

DOI: 10.13382/j.jemi.B2104982

# NoC 中基于数据包分类的功率门控策略\*

欧阳一鸣 王世杰 王奇

(合肥工业大学计算机与信息学院 合肥 230601)

**摘要:**随着制造工艺的精进,静态功耗逐渐成为路由器的主要功耗来源之一。针对基于片上网络架构的多核系统高功耗问题,提出了一种基于数据包分类的功率门控策略,并修改了路由器架构。首先对数据包进行分类,其次利用旁路对分类数据包进行不同的处理,来绕过休眠路由器,从而降低由于功率门控的应用而导致的额外数据包延迟和功耗。旁路可以保证路由器休眠时的数据包传输,增加休眠路由器的休眠时间,减少静态功耗。数据包的分类处理使得路由器的休眠和唤醒更加合理高效。仿真结果表明,本文方案能明显减少路由器的静态功耗和网络的延迟,相比传统的功率门控技术,最大减少 72.4% 静态功耗和 16.8% 平均数据包延迟;相比经典的采用功率门控的旁路路由器,减少 12.4% 静态功耗和 4.7% 平均数据包延迟。路由器的额外硬件开销仅增加了 3.2%。

**关键词:** 片上网络;功率门控;旁路;能效

**中图分类号:** TP301      **文献标识码:** A      **国家标准学科分类代码:** 520.30

## Power gating strategy based on packet classification in NoC

Ouyang Yiming Wang Shijie Wang qi

(School of Computer Science and Information Engineering, Hefei University of Technology, Hefei 230601, China)

**Abstract:** With the improvement of the manufacturing process, static power consumption has gradually become one of the main power consumption sources of routers. Aiming at the high power consumption of multi-core systems based on the on-chip network architecture, this paper proposes a power gating strategy based on packet classification, and modifies the router architecture. In this scheme, first classify data packets, and then use bypass to perform different processing on classified data packets to bypass the dormant router, thereby reducing the extra data packet delay and power consumption caused by the application of power gating. Bypass can ensure the transmission of data packets when the router is sleeping, increase the sleep time of the sleeping router, and reduce static power consumption. The classification of data packets makes the sleep and wake-up of the router more reasonable and efficient. The simulation results show that the solution in this paper can significantly reduce the static power consumption of the router and the delay of the network. Compared with the traditional power gating technology, it can reduce the static power consumption by 72.4% and the average packet delay by 16.8%; compared with the classic power gate, the controlled bypass router reduces the static power consumption by 12.4% and the average packet delay by 4.7%. The additional hardware overhead of the router has only increased by 3.2%.

**Keywords:** network on chip (NoC); power gating; bypass; efficiency

## 0 引言

随着半导体制造工艺的发展,单个芯片上可以集成数百以至数千个 IP (intellectual property) 核,这使得将系

统集成到单个芯片上成为可能,片上系统(system on chip, SoC)因此产生。随着系统中电子原件的逐渐缩小和系统集成度的增加,现有的总线结构已经不能满足系统对延迟、吞吐量、功耗以及扩展性的要求。因此设计一种新的体系结构以解决众多 IP 核之间的高速通信成为

研究的关键所在。片上网络(network on chip, NoC)的出现为多核芯片通信提供了新的解决方案。现在 NoC 已经成为芯片多处理器系统的标准通信架构<sup>[1-2]</sup>。受片上资源的限制, NoC 的设计面临许多挑战, 功耗管理问题即是其中之一。然而晶体管尺寸的进一步缩小和芯片工作电压的降低, 使得静态功耗在片上网络功耗的占比变得尤为突出。因此减少静态功耗, 成为解决片上网络功耗管理问题的关键。

功率门控是一种经典且有效减少静态功耗的技术, 通过给空闲模块增加开关来管理模块从而减少静态功耗<sup>[3]</sup>。通常直接将功率门控应用于 NoC 并不会有很好的表现, 而是采用一些策略, 例如通过控制信息数据包提前唤醒路由器, 或者通过旁路路径跳过路由器部分流水线阶段。文献[4]通过轻量级编码的控制信息, 使得数据包到达时被功率门控路由器时, 路由器恰好处于活跃状态, 最大程度减少路由器的唤醒时间, 从而减少静态功耗; 文献[5]在水平和垂直方向预定义了旁路路径。因此, 如果数据包不需要通过休眠路由器来更改传输方向或弹出到本地, 则数据包可以绕过休眠路由器, 从而降低路由器的唤醒频率, 有效地减少静态功耗; 文献[6]通过预定义的虚拟旁路路径来绕开活跃或者休眠的中间路由器, 从而延长中间路由器的休眠时间, 减少路由器的频繁唤醒, 从而减少静态功耗。文献[7]提出了一种粗粒度的功率门控技术, 通过旁路机制, 高效的处理所有类型的微片, 包括注入、弹出、直行和转弯, 几乎消除了唤醒过程带来的延迟开销。该技术可以很好地与自适应路由算法配合使用。由于减少了数据包的绕行且不需要任何额外的路由器间布线, 具有良好的可扩展性

通过上述研究结果可以发现, 在合理的策略下使用功率门控, 不仅可以最大程度的发挥其减少静态功耗的优点, 同时也可以尽量去减少使用其带来的额外开销。

当片上网络处于低流量时, 由于 NoC 的分布式结构和流量不均匀, NoC 中的路由器不会长期的处于空闲状态, 此时采用不合理的功率门控策略不会减少静态功耗, 甚至有较大的额外开销。基于片上网络不同流量模式的通信特点, 本文首先对数据包做更细的分类, 然后根据不同的数据包类型构建功率门控旁路, 并针对不同类型的数据包做相应处理, 减少路由器的休眠和唤醒。当路由器处于活跃状态时, 旁路的使用可以进一步减少延迟和功耗。

## 1 片上网络与功率门控

### 1.1 片上网络路由器基本架构

在片上网络中, 核之间通过路由器进行数据通信。图 1 展示了一个具有 4 阶段流水线的五端口虚通道

(virtual channel, VC) 路由器。路由器外部包括 5 个输入端口和 5 个输出端口, 分别连接东南西北 4 个方向上的邻居路由器和本地处理单元(process element, PE)。路由器内部包含输入缓冲区(input buffer)、路由计算单元(router computing unit)、虚通道分配器(virtual channel allocator)、交叉开关仲裁器(switch arbitrator, SA)和 5×5 的交叉开关(crossbar)。一个数据包被分割为一个头微片, 多个体微片和一个尾微片。一个数据包到达路由器, 以微片的形式到达输入端口, 首先会存储在流水线寄存器当中。如果是数据微片, 则经过流水线寄存器后, 根据虚通道编号进入输入缓冲区中相应 VC。特别的, 当该微片是数据包的头微片时, 该微片将被送往路由计算单元, 根据其中携带的目的地址等路由信息, 按照一定的路由算法来为其选择输出端口, 一旦输出端口确定以后, 头微片还需要通过虚通道分配单元进行 VC 分配, 然后通过交叉开关仲裁单元进行仲裁, 一旦交叉开关仲裁成功后, 就进行交叉开关传输。体微片和尾微片无需进行 RC、VC 分配。数据包传输完成后, 尾微片还需要将 VC 释放。

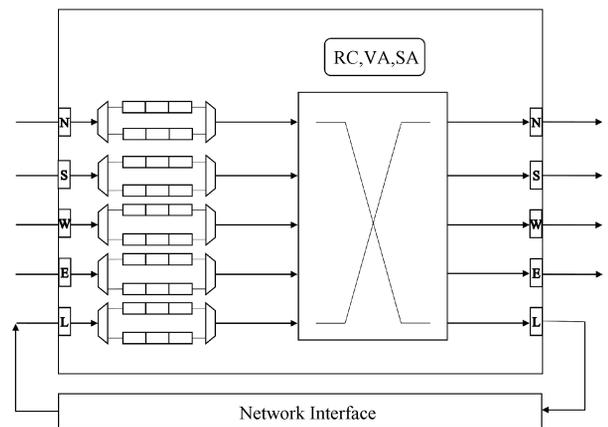


图 1 片上网络路由器基本架构

Fig. 1 The basic architecture of the NoC

网络接口(network interface, NI)提供路由器与更高级别协议之间的连接, 并负责封装和向/从网络发送/接收微片。为了减少路由器延迟, 先前的研究提出了利用 NI 来实现源路由和推测路由<sup>[8]</sup>, 其中 NI 可以执行路由计算(router computing, RC)和虚通道分配(virtual channel allocation, VA)阶段而不是相应的路由器。

### 1.2 片上网络功率门控路由器基本架构

片上网络功率门控路由器的基本架构如图 2 所示, 该路由器在上述五端口虚通道路由器增加了功率门控模块。实现方法是在路由器和 VDD 线或者路由器和 GND 线之间插入电源开关, 并通过打开或者关闭电源开关来管理空闲路由器, 从而达到节省静态功耗的目的, 并增加

相应的信号线来获取和控制相邻路由器的状态。根据被功率门控模块的大小,功率门控又分为粗粒度功率门控和细粒度功率门控。一般来说传统的功率门控属于粗粒度,且应用广泛,通过一个电源开关控制整个路由器的电源,优点是便于管理和控制,开销小,缺点是唤醒延迟相对较高,开销一般为微秒级别<sup>[9]</sup>。然而实际情况下并不一定需要关闭整个路由器,因为路由器的部分组件是频繁地使用的,而另一部分组件可以通过一些方法减少使用频率。因此产生了细粒度的功率门控<sup>[10]</sup>,一方面可以进行更细粒度的功率门控,另一方面是唤醒延迟的减少<sup>[11-12]</sup>。缺点是设计相对复杂,面积开销较大。

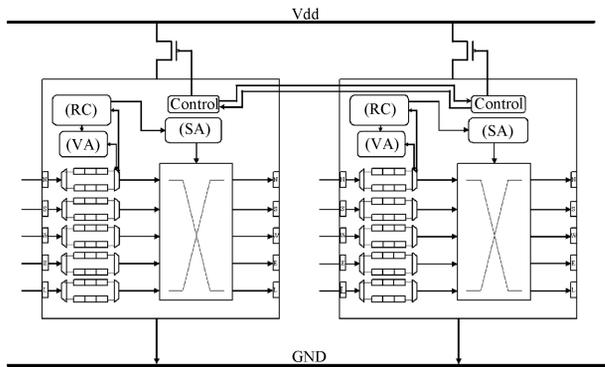


图 2 片上网络功率门控路由器基本架构

Fig. 2 Basic architecture of NoC power-gated router

### 1.3 功率门控中的时间参数

在功率门控设计中,一般通过唤醒延迟,累计延迟和盈亏平衡时间(break even time, BET)这几个时间参数来衡量设计的效率。

**唤醒延迟:**路由器由休眠状态充电至完全活跃状态所经历的时钟周期,文献[13]表明一般为6~12个时钟周期,这种唤醒延迟会阻塞数据包的传输,增加数据包传输延迟。

**累积延迟:**数据包在传输过程中可能会遇到不止一个断电路由器,唤醒所有断电路由器的所经历的时钟周期之和。

**盈亏平衡时间:**每次唤醒断电的路由器都会消耗很多能量,BET定义为被门控元件必须保持休眠的连续周期数以补偿唤醒该断电元件的能量开销。比如功率门控的开销为20 pJ,而被功率门控元件一个时钟周期可以节省2 pJ的功耗,则当被功率门控元件休眠10个时钟周期才能抵消它唤醒一次的功耗开销。

### 1.4 功率门控的状态转化

功率门控路由器通常有以下几种状态,活跃(Active),空闲(Idle),休眠(Sleep),待活跃(To\_active)。如图3所示,阐明了几种状态之间的转化。

**Active:**活跃状态,路由器正常工作,并向邻居路由器

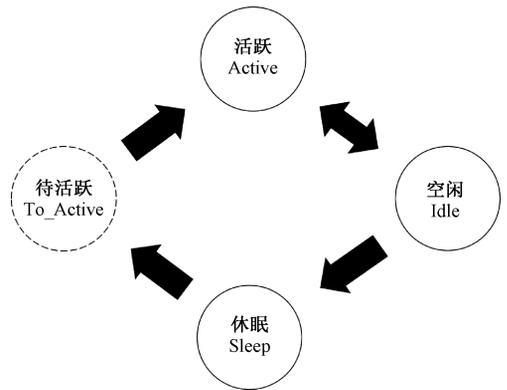


图 3 功率门控路由器的状态转化

Fig. 3 State transition of power gated router

发送活跃信号,表示自己处于活跃状态。若检测到当前路由器  $T_{detect}$  个周期未收到微片,且路由器的缓冲区为空,路由器中也没有正在处理的微片,则路由器由 Active 状态进入 Idle 状态。

**Idle:**空闲状态,此时路由器缓冲区为空,且没有正在处理的微片。若此时路由器收到微片,则路由器立马返回 Active 状态,若经过  $T_{Sleep}$ ,未收到微片,则进入 Sleep 状态。

**Sleep:**休眠状态,路由器会向相邻路由器发送休眠(sleep)信号,此时路由器处于断电状态,不能进行工作。

**To\_active:**待活跃状态, Sleep 到 Active 的中间状态,此时路由器仍不能工作,经过  $T_{Wakeup}$ ,变为 Active 状态。

## 2 功耗问题分析

### 2.1 片上网络的功耗

通常 NoC 的功耗被分为两大部分,1) 动态功耗,也称动态开关功耗,主要是由数据包路由和传输带来的功耗,例如路由器对输入端口缓存的头微片进行路由计算带来的功耗;2) 静态功耗也称静态泄漏功耗,是片上网络在开启状态下,及时没有任何数据包传输也需要的维持功耗。

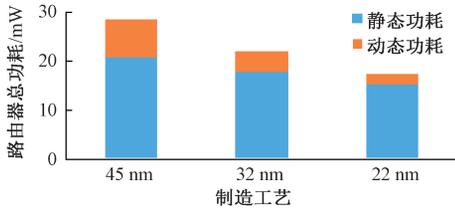
片上网络作为多核芯片的组成部分,近年来,随着制造工艺的精进,路由器的总功耗有所下降。但是路由器的静态功耗几乎没有变化,因此路由器中的静态功耗占比逐渐增加,成为路由器的主要功耗。所以亟需一种高效的技术来减少路由器静态功耗。

文献[14-15]表明,NoC 的功耗中路由器的静态功耗占很大一部分,随着晶体管尺寸的进一步缩小和芯片工作电压的降低,片上网络中静态功耗的占比变得尤为突出。文献[14-18]中发现,在 65 nm 制造工艺下,静态功

耗占 NoC 总功耗的 17.9%;在 45 nm 制造工艺下,静态功耗占 NoC 总功耗的 47.7%;在 22 nm 制造工艺下,静态功耗占 NoC 总功耗的 74%。随着制造工艺的进一步发展,静态功耗将成为 NoC 功耗中的主要部分<sup>[19-20]</sup>。

### 2.2 路由器的功耗

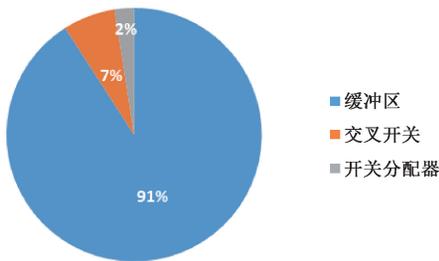
通过 DSENT<sup>[15]</sup>,一种 NoC 建模工具,用于电气和光网络的快速设计和空间探索,能够对光电片上互联进行快速地跨层次面积和功耗评估,得到如图 4(a)、(b)、(c)所示。



(a) 不同制造工艺下路由器的功耗  
(a) Power consumption of routers under different manufacturing processes



(b) 45 nm 工艺下各部件动态功耗占比  
(b) The proportion of dynamic power consumption of each component under the 45nm process



(c) 45 nm 工艺下各部件静态功耗占比  
(c) The proportion of static power consumption of each component under the 45nm process

图 4 不同芯片制造工艺下路由器及不同部件的功耗占比

Fig. 4 The power consumption ratio of routers and different components under different chip manufacturing processes

图 4(a)、(b)、(c)描述了不同芯片制造工艺下路由器中动态功耗与静态功耗以及路由器不同部件所消耗的动态功耗和静态功耗占比。

图 4(a)为不同制造工艺下路由器的功耗,在 45 nm 制造工艺下,静态功耗约为 20.5 mW,在 32 nm 制造工艺下,静态功耗约为 17.5 mW,在 32 nm 制造工艺下,静态

功耗约为 15 mW。图 4(b)为 45 nm 制造工艺下路由器各个部分的动态功耗,其中缓冲区的静态功耗占比最大为 76%,交叉开关紧随其后为 14%,时钟的功耗为 7%,交叉开关分配器及其余部分占 3%。图 4(c)为 45 nm 制造工艺下路由器各个部分的静态功耗,其中其中缓冲区的静态功耗占比最大为 91%,交叉开关的静态功耗占比为 7%,交叉开关及其余部件静态功耗占比为 2%。结合图 4(b)、(c),不难发现路由器中缓冲区消耗占比逐渐增加。

## 3 基于数据包分类的功率门控策略

### 3.1 数据包分类

首先本文根据数据包微片在当前路由器的路由方式将数据包分为 4 类,1)直行数据包,即直行穿过该路由器的数据包,比如从东端口进入从西端口输出,或者从南端口进入从北端口输出之类的等;2)转弯数据包,即在该路由器进行转弯的数据包,比如从东端口进入从南(或者北)端口输出;3)注入数据包,即从本地路由器 PE 核通过路由器注入到网络中的数据包的;4)弹出数据包,即通过当前路由器从网络中弹出到本地 PE 核的。

如图 5 所示为数据包的分类,路由算法为 XY 路由算法。路由器 01 发向路由器 03 的数据包,在路由器 02 被视为直行数据包。路由器 01 发向路由器 12 的数据包,在路由器 02 被视为转弯数据包。由路由器 02 发出到网络中其他路由器的数据包即为 PE 注入数据包。路由器 01 发向路由器 02 的数据包,在路由器 02 被视为弹出数据包。

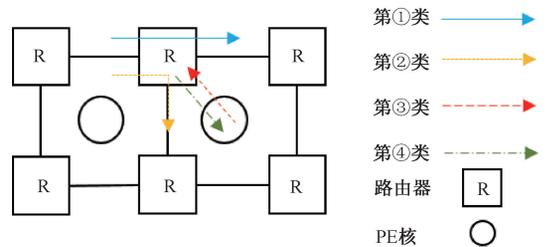


图 5 数据包的分类

Fig. 5 Classification of packets

### 3.2 流量模式与数据包的关系

本文在传统的有线 NoC 中,观察到在随机、位翻转以及洗牌流量模式当中,直行数据包的平均占比为 54.7%,弹出和注入数据包的平均占比为 30.2%,转弯数据包的占比为 15.1%。

图 6 为不同流量模式下不同类型微片的占比,可以发现其中直行、注入和弹出的数据包占了所有数据包的 85%左右。如果可以通过对直行、注入、弹出类型的数据

包进行不同的处理,从而避免或者减少路由器的唤醒,增加路由器休眠的时间,从而节省更多的静态功耗。减少由于唤醒路由器而带来的额外功耗和数据包在传输当中由于频繁唤醒路由器所带来的累计唤醒延迟。

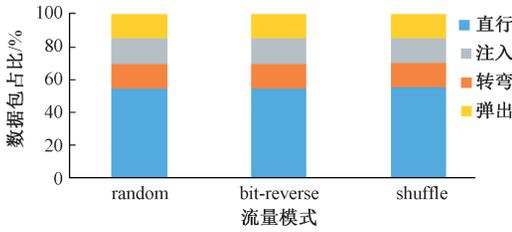


图 6 不同流量模式不同类型微片占比

Fig. 6 Proportion of different types of flit in different comprehensive traffic modes

### 4 BbTo 路由器架构设计

本文根据数据包的分类处理和旁路的使用,提出了 BbTo (based on bypass turn on) 路由器设计方法。对于不同类型数据包的处理,本文通过 BbTo 路由器的整体设计和网络接口的修改来完成。同时根据本文方案的改动,对功率门控控制单元中的休眠预测器也进行了同步的改动。

#### 4.1 BbTo 路由器整体设计

BbTo 路由器的架构如图 7 所示。本文在片上网络功率门控路由器架构的基础上增加了 BbTo 控制器 (BbTo controller)、旁路锁存器 (Bypass latch) 和相应的信号线。微片通过旁路传输时由旁路锁存器暂存。

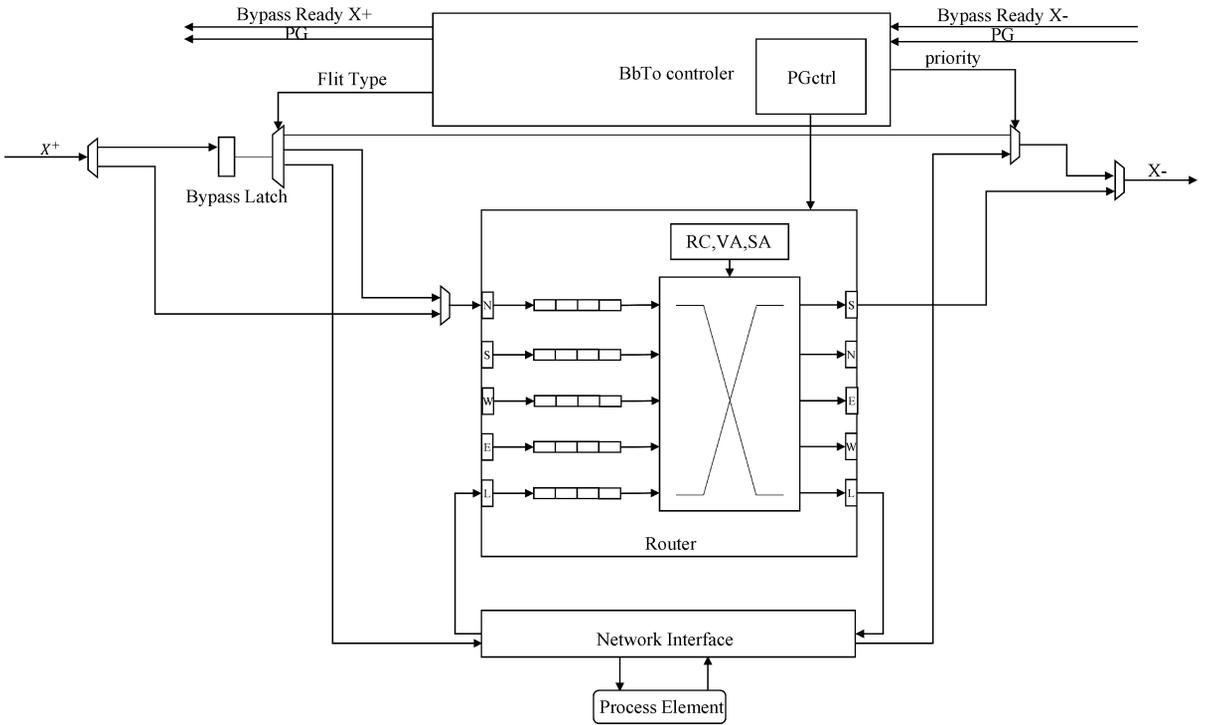


图 7 BbTo 路由器架构

Fig. 7 BbTo router architecture

BbTo 控制器包含 PGctrl 单元,以及对旁路的控制逻辑。Bypass\_Ready 信号用于获取相邻路由器旁路锁存器状态。PG 信号用于获取相邻路由器功率门控状态。

PGctrl 单元则负责本地路由器的休眠和唤醒以及功率门控状态的信息传递,其中路由器的休眠主要通过休眠预测器。

图 7 中描述的是东西方向的直行旁路及相关组件,其他方向原理和构造相似。其中 X+ 表示从西方向路由器接收的微片, X- 表示向东方向路由器的微片。且路径

中的所有旁路锁存器和多路复用器都要具有相同的微片宽度,本文方案设置为 1 微片大小。

#### 4.2 功率门控机制

当微片到达时,根据当前路由器的功率门控状态,会有下述处理。

##### 1) 路由器处于休眠状态

微片到达当前路由器,由于路由器休眠,微片将会进入到旁路锁存器中。此时 BbTo 控制器将会检查该微片类型并进行相应的处理。如果该微片是转弯微片,则

BbTo 控制器将会通过 PG 控制器唤醒路由器。如果该微片是弹出微片,则微片将通过旁路到达 NI 接口,然后弹出到 PE。

如果当前路由器需要将微片注入网络中,则微片通过 PE 进入 NI,通过封装器进行封装相应路由信息后,在控制器的控制下,通过旁路到达相应的输出端口。

对于注入微片,首先需要获取输出路由器功率门控状态,如果路由器处于活跃状态,则需判断旁路使用情况。如果输出路由器的旁路空闲,且当前旁路无正在传输微片,对于直行微片,则将其保存到输出路由器旁路锁存器当中。若为转弯微片则唤醒路由器。若为弹出微片,则通过旁路弹出到 NI 中。若旁路不可用,则通过路由器进行注入。

如果输出路由器休眠,则需判断旁路使用情况,如果输出路由器旁路空闲,且当前旁路无正在传输微片,对于直行微片,则将其保存到输出路由器旁路锁存器当中。若为转弯微片则打开路由器。若为弹出微片,则通过旁路弹出到 NI 中。若旁路不可用,则唤醒路由器,一个睡眠等待时间后,路由器处于活跃状态。

在输出端口,本文方案对微片的优先级进行了规定,以避免发生死锁。本文规定旁路中的微片优先级最高,本地注入到网络中的微片优先级紧随其后。为避免本地微片发生饥饿,本文方案设置了饥饿阈值 Local\_hungry。达到饥饿阈值时,改为本地注入到网络中的数据包优先,本地数据包注入到网络后,饥饿阈值减少,优先级恢复。

## 2) 路由器处于活跃状态

微片到达当前路由器,若 Bypass Ready 信号有效,则进入旁路锁存器(bypass latch),然后 BbTo 控制器通过微片头部信息判断微片类型,执行与路由器休眠时同样的数据包处理,不同之处是转弯微片不需要等待路由器打开。若 Bypass Ready 无效,则数据包进入路由器,通过路由器处理。特别地当旁路中数据包传输端口与路由器中传输数据包输出端口不一致时,可同时传输,提高了并行性。

## 4.3 路由器的休眠预测器

在基础的功率门控路由器中,如果路由器连续 4 个周期处于空闲状态,那么可以预测路由器在紧接着的 10 个周期也会处于空闲状态,那么此时将对路由器采用功率门控。但是在本文方案中,对路由器进行了修改之后,唤醒条件也随之发生了改变,因此预测器也需要作出相应的改变。因为当直行、注入和弹出数据包到来时,不需要唤醒路由器。

如图 8 所示,本文方案的预测器通过一个 4 位移位寄存器和一些逻辑门实现。任一时钟周期,如果有转弯微片到达路由器,则向移位寄存器输入一个逻辑 1,接下来的 4 个周期功率门控信号( $PG=0$ )被禁用,则路由器

唤醒。反之,如果没有转弯微片到达路由器,则向移位寄存器输入一个逻辑 0。如果连续 4 个周期没有转弯微片到达,那么或非门输出为逻辑 1。同时,如果路由器的所有缓冲区为空,且路由器没有正在传输的微片,即  $Empty=1$ ,则触发功率门控信号( $PG=1$ ),路由器开始休眠。

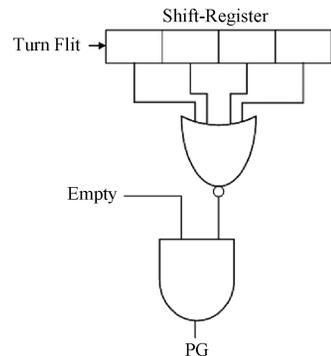


图 8 BbTo 路由器的休眠预测器

Fig. 8 The sleep predictor of the BbTo router

## 4.4 网络接口

在基础的虫孔交换的 NoC 中,NI 负责接收来自节点的数据并将其封装到数据包和微片中,分配 VC 并在相关路由器的 NI 输入端口检查流控制的信令信息,并将格式化的微片注入网络。通过 NI 接口实现了路由和计算的分离。

图 9 为本文的网络接口,增加了控制器(controller),并对封装器(wrapper)进行了相应修改。封装器主要是对注入到网络中的微片增加路由信息。控制器则根据相邻路由器的状态以及路由信息将注入数据包通过旁路或者交叉开关传输到相邻路由器。

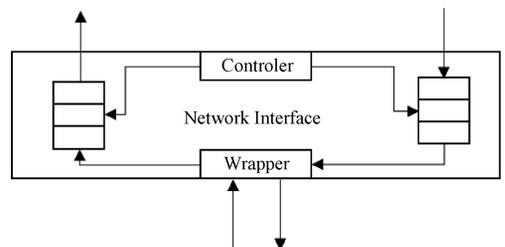


图 9 网络接口

Fig. 9 Network interface

## 5 实验评估

实验主要从静态功耗、平均数据包延迟、面积开销 3 个方面的工作性能进行评估。本文参考的实验对象有 4 种,方案 1 为传统有线 NoC;方案 2 为传统功率门控 NoC;方案 3 为采用前向路由的功率门控 NoC;方案 4 采用文献[3]提出的 TOOT 架构的功率门控 NoC;方案 5 为本文方案。

### 5.1 功耗和延迟

本文使用 Noxim, 一个周期精确的片上网络模拟器进行模拟仿真, 通过 DSENT<sup>[7]</sup> 丰富了 Noxim 45 nm 工艺建模的静态和动态功耗。首先将模拟器预热了 1 000 个周期, 然后收集了 10 000 个周期网络的统计数据。表 1 列出了相关参数的设置, 实验流量模式采用随机模式、位反转模式、洗牌模式, 对网络的平均数据包延迟, 静态功耗进行实验分析。

表 1 路由器及功率门控关键参数

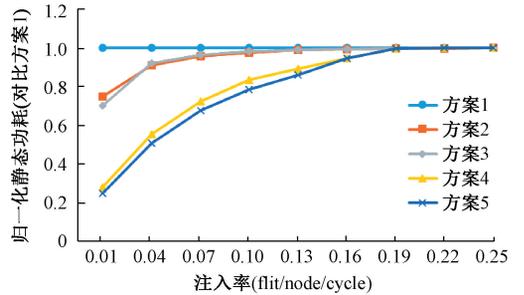
Table 1 Key parameters of router and power gating

参数	值
网络拓扑	8×8 mesh
路由器	4 阶段流水线
虚通道数量	1
输入缓冲区深度	4 微片
路由算法	XY 路由算法
微片大小	128 bit
预热时间	1 000 时钟周期
运行时间	10 000 时钟周期
时钟频率	1 GHz
路由器唤醒延迟	8 时钟周期
盈亏平衡时间	10 时钟周期
睡眠等待时间	4 时钟周期
流量模式	随机模式, 位反转模式, 洗牌模式

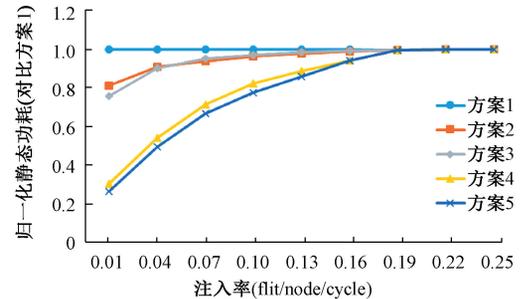
图 10(a) 为在随机流量模式下, 5 种方案在不同数据包注入率下的静态功耗比较。随着注入率的增加, 越来越多的路由器被唤醒, 功率门控节省的功耗逐渐降低。本文方案较方案 1 最大降低了 74.9% 的静态功耗, 较方案 2、方案 3 和 4 分别最大降低了 66.7%、64.2%、10.7% 的静态功耗。所有方案针对方案 1 进行了归一化处理, 即将其他方案与归一化对象进行相比, 只显示比值。方案 1 即为本文归一化对象。方案 1 的静态功耗最大, 这是因为方案 1 为传统的有线 NoC, 并未使用功率门控, 当流量较低时, 所有路由器仍然打开, 导致静态功耗增加。方案 2 采用了传统的功率门控, 在流量较低时, 可以将空闲路由器关闭, 从而在一定程度上减少了静态功耗, 然而由于策略比较简单, 静态功耗的节省并不明显。方案 3 采用了使用前向路由算法的功率门控, 由于只是改变了路由算法, 从而隐藏了一部分唤醒延迟, 所以对静态功耗没有影响, 甚至略有增加, 但是减少了延迟。方案 4 为采用文献[3]功率门控策略及架构的路由器, 通过增加旁路来减少休眠路由器的打开, 从而大幅度的减少静态功耗。

图 10(b) 为在位翻转流量模式下, 5 种方案在不同数据包注入率下的静态功耗比较。本文方案较方案 1 最大降低了 72.4% 的静态功耗, 较方案 2、方案 3 和 4 分别最大降低了 66.6%、64.5%、10.0% 的静态功耗。所

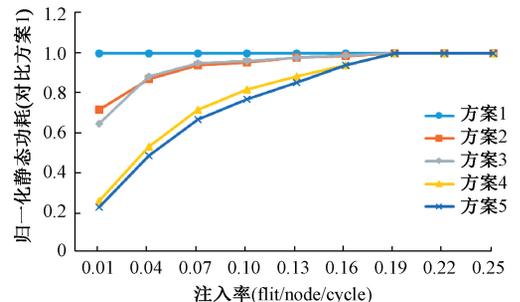
有方案针对方案 1 进行了归一化处理。



(a) 随机流量模式下的静态功耗  
(a) Static power in random traffic



(b) 位翻转模式下的静态功耗  
(b) Static power in bit-reverse traffic



(c) 洗牌模式下的静态功耗  
(c) Static power in shuffle traffic

图 10 不同流量模式下的静态功耗

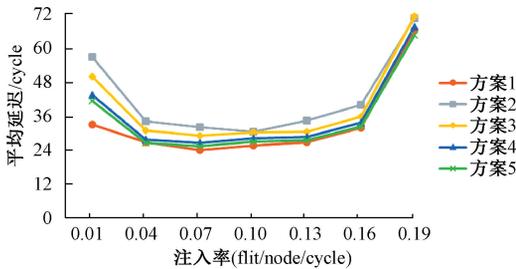
Fig. 10 Static power in different traffic

图 10(c) 为在洗牌流量模式下, 5 种方案在不同数据包注入率下的静态功耗比较。本文方案较方案 1 最大降低了 77.1% 的静态功耗, 较方案 2、3 和 4 分别最大降低了 67.6%、64.6%、11.5% 的静态功耗。所有方案针对方案 1 进行了归一化处理。本文在位翻转和洗牌的流量模式下, 静态功耗的节省均有提升。

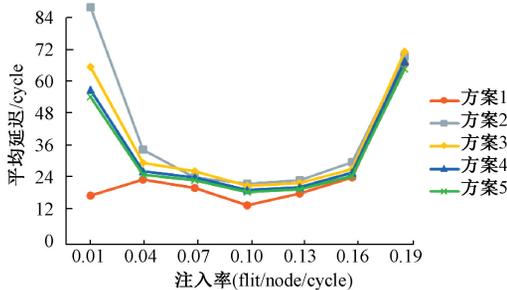
图 11(a) 为在随机流量模式下, 5 种方案在不同数据包注入率下的平均数据包延迟比较。当注入率为 0.16 flit/node/cycle 时, 本文方案较方案 1 降低了 73.3% 的平均延迟, 较方案 2、3 和 4 分别降低了 55.9%、17%、16.5% 的平均延迟。

图 11(b) 为在位翻转流量模式下, 5 种方案在不同数据包注入率下的平均数据包延迟比较。当注入率为 0.16 flit/node/cycle 时, 本文方案较方案 1 降低了

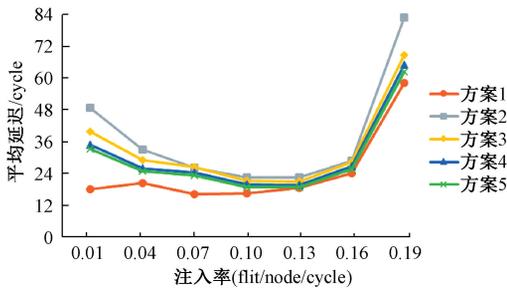
35.7%的平均延迟,较方案2、3和4分别降低了32.7%、17%、16.9%的平均延迟。



(a) 随机流量模式下的平均延迟  
(a) Average latency in random traffic



(b) 位翻转流量模式下的平均延迟  
(b) Average latency in bit-reverse traffic



(c) 洗牌流量模式下的平均延迟  
(c) Average latency in shuffle traffic

图 11 不同流量模式下的平均延迟

Fig. 11 Average latency in different traffic

图 11(c) 为在洗牌流量模式下,5 种方案在不同不同数据包注入率下的平均数据包延迟比较。当注入率为 0.16 flit/node/cycle 时,本文方案较方案 1 降低了 26.7% 的平均延迟,较方案 2、3 和 4 分别降低了 21.5%、17%、16.8% 的平均延迟。方案 1 为传统的有线 NoC,由于并未采用功率门控,所以延迟相对与其余采用功率门控的方案较低。方案 2 为传统的功率门控,由于功率门控策略的粗糙,频繁的唤醒和关闭使得延迟不减反增。方案 3 为采用前向路由算法的功率门控,由于前向路由算法的提前计算,减少了路由延迟,从而减少了平均延迟。方案 4 为采用文献[3]功率门控策略及架构的路由器,通过旁路的使用减少了路由延迟,从而减少了平均延迟。本文方案由于进行了功率门控策略的优化,可以隐藏部分的唤醒延迟,且在路由器开启时,采用旁路,抵消了由于

功率门控的使用带来的额外延迟。

## 5.2 面积开销

本文选择的实验工具是 Synopsys 公司的 Design Compiler,使用 Verilog HDL 语言进行硬件逻辑综合。通过在 45 nm TSMC 库下使用 Design Compiler 合成修改后的路由器,基准路由器的面积为  $64\ 334\ \mu\text{m}^2$ 。BbTo 路由器的面积为  $66\ 386\ \mu\text{m}^2$ 。BbTo 路由器组件产生了约 3.2% 的额外硬件开销。

## 6 结论

随着芯片制造工艺的进步,静态功耗逐渐成为 NoC 功耗的主要部分。本文提出了一种基于数据包分类的旁路功率门控技术,可以显著减少 NoC 路由器的静态功耗。通过为注入包,弹出包和直行包提供旁路路径,避免了在注入包/弹出包和直行包遇到休眠路由器时唤醒路由器。因此,本文方案通过减少唤醒次数来降低静态功耗,通过旁路休眠路由器来降低延迟。实验结果表明,与提前唤醒的传统功率门控相比,BbTo 最大能够降低 72.4% 的静态功耗,降低 16.8% 的延迟。相比经典的采用功率门控的旁路路由器,减少 12.4% 静态功耗,减少 4.7% 平均数据包延迟。路由器的额外硬件开销仅增加了 3.2%。

## 参考文献

- [1] KUMAR S, JANTSCH A, SOININEN J P, et al. A network on chip architecture and design methodology [C]. Proceedings IEEE Computer Society Annual Symposium on VLSI. New Paradigms for VLSI Systems Design. ISVLSI 2002. IEEE, 2002; 117-124.
- [2] BENINI L, DE MICHELI G. Networks on chips: A new SoC paradigm[J]. Computer, 2002, 35(1): 70-78.
- [3] HU Z, BUYUKTOSUNOGLU A, SRINIVASAN V, et al. Microarchitectural techniques for power gating of execution units[C]. Proceedings of the 2004 International Symposium on Low Power Electronics and Design, 2004; 32-37.
- [4] YADAV S, KAPOOR H K. Lightweight message encoding of power-gating controller for on-time wakeup of gated router in network-on-chip [C]. 2019 9th International Symposium on Embedded Computing and System Design (ISED). IEEE, 2019; 1-6.
- [5] FARROKHBAKHT H, TARAM M, KHALEGHI B, et al. Toot: an efficient and scalable power-gating method for noc routers[C]. 2016 Tenth IEEE/ACM International Symposium on Networks-on-Chip (NOCS). IEEE, 2016; 1-8.

- [ 6 ] WANG P, NIKNAM S, MA S, et al. EVC-based power gating approach to achieve low-power and high performance NoC [ C ]. 2019 22nd Euromicro Conference on Digital System Design ( DSD ). IEEE, 2019: 116-123.
- [ 7 ] FARROKHBAKHT H, KAMALI H M, JERGER N E. Muffin: Minimally-buffered zero-delay power-gating technique in on-chip routers [ C ]. 2019 IEEE/ACM International Symposium on Low Power Electronics and Design ( ISLPED ). IEEE, 2019: 1-6.
- [ 8 ] DALLY W J, TOWLES B P. Principles and Practices of Interconnection Network [ M ]. Morgan Kaufmann Publishers Inc, 2004.
- [ 9 ] MATSUTANI H, KOIBUCHI M, IKEBUCHI D, et al. Ultra fine-grained run-time power gating of on-chip routers for cmps [ C ]. 2010 Fourth ACM/IEEE International Symposium on Networks-on-Chip. IEEE, 2010: 61-68.
- [ 10 ] WU Y, LIU L, WANG L, et al. Aggressive fine-grained power gating of NoC buffers [ J ]. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 2020, 39(11):3177-3189.
- [ 11 ] IKEBUCHI D, SEKI N, KOJIMA Y, et al. Geyser-1: A MIPS R3000 CPU core with fine grain runtime power gating [ C ]. 2009 IEEE Asian Solid-State Circuits Conference. IEEE, 2009: 281-284.
- [ 12 ] USAMI K, OHKUBO N. A design approach for fine-grained run-time power gating using locally extracted sleep signals [ C ]. 2006 International Conference on Computer Design. IEEE, 2006: 155-161.
- [ 13 ] CHEN L, ZHU D, PEDRAM M, et al. Power punch: Towards non-blocking power-gating of noc routers [ C ]. 2015 IEEE 21st International Symposium on High Performance Computer Architecture ( HPCA ). IEEE, 2015: 378-389.
- [ 14 ] CHEN L, PINKSTON T M. Nord: Node-router decoupling for effective power-gating of on-chip routers [ C ]. 2012 45th Annual IEEE/ACM International Symposium on Microarchitecture. IEEE, 2012: 270-281.
- [ 15 ] SUN C, CHEN C H O, KURIAN G, et al. DSENT-a tool connecting emerging photonics with electronics for opto-electronic networks-on-chip modeling [ C ]. 2012 IEEE/ACM Sixth International Symposium on Networks-on-Chip. IEEE, 2012: 201-210.
- [ 16 ] CHEN X, PEH L S. Leakage power modeling and optimization in interconnection networks [ C ]. Proceedings of the 2003 International Symposium on Low Power Electronics and Design, 2003: 90-95.
- [ 17 ] BANERJEE A, MULLINS R, MOORE S. A power and energy exploration of network-on-chip architectures [ C ]. First International Symposium on Networks-on-Chip ( NOCS'07 ). IEEE, 2007: 163-172.
- [ 18 ] PARIKH R, DAS R, BERTACCO V. Power-aware NoCs through routing and topology reconfiguration [ C ]. Proceedings of the 51st Annual Design Automation Conference, 2014: 1-6.
- [ 19 ] ZHU D, LI Y, CHEN L. On trade-off between static and dynamic power consumption in NoC power gating [ C ]. 2019 IEEE/ACM International Symposium on Low Power Electronics and Design ( ISLPED ). IEEE, 2019: 1-6.
- [ 20 ] 欧阳一鸣, 王薇, 王奇, 等. WiNoC 中拥塞避免的低功耗路由器设计 [ J ]. 电子测量与仪器学报, 2019, 33(11):110-119.
- OUYANG Y M, WANG W, WANG Q, et al. Low power router design for congestion avoidance in WiNoC [ J ]. Journal of Electronic Measurement and Instrumentation, 2019, 33(11):110-119.

### 作者简介



欧阳一鸣, 博士, 现为合肥工业大学教授, 主要研究方向为片上网络与片上系统。

E-mail: oyymbox@163.com

**Ouyang Yiming**, Ph. D. Now he is a professor in Hefei University of Technology. His main research interests include network-on-chip and system-on-chip.



王世杰, 2019 年于合肥工业大学获得学士学位, 现为合肥工业大学研究生, 主要研究方向为片上网络与功率门控。

E-mail: 554341478@qq.com

**Wang Shijie** received his B. Sc. degree from Hefei University of Technology in 2019.

Now he is a M. Sc. candidate in Hefei University of Technology. His main research interests include network-on-chip and power gating.



王奇, 2016 年于吉林大学获得学士学位, 2018 年于美国斯蒂文斯理工学院获得硕士学位, 现为合肥工业大学博士生, 主要研究方向为片上网络、容错系统和基于 CMP 的神经网络加速器设计。

E-mail: keywenchester@outlook.com

**Wang Qi** received his B. Sc. degree from Jilin University in 2016, M. Sc. degree from Stevens Institute of Technology in 2018. Now he is a doctoral student at Hefei University of Technology. His main research interests include network on chip, machine learning, fault tolerance and NoC based neural network accelerators.