

一种改进的二维 Mesh 结构片上网络拓扑

姚放吾^{1,2}, 高莉¹

(1. 南京邮电大学 计算机学院, 江苏 南京 210003;

2. 南京邮电大学 计算机技术研究所, 江苏 南京 210003)

摘 要:随着多核技术的发展,片上网络(Network on Chip, NoC)越来越受到人们的关注。为了研究出更适用于片上网络的拓扑结构,在研究二维 mesh 结构的基础上,将二维 mesh 结构中每个 3×3 的小 mesh 里的对角线上的点用长边连接,形成改进后的拓扑,提出了在改进后的拓扑上的 16 节点的路由算法,并将改进后的拓扑与二维 mesh 结构做了性能分析。在 OPNET 下仿真结果表明,在同等网络规模下,改进后的拓扑较原来的二维 mesh 结构有更小的传输延迟和更大的吞吐量。

关键词:片上网络;拓扑;二维 mesh 结构;长边;性能分析

中图分类号:TP302

文献标识码:A

文章编号:1673-629X(2009)07-0024-04

An Improved 2D-Mesh NoC Topology and Routing Algorithm

YAO Fang-wu^{1,2}, GAO Li¹

(1. College of Computer, Nanjing University of Posts and Telecommunications, Nanjing 210003, China;

2. Institute of Computer Technology, Nanjing University of Posts and Telecommunications, Nanjing 210003, China)

Abstract:Based on the research of 2D-Mesh structure, linked nodes, which are located on the diagonal of each 3×3 small mesh, to improve the topology. Then proposed a 16-nodes routing algorithm for the improved topology, and conducted theoretical analysis on performance between the 2D-Mesh and the improved topology. OPNET simulation results showed that the improved topology has less transmission delay and more throughput than the original 2D-Mesh topology under the same network scale.

Key words:network on chip; topology; 2D-mesh; long edges; performance analysis

0 引言

近年来,随着多核技术的发展,片上网络越来越受到人们的关注。对于片上网络,其传输信息的能力主要依靠其拓扑结构^[1],拓扑结构定义了网络内节点与链路的布局 and 互连方式,它除了对网络延迟、吞吐量^[2]、面积、容错、功耗^[3]有很大影响外,对路由设计策略和内核到网络节点的映射也起着很重要的作用^[4,5]。

目前很多相关研究都是建立在经典的拓扑结构如二维 Mesh、Torus 之上的,这些结构一般常用于并行机网络,但片上网络通常比大规模并行机的规模要小,且在设计片上网络时要充分考虑物理实现难度,所以经典的拓扑结构并不一定能使性能达到最优。在研究 2D-Mesh 结构的基础上,针对其不足做出一些改进,并在改进后的拓扑结构上提出适合的路由算法。

1 2D-Mesh 结构描述

2D-Mesh 结构(如图 1 所示)是一种规则的二维网状结构^[6],该结构中每个非边缘交换节点有五个端口,其中四个连接邻居交换节点,一个连接 IP 核。每个边缘交换节点有三个端口,其中两个连接邻居交换节点,一个连接 IP 核。该拓扑结构简单,易于布置在芯片上,可扩展性好,便于设计简单高效的路由算法,是目前采用较为广泛的拓扑结构之一,但是该结构的网络直径和平均距离较大、跳数多,增加了数据包的延迟,因此当网络规模扩展后,该结构的功耗较大。

二维 Mesh 结构中的路由算法较多,在确定性算法中,较常用的有 XY 路由算法。为了更好地描述该算法,对一个 $N \times N$ 的网络结构,令源节点为 $C(cx, cy)$,目的节点为 $D(dx, dy)$, $X \text{ offset} = dx - cx$, $Y \text{ offset} = dy - cy$ 。

XY 路由算法的伪代码描述如下:

```
if (X offset != 0)
{
    if (X offset > 0) 向右走; else 向左走;
}
```

收稿日期:2008-10-29;修回日期:2009-01-23

基金项目:国家高技术研究发展计划(2006AA01Z208)

作者简介:姚放吾(1953-),男,教授,研究方向为计算机系统结构、计算机在通信中的应用。

else if(Y offset>0) 向上走;
else if(Y offset<0) 向下走;
else 不需移动;

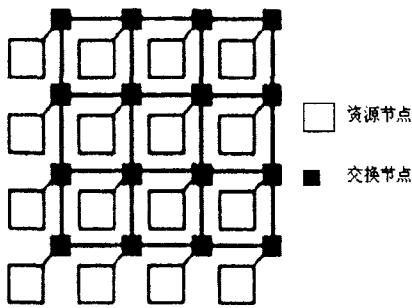


图 1 2D-Mesh 结构

2 改进后的拓扑结构及与 2D-Mesh 结构的比较

在规则的二维网状结构上增加一些长距离的边可以缩短节点间距离^[7],减少路由延时,从而提高了网络的性能。因此针对 2D-Mesh 结构的特点,改进了其拓扑结构,改进后的拓扑结构定义为:在 $N \times N$ 的 2D-Mesh 结构上,将每个 3×3 的小 mesh 里的对角线上的点用长边连接起来。图 2 是一个改进后的 4×4 拓扑,与二维 Mesh 结构相比,增加了 $2N$ 条对角线上的长边。为了分析比较,将改进后的拓扑结构图用坐标进行了标识。

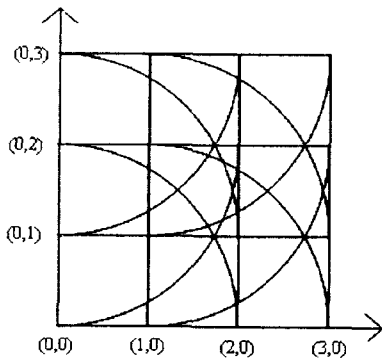


图 2 改进后的拓扑结构

2.1 网络直径

网络直径是指网络中任意两节点间最短距离的最大值^[8]。网络直径与网络的通讯延迟成线性正比关系,直径的减小可以提高网络传输的质量和速度。在 $N \times N$ 的网络中,Mesh 结构的网络直径是 $2N - 2$,改进后的拓扑的网络直径是 $N - 1$ 。

2.2 消息延时

假设消息的数据为 L 位,微片的大小等于物理数据通道的宽度 W 位,消息头设为一个微片,则消息的大小是 $L + W$ 位,使用虫孔交换技术,在无阻塞的情

况下消息延迟公式为^[9]:

$$t_{wormhole} = D(t_r + t_s + t_w) + \max(t_s, t_w) \lceil L/W \rceil$$
(1)

公式中 t_r 是路由器做出路由决策的时间; t_s 是 W 位的微片从路由器的输入传送到输出的时间,即路由器的交换延迟; t_w 是通道延迟, D 是节点间的平均距离。文中两种拓扑结构,路由器结构相同,并且使用相同通道,当消息大小相等时延迟取决于 D 。 $N \times N$ 的 Mesh 结构节点间平均距离为^[10]:

$$D_{2D-Mesh} = 2(\sqrt{2} - 1)/3N$$
(2)

当 $N = 4$ 时, $D_{2D-Mesh} = 2.5$

使用改进后的拓扑,当 $N = 4$ 时,按照距离远近,可以将节点分为距离 1、距离 2、距离 3 三类,见表 1。与 $D(0,0)$ 一类的点有 4 个,与 $D(1,0)$ 一类的点有 8 个,与 $D(1,1)$ 一类的点有 4 个,所以

$$D \text{ 改进后的} = (D(0,0) \times 4 + D(1,0) \times 8 + D(1,1) \times 4) / (16 \times 16) = 1.797$$

改进后的拓扑的节点平均距离小于 2D-Mesh,理论上,改进后的拓扑比 Mesh 结构的平均延时要小。

表 1 4×4 改进后的拓扑 3 类节点
与其他节点的距离

| 源节点 | (0,0) | (1,0) | (1,1) |
|---------------|---|--|--|
| 与源节点距离为 1 的节点 | (0,1)(1,0) (2,2) | (0,0)(2,0)(1,1) (3,2) | (0,1)(1,0)(1,2) (2,1)(3,3) |
| 与源节点距离为 2 的节点 | (0,2)(1,1) (1,2)(2,0) (2,1)(2,3) (3,2) | (0,1)(0,2)(1,2) (2,1)(2,2)(3,0) (3,1)(3,3) | (0,0)(0,2)(0,3) (1,3)(2,0)(2,2) (2,3)(3,0)(3,1) (3,2) |
| 与源节点距离为 3 的节点 | (0,3)(1,3) (3,0)(3,1) (3,3) | (0,3)(1,3)(2,3) | |

2.3 理想吞吐量

理想吞吐量是指对于给定的拓扑结构,在完美的流量控制和路由机制下,网络中的最大吞吐量^[8]。其计算公式为:

$$TP \leq 2bBc/N^2$$
(3)

Bc 是把整个网络划分成均等的两部分所需要的通道个数, b 是每个通道的带宽。对于 4×4 的二维 Mesh,把它分成均等的两部分需要断开 4 个通道,每个通道又是双向的,所以 Bc 为 8;对于 4×4 的改进后的拓扑不仅要断开二维 Mesh 原有的 4 个通道,还有另外 8 个长边通道,因此 Bc 为 24,由公式可得 $TP_{Mesh} \leq b$, TP 改进后的 $\leq 3b$,改进后的拓扑理想吞吐量比二维 Mesh 的要大。

2.4 物理实现难度

物理实现难度主要取决于结构是否对称、结构的维数、长边的个数、路由 switch 端口数以及路由算法复

杂度。

Mesh 和改进后的拓扑都是二维结构, Mesh 结构中所有互连边的长度都很短, 并且 switch 的端口数最大的只为 4, 路由算法简单, 因此物理实现难度最低; 改进后的拓扑, 增加了每个小 Mesh 结构里对角线上的长边, 当 N 大于 4 时, 路由 switch 端口数最大的为 8, 且路由算法比 Mesh 的复杂, 因此它的物理实现难度高于 Mesh。

3 改进后拓扑的 16 节点路由算法

2D-Mesh 结构中最简单的路由算法是 XY 路由, 而改进后的拓扑是在 2D-Mesh 结构的基础上增加了长边, 因此改进后的拓扑路由算法也可以在 XY 路由的基础上进行设计, 该拓扑的路由算法是一种确定性最短路径算法, 令 Cdirection 为源节点所在的长边方向, CDdirection 为目的节点相对于源节点的方向, Cdirection 方向共有四种可能性: 左上、左下、右上、右下。伪代码描述:

```

if (X offset == 0 || Y offset == 0)
{
    用 XY 路由算法;
}
else if (abs(X offset) == 2 && abs(Y offset) == 2)
{
    则直接走对角线上的长边;
}
else if (abs(X offset) >= N/2 || abs(Y offset) >= N/2)
{
    if (Cdirection 与 CDdirection 相同)
    {
        走从源节点出发的长边;
        XY 路由算法;
    }
    else if (Cdirection 与 CDdirection 同左或者同下)
    {
        if (X offset > 0)
        {
            向左走;
        }
        else
        {
            向右走;
        }
        找到第一个与 CDdirection 方向相同的长边并走该长边;
        XY 路由算法;
    }
    else if (Cdirection 与 CDdirection 同左或者同右)
    {

```

```

if (Y offset > 0)
{
    向下走;
}
else
{
    向上走;
}

```

找到第一个与 CDdirection 方向相同的长边并走该长边;

XY 路由算法;

```

}
else
{
    XY 路由算法;
}

```

4 仿真验证与分析

使用 OPNET 建立 2D-Mesh 和改进后的拓扑的仿真模型: 网络拓扑按照图 1 和图 2 建立; 数据包的长度为 16 位, 其中包括源、目的 X 和 Y 地址和负载内容; 资源节点使用可以产生数据包, 并且还可以销毁接收到的数据包的处理程序原型; 交换节点路由采用上述路由算法, 使用虫孔交换技术; 交换节点之间、交换节点与资源节点之间的通道即链路使用同一种恒定速率的全双工通信方式, 支持所定义的数据包类型, 链路的数据宽度为 16bits。在确定了基本的网络模型之后, 还定义了如下的通信场景: 所有的网络节点并发工作, 假设链路的错误率为零, 速率设为 1024bps, 每个网络节点所生成的流量均匀地分布在所有其它的网络节点之间, 即数据包中目标地址在指定区间内随机生成均匀分布的整数。图 3~图 6 分别是仿真三十分钟后, 不同网络规模下, 两种拓扑的平均传输延迟和平均吞吐量结果。

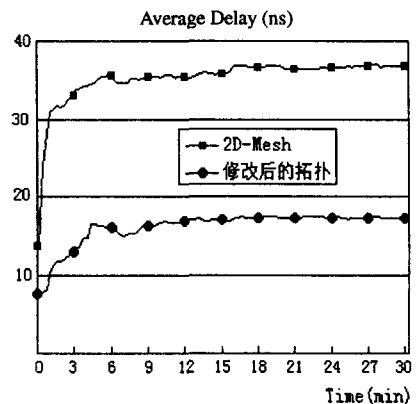
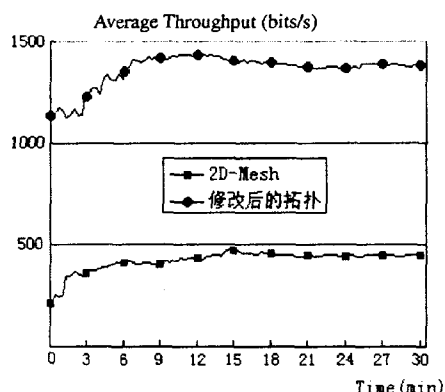
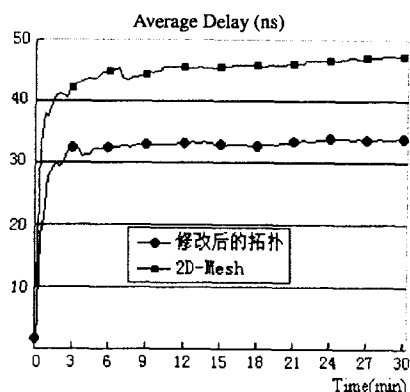
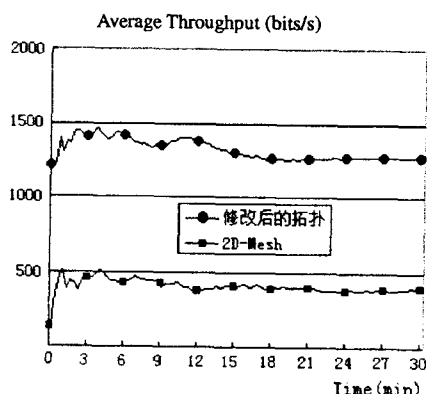


图 3 3×3 规模两种拓扑的平均延迟仿真结果表明, 在同等网络规模下, 改进后的拓扑

比 2D-Mesh 结构有更小的平均传输延迟和更好的吞吐量,原因在于,改进后的拓扑有更小的网络直径和节点间的通信距离,如 4×4 的规模下,两个对角上相距最远的节点,改进后拓扑的通信距离为 3,而 2D-Mesh 结构为 6;从 3×3 和 4×4 的规模来看,随着网络规模的扩大,无论是 2D-Mesh 结构还是改进后的拓扑延迟都会增大,而两者之间的延迟差越来越小,但对网络的吞吐量的影响不是太大。以上结果表明改进后的拓扑在小规模的片上网络上性能优于 2D-Mesh 结构。

图4 3×3 规模两种拓扑的平均吞吐量图5 4×4 规模两种拓扑的平均延迟图6 4×4 规模两种拓扑的平均吞吐量

5 结束语

在研究 2D-Mesh 结构的基础上对其拓扑做了改进并提出了 16 节点的路由算法。在同等网络规模下,改进后的拓扑比 2D-Mesh 结构有更小的传输延迟和更大的吞吐量,性能更优越,但还需考虑改进后的拓扑结构物理实现难度和路由算法的自适应性。

参考文献:

- [1] Srinivasan K, Chatha K S, Konjevod G. Linear programming based techniques for synthesis of Network-on-Chip architectures[J]. IEEE Transactions on Very Large Scale Integration (VLSI) Systems, 2006, 14(4): 407-420.
- [2] Kreutz M E, Carro L, Zeferino C A, et al. Communication architectures for system-on-chip[C]// Integrated Circuits and Systems Design, The 14th Symposium on Integrated Circuits and Systems Design. [s.l.]: [s.n.], 2001: 14-19.
- [3] Hung W, Addo Q C, Theodorides T, et al. Thermal-aware IP virtualization and Placement for Networks-on-Chip architecture[C]// Computer Design: VLSI in Computers and Processors, 2004. ICCD 2004. Proceedings. IEEE International Conference. [s.l.]: [s.n.], 2004: 430-437.
- [4] Ogras U Y, Hu J, Marculescu R. Key research problems in NoC design: A holistic perspective[C]// Proceedings of the 3rd IEEE/ACM/IFIP International Conference on Hardware / Software Code sign and System Synthesis. New York, NY, USA: ACM Press, 2005: 69-74.
- [5] Pande P P, Grecu C, Jones M, et al. Performance Evaluation and Design Trade-Offs for Network-on-Chip Interconnect Architectures[J]. IEEE Transactions on Computers, 2005, 54: 1024-1040.
- [6] Kumar S. A Network on Chip Architecture and Design Methodology[C]// VLSI, 2002. Proceedings. IEEE Computer Society Annual Symposium. [s.l.]: [s.n.], 2002: 105-112.
- [7] Ogras U Y, Marculescu R. It's a small world after all, NoC performance optimization via long link insertion [J/OL]. IEEE Transactions on Very Large Scale Integration (VLSI) Systems, 2006, 14(7). [s.l.]: [s.n.], http://www.ece.cmu.edu/~sld/pubs/papers/TVLSI_July06_Special_issue.pdf.
- [8] 朱晓静, 胡伟武, 马可, 等. Xmesh: 一个 mesh-like 片上网络拓扑结构[J]. 软件学报, 2007, 18(9): 2194-2204.
- [9] Duato J, Yalamanchili S, Ni L. 并行计算机互连网络技术: 一种工程方法[M]. 谢伦国, 张民选, 窦强, 等译. 北京: 电子工业出版社, 2004.
- [10] 董迎飞, 王鼎兴, 郑纬民. 精确计算 n 维 Mesh 网络和 n 维 Torus 网络的平均最短路径长度[J]. 计算机学报, 1997, 20(4): 376-380.