文章编号:1001-0645(2007)06-0511-06

多核处理器核间互连的新型互连网络

乔保军[1,2], 石峰[1], 计卫星[1]
(1. 北京理工大学 计算机科学技术学院, 北京 100081;2. 河南大学 数据与知识工程研究所, 河南, 开封 475001)

摘 要:提出了一种用于片上核间互连的新型互连网络——基三分层互连网络. 该网络具有明显的层次性和对称性以及良好的扩展性. 与 2-D Mesh 相比,在网络规模不大时,基三分层互连网络更适用于构建片上核间的通信网络. 仿真结果表明,该网络具有较低的平均通信延迟和较高的平均吞吐率.

关键词:多核处理器;片上互连网络;网络拓扑
中图分类号:TP 393.03 文献标识码:A

A New On-Chip Interconnection Network for Multi-Core Processor

QIAO Bao-jun[1,2], SHI Feng[1], JI Wei-xing[1]
(1. School of Computer Science and Technology, Beijing Institute of Technology, Beijing 100081, China;
2. Institute of Data and Knowledge Engineering, Henan University, Kaifeng, Henan 475001, China)

Abstract: A new interconnection network for multi-core processor, named the triplet-based hierarchical interconnection network (THIN), is presented. The topology of the network is very simple and it has obvious hierarchical, symmetric and scalable characteristics. The compare results show THIN is superior to 2-D mesh to construct interconnection network when there are not too many nodes. Results of simulation showed that THIN is a promising choice for multi-core processor with low network latency and high throughput.

Key words: multi-core processor; on-chip interconnection network; network topology

片上多核之间的连接方式对多核处理器的性能有着极为重要的影响[1]. 传统的多核处理器一般采用时分多路(time-division multiplexed, TDM)总线作为多核间的通信方式,如 IBM Core Connect[2] 和 ARM AMBA[3]. 总线方式最大缺点在于扩展性不好,由于内核时分共用该总线,总线访问冲突随着内核数的增加而增加,导致系统性能下降[4].

研究表明,片上互连网络(on-chip interconnection networks)是比总线方式更适于作为核间互连的一种通信方式[5-6]. 从图论的角度出发,许多互连网络都具有良好的数学特性,如全互连网络、环网(torus)和超立方体(hypercube)等,但是这些互连网结构比较复杂,难以在片上实现. 文献[7]117中深入研究了各种网络拓扑,认为 2-D Mesh 是一种更适合作为片上网络的拓扑结构. 目前,有很多实验性的片上网络都采用 2-D Mesh,如Kumar[7]117的设计片上网络和 aSOC[8].

作者从降低节点度、减少网络链路数和缩短网络直径的角度出发,提出一种用于核间互连的新型片上互连网络——基三分层互连网络(triplet-based hierarchical interconnection network,THIN). 将 THIN 和 2-D Mesh 的静态度量和无阻塞延迟进行比较,并针对 THIN 提出了一个充分体现网络层次特性的分层地址编码方案,设计了一种分布式确定路由算法.

收稿日期:2006-10-26
作者简介:乔保军(1975—),男,博士生,讲师,E-mail:qbj@henu.edu.cn;石峰(1961—),男,教授,博士生导师.

1 THIN 网络拓扑

THIN 是一种层次化的、可扩展的互连拓扑结构. 该结构的第 0 层是单个节点,如图 1a 所示. 通过 3 条通信链路将 3 个节点彼此互连形成一个三角形,从而构成该结构的第 1 层,如图 1b 所示. 1 层网络是构造 THIN 的基本构件.

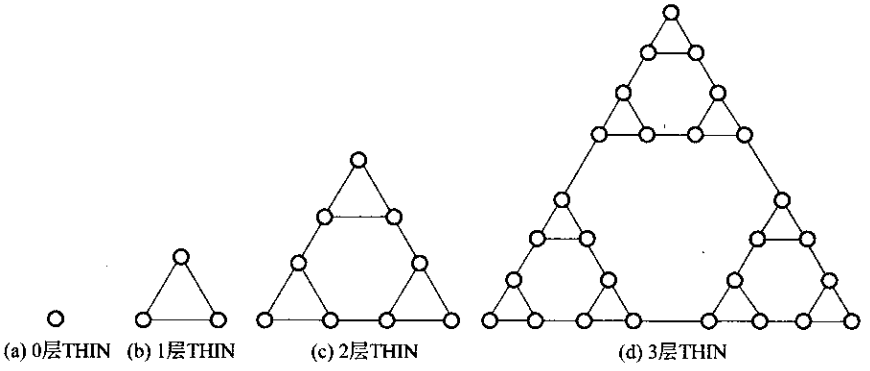

图 1 THIN 互连网络拓扑结构

Fig. 1 Topology of THIN

THIN 的递归构造过程为:在基本构件的基础上,将每个节点用一个低层网络替代,从而得到更高层的一个三角形网络结构,重复这一过程,可以构造出满足应用需求的任意层次的 THIN. 该网络拓扑结构并不局限于平衡构造,还可以进行非平衡构造.

2 网络拓扑的静态度量

网络拓扑的静态度量主要包括:网络的度、链路数和网络直径. 作者深入研究了 THIN 的各个静态度量,并与 2-D Mesh 进行了相应的比较.

定义 1 与节点 i 相邻的所有节点的个数称为节点的度(node degree),记为 d_i. 图中最大节点度称为图的度,为

$$d = \max(d_i). \tag{1}$$

THIN 中网络度表示为 d_T,

$$d_T = 3. \tag{2}$$

定义 2 k 层 THIN 的链路总数计为 L_k.

根据 THIN 的迭代构建过程,该网络中的链路总数可以使用递推公式表示为

$$\begin{cases} L_1 = 3, \\ L_k = 3L_{k-1} + 3. \end{cases} \tag{3}$$

由式(3)可知

$$L_k = 3(3^k - 1)/2 = 3(N-1)/2, \tag{4}$$

式中 N 表示 k 层 THIN 中节点的个数.

定义 3 连接 2 个节点 i 和 j 的最短路径所包含的边数称为节点 i 和 j 的距离,表示为 $P_{i,j}$. 网络中任意两节点间距离的最大值称为网络直径 P,

$$P = \max(P_{i,j}). \tag{5}$$

k 层 THIN 的网络直径记为 $P_{T(k)}$,

$$\begin{cases} P_{T(1)} = 1, \\ P_{T(k)} = 2P_{T(k-1)} + 1. \end{cases} \tag{6}$$

由式(6)可知

$$P_{T(k)} = 2^k - 1 = 2^{\log_3 N} - 1. \tag{7}$$

表 1 对比了 THIN 和 2-D Mesh 的静态度量,从表 1 中可知,THIN 和 2-D Mesh 的度都是常数,并且 THIN 的度更小. 不变的节点度使得网络接口的开销不会随着网络规模的变化而发生改变,适合 VLSI 实现,网络更容易扩展. 而且由于较小的节点度,在通道受到硬件布线限制时可以有较大的线宽,对网络的带宽有利.

表 1 THIN 和 2-D Mesh 的网络拓扑静态度量比较

Tab. 1 Comparison of topology characters between THIN and 2-D Mesh

网络类型	度	链路总数	网络直径
THIN	3	$3\times(N-1)/2$	$2^{\log_3 N}-1$
2-D Mesh	4	$2(N-\sqrt{N})$	$2(\sqrt{N}-1)$

数据链路数表示了构造网络的成本和网络的复杂度,当节点数增加时,为了使连接代价达到最小,总链路数应当按照线性规律增加. 在相同节点下,THIN 的链路总数也比 2-D Mesh 的少,这一点对于构建核间互连网络非常重要,因为较少的链路数占用的片上资源也相对较少.

网络直径是互连网络的一个重要参数,它直接影响到节点间的通信延迟,通常在包交换网络中要

求网络具有尽可能小的直径. 图 2 比较了 THIN 和 2-D Mesh 在不同节点数下的网络直径 P 的取值情况. 从图 2 中可知，当网络规模不大时，THIN 的网络直径比 2-D Mesh 的要小.

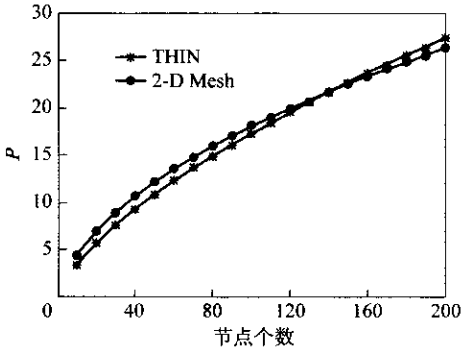

图 2　THIN 与 2-D Mesh 的网络直径的比较

Fig. 2　Comparison of network diameters

3　THIN 网络的无阻塞延迟

为了对网络性能进行全面的评价，还必须考虑网络通信延迟这一重要的动态度量，本节主要讨论 THIN 网络的无阻塞延迟.

假设消息包含 L 位数据，物理微片和微片大小相等，都等于物理数据通道的宽度 W 位，消息头假设为一个微片，这样消息的大小为 $L+W$ 位. 文献[9]中给出的互连网络在虚切入交换机制(virtual cut-through)下的无阻塞通信延迟 t 的计算公式为

$$t = D_a(t_r + t_s + t_w) + \max(t_s, t_w)\left\lceil \frac{L}{W} \right\rceil. \quad (8)$$

式中：t_r 表示作出路由决策所花费的时间；t_s 为路由器内部延迟或交换延迟，即 W 位的微片从路由器的输入传送到输出的时间；t_w 是通道延迟；D_a 是互连网络的平均距离.

根据 THIN 互连网络的拓扑结构，k 层 THIN 互连网络间的平均距离 D_T 的计算公式为

$$D_T = \frac{1}{3^{k-1}} + \frac{16 \times (6^{k-1} - 1)}{5 \times 3^k} - \frac{1}{3}. \quad (9)$$

文献[10]中给出的求解 2D-Mesh 的平均距离 $D_{2\text{-D Mesh}}$ 的计算公式为

$$D_{2\text{-D Mesh}} = \frac{2(N-1)}{3\sqrt{N}}. \quad (10)$$

在比较具有不同度的互连网络的平均距离时，由于网络度越高越有利于降低平均距离，因而只有平均距离的定义难以对通信延迟做出科学分析，因此引入归一化平均距离的概念[11].

定义 4　互连网络的归一化平均距离 μ 定义为 D_a 和 d 的乘积

$$\mu = dD_a. \quad (11)$$

由式(9)可得 k 层 THIN 互连网络的归一化平均距离 μ_T 为

$$\mu_T = 3\left[\frac{1}{3^{k-1}} + \frac{16(6^{k-1}-1)}{5\times 3^k} - \frac{1}{3}\right]. \quad (12)$$

由式(10)可知二维网络的归一化平均距离 $\mu_{2\text{-D Mesh}}$ 为

$$\mu_{2\text{-D Mesh}} = \frac{8(N-1)}{3\sqrt{N}}. \quad (13)$$

在比较互连网络的无阻塞延迟时，用网络的归一化平均距离 μ 替代 D_a. 由式(8)(12)可以得出 k 层 THIN 互连网络的无阻塞延迟计算公式为

$$t_T = 3\left[\frac{1}{3^{k-1}} + \frac{16(6^{k-1}-1)}{5\times 3^k} - \frac{1}{3}\right](t_r + t_s + t_w) + \max(t_s, t_w)\left\lceil \frac{L}{W} \right\rceil =$$

$$3\left[\frac{3}{N} + \frac{16\times(6^{\log_3 N - 1} - 1)}{5N} - \frac{1}{3}\right](t_r + t_s + t_w) + \max(t_s, t_w)\left\lceil \frac{L}{W} \right\rceil, \quad (14)$$

式中 N 是 k 层 THIN 互连网络中节点的个数.

由式(8)(13)可以得出含有 N 个节点的二维网格的无阻塞延迟计算公式为

$$t_{2\text{-D Mesh}} = \frac{8(N-1)}{3\sqrt{N}}(t_r + t_s + t_w) + \max(t_s, t_w)\left\lceil \frac{L}{W} \right\rceil. \quad (15)$$

图 3 比较了在采用相同的路由决策机制、相同的消息交换机制和相同的通信带宽的条件下，THIN 和 2-D Mesh 的无阻塞延迟情况. 在这里假设路由 t_r，t_s 和 t_w 都是一个固定常量，这里用不同节点个数

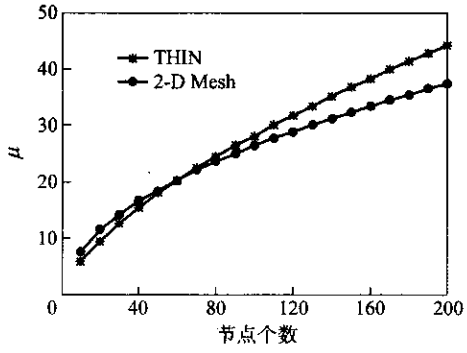

图 3　THIN 和 2-D Mesh 的无阻塞延迟比较

Fig. 3　Comparison of latency in noblocking mode

下的网络归一化平均距离来度量网络的无阻塞延迟. 从图 3 可知,当网络规模(节点个数)不很大时,THIN 的无阻塞延迟要比 2-D Mesh 的小.

通过比较 THIN 和 2-D Mesh 的静态度量和无阻塞延迟可知,在网络规模不很大的情况下,THIN 比 2-D Mesh 更宜于用来构建多核处理器上的核间互连网络.

4 节点编码方案及通信端口连接关系

THIN 采用了一种能够充分体现网络层次特性的分层地址编码方案,作者仅就平衡递归构造的 THIN 进行描述. 为了便于说明,先给出如下定义.

定义 5 用符号 $N(k)$ 表示 k 层 THIN,$n(i)$ 代表 THIN 中编码为 i 的节点. THIN 的度为 3,每个节点有 3 个通信端口,分别标记为 0_i,1_i 和 2_i,其中下标 i 是节点对应的编码.

定义 6 对于 k 层 THIN,网络的节点集记为 V_k,$|V_k|=3^k$.

定义 7 运算符$\oplus$为二进制位串的拼接运算.

定义 8 $a\longleftrightarrow b$ 表示在端口 a 和 b 之间存在通信链路,使之互连.

定义 9 记号 $0_{t,(s)_{k-1}}$ 表示了 $N(k-1)$ 中节点 $n(s)$ 的 0 号通信端口,且在构建 $N(k)$ 过程中,$N(k-1)$ 将替代基本构件 $N(1)$ 的节点 $n(t)$.

包含 3^k 个节点的 k 层 THIN,根据其递归构造过程,其节点的编码方案描述如下:

① $k=0$ 时,THIN 仅含一个节点,节点编码并不重要,可任意设定.

② $k=1$ 时,网络 $N(k)$ 中有 3 个节点,每个节点的编号 b_1b_0 用二进制形式表示,分别为 01,10 和 11,$V_1=\{01,10,11\}$. 按照式(16)给出的链路连接关系,将这 3 个节点的通信端口进行连接. 对于每个节点,编码为 0 的通信端口 $0_{b_1b_0}$ 用于外连构建更高层 THIN.

$$O_{\mathrm{P}}=(\text{目的节点 } b_1b_0-\text{源节点 } b_1b_0)\bmod 3. \tag{16}$$

式中 O_{P} 为源节点到目的节点的通信端口号.

图 4a 给出了 1 层 THIN 的节点和链路编码的示例.

③ 假设 $N(k-1)$ 的节点编码和通信端口连接已经完成,$N(k-1)$ 的节点集记为 V_{k-1}

$$V_{k-1}=\bigcup_{j=1}^{3^{k-1}}t_j, \tag{17}$$

式中 t_j 是 $N(k-1)$ 中节点编码,标记为 $b_{2k-3}b_{2k-4}\cdots b_{2i-1}b_{2i-2}\cdots b_1b_0(1\leqslant i\leqslant k-1)$.

$N(k-1)$ 的 3 个外连通信端口分别为 $0_{01\cdots01}$,$0_{10\cdots10}$和$0_{11\cdots11}$,将3 个 $k-1$ 层的 THIN 网络按照第 2 小节介绍的构造方法,遵从式(18)给出的链路连接规则可以构造出 $N(k)$.

$$\begin{cases}0_{11,(01\cdots01)_{k-1}}\longleftrightarrow 0_{01,(11\cdots11)_{k-1}},\\ 0_{11,(10\cdots10)_{k-1}}\longleftrightarrow 0_{10,(11\cdots11)_{k-1}},\\ 0_{01,(10\cdots10)_{k-1}}\longleftrightarrow 0_{10,(01\cdots01)_{k-1}}.\end{cases} \tag{18}$$

则 k 层 THIN 的节点集 V_k,

$$V_k=\bigcup_{j=1}^{3}\bigcup_{m=1}^{3^{k-1}}s_j\oplus t_m, \tag{19}$$

式中 $s_j\in V_1$,$t_m\in V_{k-1}$,$s_j\oplus t_m$ 则是 $N(k)$ 中节点的编码.

根据以上的节点编码方案和通信端口连接关系,可以得到一个完整的 THIN,其中每个节点的编码为 $b_{2k-1}b_{2k-2}\cdots b_{2i-1}b_{2i-2}\cdots b_1b_0$. 这种编码方案结

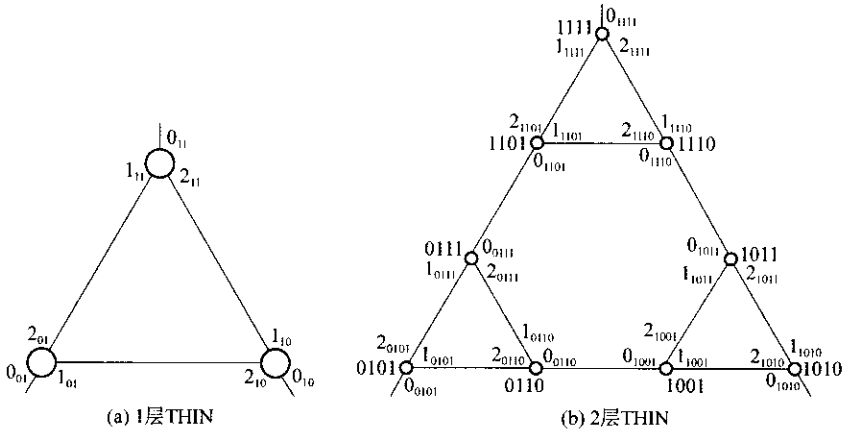

图 4 THIN 的节点编码示意图

Fig. 4 Address encoding scheme of THIN

构清晰，编码本身涵盖了网络的层次特征，能够极大地简化路由过程中的运算. 图 4b 给出了 $N(2)$ 的节点编码示例.

5　DDRA 路由算法设计

THIN 与 2-D Mesh 不同，其拓扑结构无法分解为正交的维，因此用于 2-D Mesh 上的著名的维序路由算法无法在 THIN 上运行. 作者针对 THIN 网络，提出一种能够充分利用该网络层次性的分布式确定路由算法 DDRA（distributed deterministic routing algorithm）. DDRA 在确定路由时，不需要整个网络的状态信息，从而避免了为每个节点提供全局路由信息所额外生成的通信开销和节点存储开销.

根据第 4 节有关节点编码和通信端口连接关系的描述，不难得到关于消息的目的节点、当前节点以及消息在当前节点的输出通信端口之间的对应关系. 假设一个 k 层 THIN，节点 $n(b_{2k-1}b_{2k-2}\cdots b_1b_0)$ 收到一个目的节点为 $n(b'_{2k-1}b'_{2k-2}\cdots b'_1b'_0)$ 的消息，则有

$$(b_{2k-1}b_{2k-2}\cdots b_1b_0 \neq b'_{2k-1}b'_{2k-2}\cdots b'_1b'_0)\rightarrow \exists i[(b_{2k-1}b_{2k-2}\cdots b_{2i+1}b_{2i} = b'_{2k-1}b'_{2k-2}\cdots b'_{2i+1}b'_{2i}) \wedge (b_{2i-1}b_{2i-2}\neq b'_{2i-1}b'_{2i-2})]. \tag{20}$$

该消息将经由当前节点的输出通信端口 O 续传给下一个邻节点，其中

$$O=(b'_{2i-1}b'_{2i-2}-b_1b_0)\bmod 3. \tag{21}$$

DDRA 路由算法的基本思想如下：根据消息的目的节点编码和当前节点编码，从高位到低位（2 位为 1 组）进行逐层对比，如果 2 个编码相等，表明消息已经到达目的节点，当前节点接收此消息，否则，就按照公式确定输出通信端口，将此消息续传给下一个相邻节点. k 层 THIN 上的 DDRA 路由算法的具体描述如下.

输入：当前节点编码 $b_{2k-1}b_{2k-2}\cdots b_1b_0$

　　消息的目的节点编码 $b'_{2k-1}b'_{2k-2}\cdots b'_1b'_0$

输出：节点的通信端口号，其中 3 表示消息已经到达目的地

DDRA $(b_{2k-1}b_{2k-2}\cdots b_1b_0,\ b'_{2k-1}b'_{2k-2}\cdots b'_1b'_0)$

Begin

/ 从高层到低层，逐层对比当前节点和目的节点编码 /

　FOR $(i=k;i>0;i--)$

　　IF $(b_{2i-1}b_{2i-2}\neq b'_{2i-1}b'_{2i-2})$ THEN break

　　IF $(i\leqslant 0)$ THEN

　　return 3 /该消息已经达到目的节点 /

　　ELSE

　　return $((b'_{2i-1}b'_{2i-2}-b_1b_0) \bmod 3)$

　　END IF

　END；

式（20）（21）的结论保证了算法 DDRA 的正确性，算法的复杂度是 $O(k)$.

DDRA 路由算法充分利用 THIN 的层次特性，算法设计简单，易于硬件实现，并且可以保证算法的高效性，从而实现传输速率高、低延迟的互连网络. 由于 DDRA 路由算法在每个节点上不需要存储路由表，从而降低了节点的存储开销，同时还避免了由于路由表而引起的维护开销.

6　仿真与分析

为了进一步研究 THIN 的网络性能，作者采用美国 Xilinx 公司的 XC2V8000 FPGA 进行 THIN 的硬件实现. 利用 Xilinx ISE 和仿真软件 ModelSim 构建一个含 9 个节点的 2 层 THIN，用 VHDL 硬件描述语言在行为级和 RTL 级进行描述，综合、仿真后下载到 FPGA 芯片内.

为了降低实现的复杂度，该仿真平台主要由处理器核和路由器核组成. 处理器核所实现的功能比较单一，主要包括：随机生成去往其他处理器核的消息，消息的目的地址均匀分布；接收到达自身的消息. 路由器 R 运行 DDRA 路由算法为到来的消息进行寻径，并采用 virtual cut-through 的交换方式将消息从输入端口切换到输出端口，路由器的输入/输出缓冲区由宽度为 16　bit，长度为 16 的 FIFO 构成. 在该仿真环境中，路由器之间、路由器和处理器核之间都是全双工通信方式，并使用两个单工链路取代每条全双工链路，链路的数据宽度是 16　bit，系统的工作时钟为 50　MHz.

测试结果表明：大小为 256　bit 的消息包（分成 16 个 flits，每个 flit 的宽度为 16　bit），在无阻塞的情况下，消息包的头节片从进入输入端口开始，经过寻径、仲裁、直至输出到对应的输出端口耗时 160　ns，而整个消息包通过路由器的时间为 240　ns.

为了进一步验证 THIN 网络的性能，作者在该仿真平台上分别测试了在不同的消息生成负载下，消息包大小为 16　flits（256　bit）和 32　flits（512　bit）的平均通信延迟和网络的平均吞吐量，如图 5

和图 6 所示. 仿真结果表明,THIN 网络具有较低的网络延迟和较高的平均吞吐量,适于用来构建核间互连的片上网络.

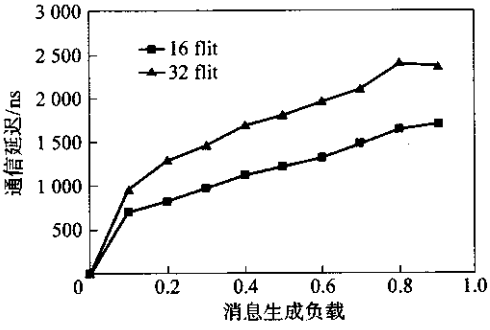

图 5 2 层 THIN 的平均通信延迟

Fig. 5 Average communication latency in $N(2)$

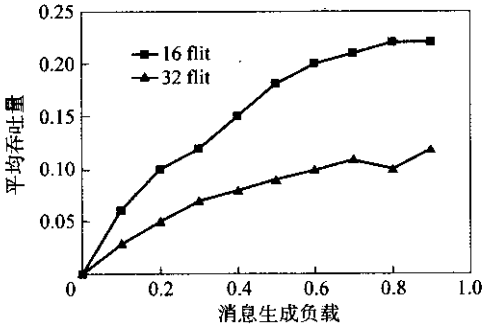

图 6 2 层 THIN 的平均吞吐量

Fig. 6 Average throughput in $N(2)$

7 结束语

作者提出一种用于多核处理器的新型片上互连网络——基三分层互连网络(THIN). 在综合考虑网络度、数据链路、网络直径和无阻塞延迟的情况下,当处理器核较少时,THIN 比 2-D Mesh 更宜于用来构建核间的通信网络. 针对 THIN 网络,作者提出一种分布式确定路由算法(DDRA),该算法设计简单,易于硬件实现,可以保证算法的高效性,从而实现传输速率高、低延迟的互连网络.

参考文献:

[1] Parcerisa J M, Sahuquillo J. On-chip interconnects and instruction steering schemes for clustered microarchitectures[J]. IEEE Transactions on Parallel and Distributed Systems, 2005, 16(2): 130 – 144.

[2] IBM. CoreConnect bus architecture[EB/OL]. (2005 – 04 – 15). http://www-03.ibm.com/chips/products/coreconnect/index.html.

[3] ARM. AMBA[EB/OL]. (2005 – 04 – 13). http://www.arm.com/products/solutions/AMBAHomePage.html.

[4] Wiklund D, Liu D. Design of a system-on-chip switched network and its design support[C]//Proceedings of IEEE 2002 International Conference on Communications, Circuits and Systems and West Sino Expositions. New York, NY, USA: IEEE Press, 2002: 1279 – 1283.

[5] Dally W J, Towles B. Route packets, not wires: on-chip interconnection networks[C]//Proceedings of the 38th Design Automation Conference. Las Vegas, USA: [s. n.], 2001: 681 – 689.

[6] Benini L, De Giovanni M. Networks on chips: a new SOC paradigm[J]. IEEE Computer, 2002, 35(1): 70 – 78.

[7] Kumar S. A network on chip Architecture and design methodology[C]//Proceedings of IEEE Computer Society Annual Symposium on VLSI. Pittsburgh, Pennsylvania, USA: [s. n.], 2002: 117 – 124.

[8] Liang J, Swaminathan S, Tessier R. aSOC: a scalable, single-chip communication architecture[C]//Proceedings of PACT 2000. Los Alamitos: IEEE Computer Society, 2000: 37 – 46.

[9] Duato J, Yalamanchili S, Ni L. Interconnection networks: An engineering approach[M]. Los Alamitos, CA, USA: The IEEE Computer Society Press, 1997.

[10] 董迎飞,王鼎兴,郑纬民. 精确计算 n 维 Mesh 网络和 n 维 Torus 网络的平均最短路经长度[J]. 计算机学报, 1997, 20(4): 376 – 380.
Dong Yingfei, Wang Dingxing, Zheng weimin. Exact computation of the mean minimal path length of n-Mesh and n-Torus[J]. Chinese Journal of Computers, 1997, 20(4): 376 – 380. (in Chinese)

[11] 高鹏. 并行数字信号处理机中的结点互连技术研究[D]. 哈尔滨: 哈尔滨工程大学水声工程学院, 2004.
Gao Peng. Research on interconnection networks of parallel signal processor[D]. Harbin: College of Underwater Acoustic Engineering, Harbin Engineering University, 2004. (in Chinese)

(责任编辑:康晓伟)