit comes. The latency of sleep/wakeup is small, e.g., only one SClk cycle for Cube-0. As well as the fast vertical link shutdown, the vertical links also can be turned on within a very short time.

For example, when a flit demands on to traverse a dormant vertical link, the DTx can be powered on within one SClk cycle.

In the case of lower vertical link utilization, the FSM is introduced for reducing the power consumption of URSL. In this method, the overhead of long latency to sleep/wakeup whole vertical link can be compensated by the low utilization. Figure 9 shows how the sleep and wakeup mechanisms work for the situation of lower vertical link utilization. We decompose Fig. 2 into two figures. That is, Figs. 9(a) and (b) represent the side views of data links and credit links, respectively. The data and credit links have opposite directions. When the utilization of the red link is lower than a threshold, the sleep mechanism shuts down the whole red link. A sleep signal from layer 3 is sent to DRx and CTx which are located on the layer 2 via the red link one SClk cycle in advance. Thus, the whole link can be shut down after one SClk cycle. When we want to wake up the dormant vertical link, the green credit link (Fig. 9(b)) is employed for sending wakeup signal to the other parts except DTx. However, before shutting down the whole red link, the state of the credit transceiver, which belongs to the blue vertical link, must be checked. The blue vertical link is between layer 2 and layer 3 as well as the red vertical link. As mentioned previously, the rule which can avoid the deadlock is introduced: if the credit transceiver is power on, we can shut down the whole red link. Otherwise, we have to shut down the DTx only. If both the whole links (red and blue) are shut down, they cannot be woken up. The reason is that they use each other’s credit transceiver to wake up DRx, CTx, and LClk. That is, we cannot shut down the two vertical links which are between the same two layers simultaneously.

4.4 Sleep Control method for BRHL

The situation of BRHL is more complicated than that of URSL. We also analyze two cases: (1) high link utilization and (2) low link utilization. In this paper, the direction of vertical data link represents the one of whole vertical link. The direction of BRHL is configured to counterclockwise by default. In other words, all the vertical links are in the counterclockwise ring network if the clockwise link is not requested.

When the link utilization is high, PSM is also used here. The DTx is shut down by the sleep mechanism when no flit comes currently. As shown in Fig. 10, the red link is the target vertical link to be shut down. When the lower utilization of the red link is found, the DTx on layer 2 is powered off. If coming flit demands the red link, we can power on the DTx within one SClk cycle. However, unlike the case of URSL, the following three issues should be addressed.

1) How to configure the direction of the vertical links? As Fig. 10 shows, the direction of the green vertical link can be configured to clockwise (red link). Firstly, the green credit link is used to send a configuration signal to update the state of the DTx (CRx) to DRx (CTx). Then, DRx (CTx) can be configured as
DTx (CRx). Thus, the green link can be changed to the red one. The configuration must wait until all flits on the target vertical link are transferred even though the latency is increased. Since the default direction of the vertical link network is counterclockwise, the configuration of the vertical link becomes the default state automatically as long as the transmission on clockwise vertical link finishes.

2) Which links can be configured? For Cube-0 prototype system which has four chips, we assume that a flit is transferred from the source node R4 to the destination node R6 (Fig. 10). Thus seven hops are needed to reach the destination in the uni-direction ring network. In bi-direction ring, on the other hand, only a single hop is needed. Although the hop number is reduced, there is a trade-off between distance and latency, because several cycles are required to configure the target vertical link. Hence, for Cube-0, if there are less than three hops in the routing path in the clockwise ring network, we use clockwise vertical links to transfer flits. In general, zero-load latency of BRHL $T$ is calculated as

$$T = \begin{cases} T_{cw}, & \text{if the link direction is in default} \\ T_{cw}', & \text{if the link direction is clockwise} \end{cases}$$

where $T_{cw}$ is the latency for transferring a single flit from source node to destination node in the counterclockwise ring network, and $T_{cw}'$ is the latency for the clockwise ring network from the same source node to the same destination node. These values are calculated respectively as

$$T_{cw} = (H_{cw} + 1)T_{router} + H_{cw}T_{link}, \quad (4)$$

$$T_{cw}' = (H_{cw} + 1)T_{router} + H_{cw}T_{link} + H_{cw}T_{config}, \quad (5)$$

where $H_{cw}$ and $H_{cw}'$ are hop counts in counterclockwise ring network and clockwise one, respectively. $T_{router}$ and $T_{link}$ are latencies for transferring a header flit on a router and a vertical link, respectively. The routing logic uses the above formulas to decide which ring network to be used. If $T_{cw} < T_{cw}'$, the clockwise ring is selected for transferring the flits. Otherwise, the default ring network is still used to transfer flits.

3) About the competition: When two routers require the vertical link at the same SClk cycle, the router which uses clockwise ring network has higher priority than the one that uses counterclockwise ring. The rule is used in order to ensure smaller network latency as far as possible. That is, for instance, if router 4 and router 6 require the vertical link between layer 2 and layer 3, DRx on layer 2 can be configured to DTx as long as router 4 wants to use this link regardless of the link request from router 6.

When the vertical link utilization is lower than a threshold, the FSM shuts down the whole link. However, before shutting down the whole red link, we must check the state of credit link which belongs to the blue link as shown in Fig. 11.

For the wakeup mechanism, there are also two situations:

1) If the flit that comes from R6 (counterclockwise ring network) wants to use the red link, we can use the green credit link to wake up the dormant DRx, CRx, and LC\text{lk} (Fig. 11).

2) The flit that comes from R4 cannot wake up the red vertical link due to a latency overhead, because in this case the blue vertical link has to be configured. Hence, the flit should go through the counterclockwise.

4.5 Sleep Control method for BRFL

In the case of the BRFL, we analyze the PSM and FSM for two different utilization cases. The BRFL also has pre-configured transceivers. That is, after fabrication, the chip cannot configure the vertical link direction dynamically.

The PSM is used when vertical link utilization is low as well as the other two vertical link ring networks (i.e.,URSL and BRHL), so we omit detail of this method here. The difference is that, compared to URSL, BRFL provides more resources which can be used for controlling the sleep and wakeup. This can ease the sleep and wakeup control especially when the vertical network utilization is high.

There are two vertical links with reciprocal directions connected to the same two routers. We call them brother vertical links. The FSM can use one data link to control sleep/wakeup on the brother vertical data link. However, the difference between the BRFL and URSL is that the BRFL has one more path to wake up the whole vertical link. Both brother link and neighboring link can be used to transfer wakeup signal. Other behaviors are the same as those of URSL.

5. Evaluations

Firstly, experiment platform and method are introduced. Then, the BRHL and BRFL are compared with the URSL used in Cube-0 in term of application performance and power consumption in the situations with and without power optimization. Next, the power consumption and application performance of three ring network types are compared with those without power optimization. We also discuss the way how to decide the vertical link utilization threshold for each
benchmark programs.

5.1 Experiment Setup

A full-system CMP simulator, Simics [23] with GEMS [24] is used to verify our proposed methods. It is performed to measure the real application performance. Tables 2 and 3 list the processor, cache, memory, and network parameters, where $N$ is number of chips stacked. Proposed methods are evaluated in terms of application performance and the overall power consumption of 3-D wireless NoC.

Full-system simulations of the wireless 3-D CMPs that stack four chips which form a wireless ring network are performed to measure the real application performance. Each chip has one processor, four shared L2 cache banks, and two on-chip routers. Two memory controllers are connected to the bottom chip. The cache architecture is SNUCA [25]. We modified a detailed network model of GEMS, called Garnet [26]. Garnet is an interconnection network model inside a full-system simulation framework GEMS. The power parameters based on power analysis of real router and vertical link design (Table 1) are fed to the simulator. The result (real power consumption and benchmark program execution time) of each evaluation can be automatically output after simulation. Then we can get the power consumption results of vertical links before and after power optimization. We use five parallel programs from the OpenMP implementation of NAS Parallel Benchmarks [27]. Sun Solaris 9 operating system is running on the simulator. These benchmark programs were compiled by Sun Studio 12 and are executed on Solaris 9.

The most important contribution of this paper is to improve the energy efficiency of the wireless 3-D NoC over Cube-0 by means of introducing two sleep modes. So we quantitatively compare the proposed approaches with the conventional Cube-0 implementation, explained at the beginning of Sect. 4.1.

As shown in Table 5 and 6, for example, we can see the changes after using the proposed methods in the case of LU and IS, respectively. However, we can easily find the big gap between these two benchmark programs (e.g., the execution time running LU is as 7.5 times as IS’s on average). Hence, in the following sections, valuation results of power consumption and execution time are shown as relative values (i.e., percentage) instead of absolute values in order to show the effect of the proposed methods clearly. Please note that, moreover, “before using the proposed methods” means using conventional method.

5.2 Baseline Comparison among URSL, BRFL, and BRHL

Here, we compare the application execution time and network power consumption of the URSL, BRFL, and BRHL. The power consumption comparison among the URSL, BRFL, and BRHL is shown in Figs. 12 and 13. The power consumption (Y-axis) is normalized so that the power consumption using the URSL (before using the proposed methods) indicates 100% in Figs. 12 and 13. Meanwhile, execution time (Y-axis) of benchmark programs also using the URSL (before the proposed methods) is used as baseline in Figs. 14 and 15.

Figures 12 and 13 respectively show the network power consumption before and after optimization, when five benchmark programs (IS, LU, DC, MG, and BT) running on the full-system simulator. Meanwhile, Figs. 14 and 15 show the application execution time of the same benchmark programs before and after the optimization.

---

**Table 2** Processor parameters ($N = 4$).

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Processor</td>
<td>UltraSPARC-III</td>
</tr>
<tr>
<td>L1 I/D cache size</td>
<td>64 KB (line:64B)</td>
</tr>
<tr>
<td># of processors</td>
<td>4</td>
</tr>
<tr>
<td>L2 cache bank size</td>
<td>256 KB (assoc:4)</td>
</tr>
<tr>
<td># of L2 cache banks</td>
<td>16</td>
</tr>
<tr>
<td>Memory size</td>
<td>4 GB</td>
</tr>
<tr>
<td># of memory controllers</td>
<td>2</td>
</tr>
</tbody>
</table>

**Table 3** NoC parameters ($N = 4$).

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Topology</td>
<td>URSL, BRFL, BRHL ring</td>
</tr>
<tr>
<td># of routers</td>
<td>8</td>
</tr>
<tr>
<td>Router pipeline</td>
<td>[RC/VSA][ST][LT]</td>
</tr>
<tr>
<td>Flit size</td>
<td>128 bit</td>
</tr>
<tr>
<td># of message classes</td>
<td>3</td>
</tr>
<tr>
<td>Control packet size</td>
<td>1 flit</td>
</tr>
<tr>
<td>Data packet size</td>
<td>5 flit</td>
</tr>
</tbody>
</table>

---

**Fig. 12** Network power consumption before using the proposed methods.

**Fig. 13** Network power consumption after using the proposed methods.
As shown in Figs. 12 and 13, the BRFL consumes the largest power, and it exceeds the URSL by 11.7%-33% and the BRHL by 8.5%-28.5%, respectively. Regardless of the proposed on/off vertical link methods, the power consumption of the BRFL is the largest, while the URSL consumes the least power, because twice numbers of inductors used in the URSL are needed in the BRFL as mentioned in Sect. 4. The power consumption of the BRHL is slightly larger than that of URSL but much smaller than that of BRFL. The dynamic inductor configurations cause a certain amount of power consumption in the case the BRHL. That is, the total power consumption of BRHL is a little larger than that of URSL when a certain amount of inductors need to be configured dynamically. The application execution time of the URSL is the longest as shown in Figs. 14 and 15. It is much longer than the BRFL and BRHL. The BRFL and BRHL outperform the URSL by 10.6%-29.9% and 7%-25.8% respectively. To some extent, the BRHL has some advantages over the other two networks in the 3-D NoC.

5.3 Power Saving

The main purpose of the proposed vertical on/off link control methods is to reduce power consumption of the wireless 3-D NoC. We also estimated the power saving when the same benchmark programs run on the simulator as shown in Fig. 16. 23.4%-29.3% power saving is achieved. On average, 26.6%, 25.4%, and 26.8% power consumption can be saved in the URSL, BRFL, and BRHL, respectively.

5.4 Performance Overhead

Commonly, power saving techniques cause a certain performance degradation. The following equation shows the latency overhead to wake up a sleeping vertical link.

$$T_{\text{latency}} = T_{\text{router}}N_{\text{router}} + T_{\text{link}}N_{\text{link}} + T_{\text{activate}},$$

where, $N_{\text{router}}, N_{\text{link}}$ represent the number of routers, links traversals on average. $T_{\text{router}}, T_{\text{link}}$ correspond to the latency by transmitting wake up signal via one single router and one link. $T_{\text{activate}}$ is the activation time for circuit elements after receiving wake up signal. As shown in Eq. (6), the performance overhead consists of three parts. $T_{\text{router}}$ depends on run-time traffic situation. Meanwhile, we assume $T_{\text{link}}$ is one clock cycle. The first two parts depend on the number of routers and links by which wake up signals transfer, respectively. As mentioned in Sect. 4.1, we assume $T_{\text{activate}}$ is one clock cycle. It follows that performance overhead is not only the time to activate sleeping circuits, but also the time to transfer wake up signals. Furthermore, wake up signal’s transfer time is the main cause of the performance overhead. For the latency of shutting down, we assume $T_{\text{link}}$ one cycle clock. The first two parts depend on the number of routers and links by which wake up signals transfer, respectively. As mentioned in Sect. 4.1, we assume $T_{\text{activate}}$ is one clock cycle. It follows that performance overhead is not only the time to activate sleeping circuits, but also the time to transfer wake up signals. Furthermore, wake up signal’s transfer time is the main cause of the performance overhead. For the latency of shutting down, we assume one cycle clock to transfer shut down signal and one cycle clock to deactivate the vertical link.

Figure 17 shows how much performance is degraded by the proposed methods. Application performance degrades by 0.1%-2.4%. On average, the application execution time can be 1.4%, 0.57%, and 1.0% longer than the time before using our proposed method in the URSL, BRFL, and
Table 4

<table>
<thead>
<tr>
<th>Optimization</th>
<th>Area [mm²]</th>
<th>Frequency [MHz]</th>
<th>Power [mW]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Before</td>
<td>2.6181</td>
<td>200</td>
<td>1.24</td>
</tr>
<tr>
<td>After</td>
<td>2.6250</td>
<td>200</td>
<td>1.28</td>
</tr>
</tbody>
</table>

Table 5

<table>
<thead>
<tr>
<th>Optimization</th>
<th>Network</th>
<th>Execution time [clock]</th>
<th>Energy [power*clock]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Before</td>
<td>BRFL</td>
<td>26731945</td>
<td>3.00E+8</td>
</tr>
<tr>
<td></td>
<td>URSL</td>
<td>3252239</td>
<td>2.25E+8</td>
</tr>
<tr>
<td></td>
<td>BRHL</td>
<td>28620450</td>
<td>2.41E+8</td>
</tr>
<tr>
<td>After</td>
<td>BRFL</td>
<td>27054033</td>
<td>2.22E+8</td>
</tr>
<tr>
<td></td>
<td>URSL</td>
<td>33113266</td>
<td>1.71E+8</td>
</tr>
<tr>
<td></td>
<td>BRHL</td>
<td>29258882</td>
<td>1.79E+8</td>
</tr>
</tbody>
</table>

Table 6

<table>
<thead>
<tr>
<th>Optimization</th>
<th>Network</th>
<th>Execution time [clock]</th>
<th>Energy [power*clock]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Before</td>
<td>BRFL</td>
<td>3393218</td>
<td>4.31E+8</td>
</tr>
<tr>
<td></td>
<td>URSL</td>
<td>4785367</td>
<td>2.97E+8</td>
</tr>
<tr>
<td></td>
<td>BRHL</td>
<td>3601946</td>
<td>3.20E+8</td>
</tr>
<tr>
<td>After</td>
<td>BRFL</td>
<td>3412271</td>
<td>3.27E+8</td>
</tr>
<tr>
<td></td>
<td>URSL</td>
<td>4864288</td>
<td>2.19E+8</td>
</tr>
<tr>
<td></td>
<td>BRHL</td>
<td>3620275</td>
<td>2.34E+8</td>
</tr>
</tbody>
</table>

BRHL, respectively. The results demonstrate that performance overhead of the proposed methods is quite small.

5.5 Synthesis Results

We synthesized the Cube-0 chip (not including wireless transceivers) before and after using the proposed vertical link on/off optimization with Synopsys Design Compiler. Fujitsu CS202SZ 65nm process technology is used for the design. The implementation costs of optimization hardware modules (Vertical link monitor, Arbitration logic and PSM/FSM manager) mainly embody in three aspects: area, frequency, and power consumption. Table 4 shows that the implementation area increases by 2.7% after the power optimization while the power consumption of one single chip, not including vertical links, increases by 3.1%. This power overhead is quite small compare with the energy reduction after using the proposed methods. Because the power consumption (8.1mW) of inductor as mentioned in Table 1, is much larger than the other parts, not to mention the power overhead (0.02mW shown in Table 4). In comparison to power saving gained from the proposed approach, this power overhead is negligible. Moreover, the operating frequency of Cube-0 is not affected by using our proposed low power regulations for vertical links. The whole overheads are quite small.

5.6 Effect of the Threshold on Power-Performance

Power savings and latency-throughput are conflicting optimization goals. Higher power savings require shutting down more vertical links. However, fewer active vertical links increase the inter-chip traversal latency of packets, which was shown to degrade network throughput. Different applications have different power and performance demands. We should find an appropriate balance between these two goals.

In the history-based policy, a set of threshold parameters can be dynamically changed to adjust the on/off vertical link policy. Since latency/throughput performance is very important, we change the thresholds to explore the trade-off between power savings and latency/throughput performance under those network traffic conditions. In Fig. 18, we show the values of the thresholds we set during each simulation. Under different threshold settings, latency and dynamic power savings are also different. With more aggressive threshold, we can obtain more power savings but at the cost of a higher latency.

In Fig. 19, we show latency vs. power saving for LU application under URSL network. From the simulation results, we can see that, more power savings can be obtained but at the cost of a higher latency and throughput. That is, an improvement in one goal can only be obtained by degrading the other. Hence, we must select a good balance point for this benchmark program.
6. Conclusions

In this paper, we proposed vertical links on/off methods for wireless 3-D NoC with ring topology to reduce the power consumption of the wireless 3-D NoC efficiently. We explained how to shut down and wake up the target vertical links with PSM and FSM methods. We also introduced three ring network architectures, URSL, BRFL, and BRHL, for wireless 3-D NoCs, and the proposed power optimization methods were applied to these networks. We focus on ring networks, though the proposed methods can be applied to other network topologies.

We compared the application performance and power consumption among the three proposed ring network architectures. In addition, the power saving and performance overhead were analyzed in detail by using a full-system CMP simulator. The results showed that the proposed methods reduced the network power consumption by 23.4%-29.3% while the performance overhead is only 0.1%-2.4%.

Acknowledgements

This work is supported in part by a Grant-in-Aid for the Global Center of Excellence for High-Level Global Cooperation for Leading-Edge Platform on Access Spaces from the Ministry of Education, Culture, Sport, Science and Technology in Japan.

References


Hao Zhang received B.E degree from Shaanxi University of Science and Technology, China in 2004 and M.S degree from Northwestern Polytechnic University, China in 2008. He is currently a Ph.D candidate at Keio University, Japan. His research interests include Network on Chip and Parallel architectures.

Hiroki Matsutani received the BA, ME, and PhD degrees from Keio University, Japan, in 2004, 2006, 2008, respectively. He is currently an assistant professor in the Department of Information and Computer Science, Keio University. His research interests include the areas of computer architecture and interconnection networks.

Yasuhiro Take received the B.S. and M.S degree in electrical engineering from Keio University, Yokohama, Japan, in 2010, 2012, respectively, where he is currently working toward the Ph.D. degree. Since 2009, he has been engaged in research on the 3-D-stacked inductive interchip wireless interface for system in a package.

Tadahiro Kuroda received the Ph.D. degree in electrical engineering from the University of Tokyo, Japan in 1999. In 1982, he joined Toshiba Corporation. From 1988 to 1990, he was a Visiting Scholar with the University of California, Berkeley. In 1990, he was back to Toshiba, and engaged in the research and development of BiCMOS ASICs, ECL gate arrays, high-speed CMOS LSIs for telecommunications, and low-power CMOS LSIs for multimedia and mobile applications. In 2000, he moved to Keio University, Japan, where he has been a professor since 2002. His research interests include ubiquitous electronics, sensor networks, wireless and wireline communications, and ultra-low-power CMOS circuits.

Hideharu Amano received Ph.D degree from the Deparment of Electronic Engineering, Keio University, Japan in 1986. He is currently a professor in the Department of Information and Computer Science, Keio University. His research interests include parallel architectures and reconfigurable systems.