

2D Mesh 结构中一种新的自适应路由模式*

张麟麟, 李东生, 雍爱霞

(电子工程学院, 合肥 230037)

摘要: 针对 2D Mesh 拓扑结构中的损坏节点会导致静态 XY 路由算法无法有效传输的问题, 提出了一种新的路由算法——FTXY 路由算法。首先对网络拓扑结构中的平均延时、理想平均吞吐量和能量消耗进行了理论评估, 然后在 NIRGAM 仿真软件平台上采用 FTXY 路由算法对上述三个参数进行验证, 并与 XY 路由算法进行比较。实验结果显示, 新的路由算法可以有效地绕过损坏节点, 并且不会造成阻塞, 提高了网络的传输性能。

关键词: 片上网络; NIRGAM 仿真; 路由算法; 平均延时; 平均吞吐量; 能量消耗; 2D Mesh

中图分类号: TN47 **文献标志码:** A **文章编号:** 1001-3695(2012)05-1828-04

doi:10.3969/j.issn.1001-3695.2012.05.060

New adaptive routing pattern for 2D Mesh structure

ZHANG Lin-lin, LI Dong-sheng, YONG Ai-xia

(Electronic Engineering Institute, Hefei 230037, China)

Abstract: For the problem that the static XY routing algorithm could not transmit effectively on account of the damaged nodes in 2D Mesh topology, this paper introduced a new routing algorithm: FTXY routing algorithm. Firstly, it evaluated the average latency, average throughput and the energy consumption theoretically, then used FTXY routing algorithm to validate the above three parameters in NIRGAM software, and compared with the XY routing algorithm. The experimental results show that the new routing algorithm can effectively round the damaged nodes and it cannot induce block, thus improving the transmission performance of the network.

Key words: network on chip (NoC); NIRGAM simulation; routing algorithm; average latency; average throughput; energy consumption; 2D Mesh

0 引言

片上网络的研究开始于 1999 年, 初期探索 SoC 通信部分的系统级设计方法, 之后不久便涉及从物理层到体系结构、操作系统以及应用等各个方面。集成电路 NoC 技术的核心思想是将计算机网络技术移植到芯片设计中, 以彻底解决多 CPU 的体系结构问题。目前 NoC 的概念很广泛, 包括硬件通信结构、中间件、操作系统通信服务, 以及设计方法和工具等。从系统角度看, NoC 研究的重点在于拓扑结构、通信协议、服务质量和系统低功耗等。

目前对片上网络路由的研究主要是针对普通的拓扑结构如何提高其传输性能。文献[1,2]提出了一种路由方式来提高片上网络的传输质量(QoS), 但对路由的容错都没有涉及。文献[3]针对不规则 2D Mesh 提出了一种路由算法, 并在两种不同模式下进行了仿真验证。文献[4]运用 NS2 仿真软件对片上网络的不同拓扑结构进行了评估, 并给出了不同拓扑结构的适用范围。文献[5]主要在基于热点负载情况下设计了一种路由模式, 通过仿真片上网络的性能参数, 来分析如何提高片上网络的数据流传输性能。文献[6]通过调节网络中虚拟通道的大小, 设计了一种路由来提高服务质量。

本文在研究 2D Mesh 结构及其主流路由算法的基础上, 提

出一种新的路由算法, 采用的拓扑结构为 5×5 的 2D Mesh 结构。如果网络中有损坏节点时, 在该路由算法下, 数据流依然可以到达目标节点, 并且有较好的传输性能。

1 基本概念

1.1 2D Mesh 拓扑结构

如图 1 所示, 2D Mesh 结构作为一种最简单、最直观的拓扑结构, 每个节点连接着一个资源和四个相邻的路由器, 每个资源通过一个网络接口(NI)连接着一个路由器。其中的资源可以是一个处理核、内存、一个用户自定义硬件模块, 也可以是其他任何可以插入插槽且可以与网络接口相匹配的 IP 模块。路由器之间、路由器与资源之间是由一对输入与输出通道连接, 通道是由两条单向的点对点总线组成。2D Mesh 结构是 NoC 中研究得最早的一种拓扑结构, 其结构规则、简单, 易于实现。

1.2 2D Mesh 拓扑结构的主流路由算法

NoC 的拓扑结构必须保证每个节点可以发送数据包到其他节点。当没有完善的拓扑结构时, 路由算法决定数据包从源地址开始选择哪一条路径到目的地址。所以一个有效的路由算法对 NoC 网络性能的好坏是至关重要的。

路由算法既可以是确定性路由, 也可以是自适应路由。确

收稿日期: 2011-09-04; 修回日期: 2011-10-08 基金项目: 国家自然科学基金资助项目(61179036); 中国博士后科学基金资助项目(20090461419)

作者简介: 张麟麟(1980-), 男, 硕士研究生, 主要研究方向为片上网络(zhanglin 780@sohu.com); 李东生(1963-), 男, 博导, 主要研究方向为多核芯片设计、高速信号处理; 雍爱霞(1974-), 女, 博士, 主要研究方向为片上网络。

定性路由算法是在给定源节点—目的节点的前提下确定路径;而自适应路由则是根据网络流量或通道使用状态,为避免网络拥塞或提高网络性能的自主择径。

1.2.1 自适应路由

确定路由的优点是路由算法简单,在网络低拥塞环境下能获得较低延迟。但是由于其不能响应动态的网络状态变化,所以当网络拥塞时,性能迅速降低。所谓自适应路由,就是指数据包的路由路径不仅与起点地址和终点地址有关,还要考虑网络状态。也就是说,有同一对起点和终点地址的数据包,在不同的网络状态下,它们的路由路径也可能不同。

自适应路由的优点是采用自适应路由的路径,避免了网络拥塞,可以得到更高的网络带宽饱和值;但其路由逻辑较复杂,在网络低拥塞的情况下开销较大,而且还存在死锁问题。

1.2.2 XY 路由

2D Mesh 中的 XY 路由是一种最简单的确定性路由算法。其路由过程是:先沿 X 方向将数据包送至目的节点所在的列;再沿 Y 方向将数据包送至目的节点所在的行。当路由器开始处理一个数据包时,因为采用的是 XY 路由,它会判断在 X 方向上是否到达了目的节点所在的列。如果没到达,再比较目的 X 和当前的 X,如果目的 X 比当前的 X 大,则应该向东路由;如果目的 X 比当前的 X 小,则说明应该向西路由。如果判断出 X 方向上已经到达了目的节点所在的列,则比较目的 Y 和当前的 Y,如果相等,则说明该数据包已经到达了目的地;如果目的 Y 大于当前的 Y,则还应该向南方路由;如果目的 Y 小于当前的 Y,则说明还应该向北方路由。具体举例来说,一个从(1,2)发出的数据包,其目的地是(3,4),则它采用 XY 路由算法的路由路径是(1,2)→(2,2)→(3,2)→(3,3)→(3,4)。

2 FTXY 路由算法

普通的 XY 路由算法并没有考虑到如果网络中有节点出现故障时数据流将怎样传输。图2为一个5×5的2D Mesh 拓扑结构,节点6、7、12、18为网络中损坏的节点。

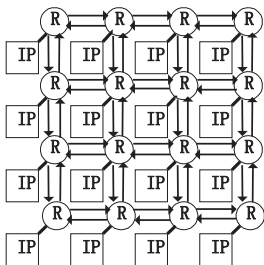


图1 2D Mesh拓扑结构

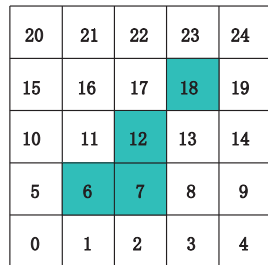


图2 5x5的2D Mesh拓扑结构

如果假设节点0要向节点24传输数据,按照普通的XY路由模式应该是0→1→2→3→4→9→14→19→24。但是由于拓扑结构中有损坏节点,普通XY路由不能保证其传输质量,因此本文提出一种新的路由算法FTXY,来解决由于节点损坏而不能保证其传输质量的问题。算法的详细步骤如下:

a) 首先定义(S_x, S_y)为原节点位置坐标;(D_x, D_y)为目的节点位置坐标;id_E为原节点东边的节点;id_S为原节点南边的节点;id_W为原节点西边的节点;id_N为原节点北边的节点。

b) 以下是FTXY对应的路由算法步骤编码。

(a) 如果没有损坏的节点,路由算法为普通的XY路由算法:

```
if (Dx == Sx && Dy == Sy) return C;
if (Dy > Sy) dirx = W;
if (Dx < Sx) diry = N;
else if (Dx > Sx) diry = S;
(b) 如果节点本身是标记的损坏节点,路由算法为:
for (int i=0; i<failno; i++) {
if (id == fail[i][0]) {
if (id_E == fail[i][1] && dirx == E) {
dirx = ND; }
if (id_S == fail[i][1] && diry == S) {
diry = ND; }
}
if (id == fail[i][1]) {
if (id_W == fail[i][0] && dirx == W) {
dirx = ND; }
if (id_N == fail[i][0] && diry == N) {
diry = ND; } } }
(c) 结束本身的标记节点检查:
if (dirx == ND)
return diry;
if (diry == ND)
return dirx;
(d) 如果标记节点在X方向:
if (dirx == E) {
for (int i=0; i<failno; i++) {
if (id_E == fail[i][0] || id_E == fail[i][1]) {
markx = 1; } } }
else if (dirx == W) {
for (int i=0; i<failno; i++) {
if (id_W == fail[i][0] || id_W == fail[i][1]) {
markx = 1; } } }
(e) 如果标记节点在Y方向:
if (diry == N) {
for (int i=0; i<failno; i++) {
if (id_N == fail[i][0] || id_N == fail[i][1]) {
marky = 1; } } }
else if (diry == S) {
for (int i=0; i<failno; i++) {
if (id_S == fail[i][0] || id_S == fail[i][1]) {
marky = 1; } } }
(f) 确定最后的方向收包:
if (markx == 0 && marky == 0) choose = (id%2);
if (markx == 0 && marky == 1) choose = 0;
if (markx == 1 && marky == 0) choose = 1;
if (markx == 1 && marky == 1) choose = (id%2);
switch (choose)
{ case 0: return dirx;break;
case 1: return diry;break; }
return 0; } //choose :0 for X-dir and 1 for Y-dir
```

3 2D Mesh 拓扑结构性能的理论评估

对 NoC 拓扑结构中数据流性能参数的评估,目前主要是从理想平均延时、吞吐量及能量方面进行网络性能的理论分析,从而得出数据流在拓扑中传输的理论性能。

针对2D Mesh 拓扑结构,本文从理想平均延时、吞吐量及能量方面进行网络性能的理论分析,从而得出2D Mesh 结构的理论性能。

3.1 理想平均延时

理想平均延时是指给定一个拓扑结构,假定在路由过程中完全没有拥塞,这个结构所有节点对之间的平均路由延时即为

理想平均延时。文献[7,8]给出了一个评估理想平均延时的公式:

$$T = H \times Tr + D/v + L/b$$

其中: Tr 是在路由 switch 上的延时拍数,单位是 cycle/hop; H 是指从源节点到目标节点的平均 hop 数; D 是从源节点到目标节点的平均距离,通常情况下, $H = D$,单位是 hop; L 是包的长度,单位是 flit(流控单元); v 是在线上的传输速度,单位是 hop/cycle; b 是带宽,单位是 flit/cycle。在不同的拓扑结构中, Tr 依赖于 switch 的物理实现, H 与 D 的值是不一样的。计算平均延时主要需要 H ,也就是 D 。文献[7]给出了 2D Mesh 结构的节点间平均距离公式。假定网络规模是 $n \times n$,如下所示:

$$H_{2DM} = 2n/3$$

这个公式假定源节点与目标节点不相同,但如果假定目标节点与源节点是可以相同的,则这个公式改动为

$$H_{2DM} = (2n/3) \times (n^2 - 1)/n^2$$

因此 2D Mesh 结构的理想平均延时为

$$T_{2DM} = H_{2DM} \times Tr + H_{2DM}/v + L/b$$

3.2 理想吞吐量

理想吞吐量是指对于给定拓扑结构,在完美的流控和路由机制下,网络中的最大吞吐量。文献[7]给出了一个计算理想吞吐量的公式:

$$TH \leq 2b \times Bc/N$$

其中: Bc 是把整个网络分成均等的两半所需要的通道个数; b 是每个通道的数据宽度; N 是节点的总数。

假设对于 5×5 的 2D Mesh,要想把 Mesh 分为两半,需要切开 5 条边,每个边又都是双向的,所以 $Bc = 10$,又因为 $N = 25$,所以可以得到 5×5 的 2D Mesh 的理想吞吐量为

$$TH_{2DM} \leq \frac{4}{5}b$$

3.3 能量消耗

NoC 的能量消耗模型是最近才开始研究的一个方向,文献[9]提出了一个网络路由器的能量消耗模型。设 E_{bit} 是单位数据穿过路由器的动态能量消耗,则有

$$E_{bit} = E_{Sbit} + E_{Bbit} + E_{Wbit}$$

其中: E_{Sbit} 为 1 bit 数据在节点开关矩阵的动态能量消耗; E_{Bbit} 为 1 bit 数据在节点缓存的动态能量消耗; E_{Wbit} 为 1 bit 数据在节点内部连线的动态能量消耗。

根据以上公式,Mesh 网络的每一个模块由一个资源节点和一个通信节点组成,通信节点与四个相邻的通信节点和本地资源节点通过通道相连,组成点到点的连接。

从模块 m_i 到 m_j 发送 1 bit 数据平均消耗的能量为^[9]

$$E_{bit}^{m_i, m_j} = H_{hops} \times E_{bit} + (H_{hops} - 1) \times E_{Lbit}$$

其中: H_{hops} 为 1 bit 数据从模块 m_i 到 m_j 所遍历的中间节点数; E_{Lbit} 为 1 bit 数据在模块与模块之间的链路上消耗的能量。则发送一个完整的分组所消耗的能量为 $E_{packet} = n \times E_{bit}^{m_i, m_j}$, n 为分组的长度。

4 仿真环境和条件设置及结果分析

4.1 仿真环境设置

本文所用的 NIRGAM 仿真软平台^[10]是英国南开普敦大学基于 SystemC 语言开发的一个面向 NoC 研究的仿真平台,用

户可以通过 NIRGAM 设置仿真的各种参数以达到仿真效果,并可以自己定制拓扑结构和路由机制来实现仿真目的。模拟器的参数设置如表 1 所示。

表 1 仿真器的基本参数

parameter name	valid values	description
TOPOLOGY	Mesh	
NUM_ROWS	5	5 × 5 的 2D Mesh 拓扑结构
NUM_COLS	5	
virtual channels	4	四个虚拟通道
RT_ALGO	XY	静态 XY 路由模式
WARMUP	5	warmup period/clock cycles
SIM_NUM	5 000	simulation cycles
TG_NUM	3 000	clock cycles until which traffic is generated
HEAD_PAYLOAD	1	payload size (in bytes) in head/hdt flit
DATA_PAYLOAD	4	payload size (in bytes) in data/tail flit
CLK_FREQ	1	clock frequency/GHz

实验中设定数据流的大小和方向如表 2 所示。

表 2 数据流的大小和方向

parameter name	0	9
Application library	CBR	CBR
PKT_SIZE	8	8
LOAD	100	100
DESTINATION FIXED	24	13
FLIT_INTERVAL	2	2

表中从节点 0 和 9 发出的数据流类型均为固定比特率流量(CBR)。节点 0 数据流向为节点 24,节点 9 数据流向为节点 13。各参数的定义如下:

PKT_SIZE: packet size in bytes

LOAD: percentage of maximum bandwidth being used

DESTINATION FIXED: destination tile ID

FLIT_INTERVAL: interval between successive flits in clock cycles

4.2 仿真条件设置

通过比较 FTXY 路由和 XY 路由模式下片上网络的各项参数,来验证本文所设计的 FTXY 路由能否有效避开损坏节点,以及能否提高传输性能。

4.3 仿真结果分析

如图 3 所示,作如下假设:在节点 0 与 1 之间,红色(见电子版)表示向东路由,即节点 0→节点 1,蓝色表示向西路由,即节点 1→节点 0;同理,在节点 0 与 5 之间,绿色表示向北路由,即节点 0→节点 5,黄色表示向南路由,即节点 5→节点 0。

4.3.1 Flit 的平均延时

表 3 显示了在两种模式下,一个时钟周期内 flit 的各项平均延时比较。从比较结果可以看出,FTXY 路由的 flit 平均延时要小于 XY 路由。

表 3 一个时钟周期内 flit 的平均延时比较

parameter name	FTXY	XY
latency per channel	2.584 67	2.694 44

图 4 和 5 显示了在传输过程中各节点的 flit 传输延时比较。从图 4 和 5 的比较中可以看出,在传输过程中,两种路由模式下 flit 传输延迟都比较平稳,但 FTXY 路由模式下 flit 的总体延迟要小于 XY 路由模式。

4.3.2 Packet 的平均延时

表 4 显示了在两种模式下,时钟周期内 packet 的平均延时

比较。从比较结果可以看出,FTXY 路由的 packet 平均延时也要小于 XY 路由。

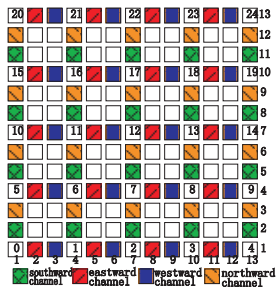


图3 仿真方向表示

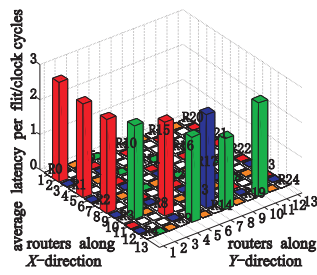


图4 FTXY 数据传输过程中节点的flit传输延时

表4 时钟周期内 packet 的平均延时比较

parameter name	FTXY	XY
latency per channel	7.754	8.083 33

图6和7显示了在传输过程中各节点的 packet 传输延时比较。

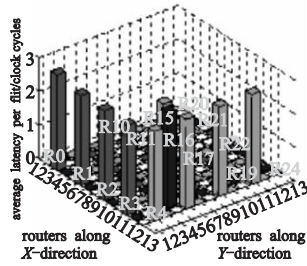


图5 XY 数据传输过程中节点的flit传输延时

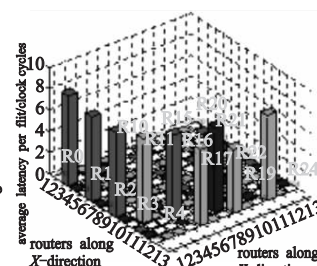


图6 FTXY 数据传输过程中节点的packet传输延时

从图6和7的比较中可以看出,在传输过程中,两种路由模式下 packet 传输延迟也都比较平稳,但在 FTXY 路由模式下 packet 的总体延迟也要小于 XY 路由模式。

4.3.3 能量消耗

表5显示了在两种模式下能量损耗的比较。从比较结果中可看出,FTXY 路由的能量消耗与 XY 路由相差不多。

表5 能量消耗比较

parameter name	FTXY	XY
total network power/watt	0.056 554 7	0.056 523 4

图8和9显示了在传输过程中各节点能量损耗的比较。

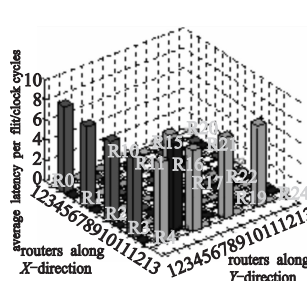


图7 XY 数据传输过程中节点的packet传输延时

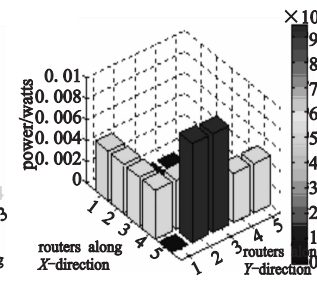


图8 FTXY 数据传输过程中节点的能量损耗

从图8和9的比较中可以看出,在传输过程中,两种路由模式下各节点的能量消耗相差不多。但在 FTXY 路由模式下,14 节点的能量损耗高于 XY 路由模式下;而在 XY 路由模式下节点4 能量损耗较高。

4.3.4 理想吞吐量

表6显示了在两种模式下传输过程中平均吞吐量的比较。从比较结果可以看出,在 FTXY 路由模式下的平均吞吐量要明显好于 XY 路由模式。

表6 平均吞吐量的比较

parameter name	FTXY	XY
average throughput	9.307 80	8.844 24

图10和11显示了在传输过程中各节点平均吞吐量的比较。

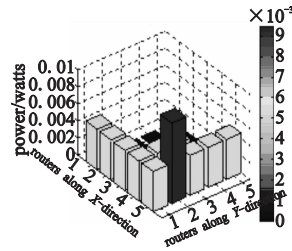


图9 XY 数据传输过程中节点的能量损耗

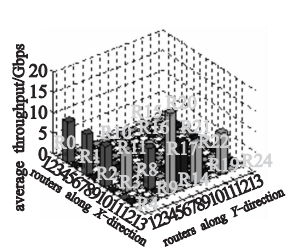


图10 FTXY 数据传输过程中节点的平均吞吐量

从图10和11的比较中可以看出,在传输过程中,FTXY 路由模式下各节点的平均吞吐量也要明显好于 XY 路由模式。

4.3.5 数据流动方向

由以上仿真可以看出,在2D Mesh 拓扑结构中如果存在损坏节点,在 FTXY 路由算法下,数据流依然可以达到目的节点,其流动方向如图12所示。

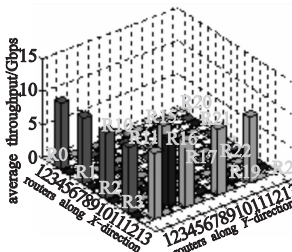


图11 XY 数据传输过程中节点的平均吞吐量

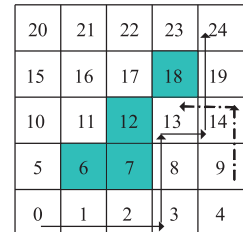


图12 数据的流动方向

5 结束语

本文针对2D Mesh 拓扑结构中如果存在损坏节点,静态路由中的 XY 路由不能很好运用的问题,提出了一种新的路由算法——FTXY 路由算法。首先从理论上对2D Mesh 拓扑结构中理想平均延时、理想吞吐量、能量消耗进行分析,最后运用NIRGAM 仿真软平台在 FTXY 路由算法下,对三个参数进行仿真。结果表明,FTXY 路由算法确实可以绕开损坏节点,使数据流顺利到达目标节点,并且具有较好的传输特性。

参考文献:

- [1] LUSALA A K, LEGAT J D. A hybrid router combining SDM-based circuit switching with packet switching for on-chip network[C]//Proc of International Conference on Reconfigurable Computing and FPGAs. [S.l.]: IEEE Press,2010:340-345.
- [2] DIEMER J, EMST R. Back suction: service guarantees for latency-sensitive on-chip networks[C]//Proc of the 4th ACM/IEEE International Symposium on Networks-on-Chip. [S.l.]: IEEE Press,2010: 155-162.
- [3] WANG Ling, SONG Hui, WEN Dong-xin, et al. A degree priority routing algorithm for irregular Mesh topology NoC[C]//Proc of International Conference on Embedded Software and Systems. [S.l.]: IEEE Press,2008:183-188.
- [4] 杨智峰,田泽.用 NS2 评估片上网络体系结构的性能[J].计算机工程与应用,2010,46(18):74-77.

(下转第1890页)

(上接第 1831 页)

- [5] YUNUS S A M J, MARSONO M N, IBRAHIM I. Modeling router hotspots on network-on-chip[C]//Proc of the 13th International Conference on Advanced Communication Technology. [S. l.]: IEEE Press, 2011:896-900.
- [6] HORCHANI M, ATRI M, TOURKI R. Design of a NoC-router guaranteeing QoS based on virtual channels reservation[C]//Proc of International Conference on Signals, Circuits and Systems. [S. l.]: IEEE Press, 2008:1-6.
- [7] DALLY J, TOWLESS B. Principles and practices of interconnection network[M]. [S. l.]: Morgan Kaufmann Publishers, 2003.
- [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] TERRY T Y. On-chip multiprocessor communication network design and analysis[D]. Stanford: Stanford University, 2005.
- [10] A simulator for NoC interconnect routing and application modeling [EB/OL]. (2007-01-10) [2011-05-29]. <http://nirgam.ecc.soton.ac.uk/2007/01/NIRGAM>.