

可设置仲裁优先程度的NOC路由节点设计

武 畅, 李玉柏, 彭启琮, 柴 松, 杨中明

(电子科技大学通信与信息工程学院DSP技术中心 成都 610054)

【摘要】针对片上网络(NOC)具体应用中各个IP之间出现不同的通信情况的问题,提出了一种可设置仲裁优先程度的NOC路由节点。该节点采用基于Lottery算法的仲裁机制,取代了目前NOC路由节点中广泛采用的轮询调度(RR)仲裁机制,可以根据具体应用的通信状况来设定仲裁响应的优先程度。在NOC路由节点中,设计了内部TUM(时间单元复用)的传输机制来解决不同输入口竞争同一输出口所带来的节点内部阻塞问题,使得即使在有多个端口竞争同一端口的情况下,路由节点仍然能保证输入的数据包和进入输出缓冲的数据包数量上的一致,大大提高了路由节点的处理性能。

关键词 片上网络; 路由节点; 仲裁器; 优先级; 仿真
中图分类号 TN929 **文献标识码** A

Design of a Customized Arbitral Priority NOC Router

WU Chang, LI Yu-bai, PENG Qi-cong, CHAI Song, and YANG Zhong-ming

(DSP Center of School of Communication and Information Engineering, University of Electronic Science and Technology of China Chengdu 610054)

Abstract For the different communication cases of specific network on chip (NOC) applications, a customized arbitral priority NOC router is proposed. This router uses the arbitral mechanism based on lottery algorithm instead of the round robin (RR) algorithm, which is widely used in the arbiter of current NOC routers. The arbitral priority of lottery router can be customized by users according to the communication cases among IPs in NOC. This router also includes a time unit multiplex (TUM) transmission mechanism to alleviate the problem of internal block, which may be induced by the case of several input ports competing for one output port. The TUM mechanism guarantees the number of input data is equal to the number of output data and improves the router processing performance.

Key words arbiter; NOC; priority; router; simulation

随着工艺技术和集成水平的高速发展,使得在单个芯片上集成几十亿个晶体管成为可能^[1],单个芯片的计算和处理能力也大大提高,同时,大量的片上资源也使SoC设计面临更多的挑战。如在更多IP协同工作中的性能提高和功耗的降低,在系统层次中,复杂的单片结构也使得资源的复用性、适应性和扩展性成为制约集成度提高的重要因素^[2]。解决这些问题的有效方法就是在单一芯片上建立一个微网络,形成片上网络的结构^[3-4]。目前研究得最广泛的是2D Mesh结构,它的每一个IP通过NI(network interface)接口与路由节点相连接。本文所研究的主要对象是NOC的路由节点,它是承担数据交换和IP之间通信的最主要部分^[5]。

在具体应用中,可以通过多种方式进行优化以提高NOC性能,如拓扑结构的变换、路由算法的改

变、缓冲大小的重新分配。其中,与路由节点结构直接相关的是缓冲大小的分配,如在具体应用中,某两个IP之间比其他节点具有更大的通信量或者需要更宽的通信带宽,那么就可以通过增加这两个IP通信信道之间的缓冲大小来达到减小延时,提高性能的目的。但是,这会增加路由节点的资源消耗量,增加面积,并不利于提高芯片的集成度。

在目前所研究的NOC路由节点^[6]的仲裁器设计中,几乎都是采用RR仲裁算法对各个端口所发出的请求进行响应。该算法平均地轮询响应各个端口的请求,是一种无优先级的均匀响应仲裁方式。在这种方式下,无法区分数据的优先级别和程度,对所有的数据都平等对待,仲裁器均匀地响应各个端口的请求。由于具体的应用中IP实现功能不同,各个IP之间具有不同的通信量和权重。因此,需要针对

收稿日期: 2007-11-13; 修回日期: 2008-03-19

基金项目: 国家自然科学基金(60575031)

作者简介: 武 畅(1979-), 男, 博士生, 主要从事片上网络技术、实时信号处理和DSP技术方面的研究。

不同的通信状况实现不同优先程度的响应。但是，如果采用固定优先级的静态响应机制，对NOC的网络结构来说并不合适。因为在网络注入率较高，网络数据交换比较繁忙的时候，就会由于静态优先级的机制造成低优先级的端口很难得到响应，有时候甚至得不到应答而发生低优先级设备的饥饿问题。

因此，本文提出了在NOC路由节点中采用基于Lottery算法的仲裁机制^[7]，每个端口根据其通信量和通信权重获得一定数量的“彩票”，优先程度高的获得彩票数量大；反之，获得彩票数量小。当端口发生竞争的时候，仲裁器就根据端口提交的彩票随机抽取一张，确定响应的对象。这就保证了各个端口均有获得响应的概率，避免了“饥饿”现象。另外，为了提高数据传输效率，缓解由于竞争引起的

内部阻塞，本文在所设计的路由节点中采用了TUM方式。这在后面路由节点结构的设计中会详细介绍。

1 路由节点结构

NOC的路由节点主要分为输入部分和输出部分，共有东、南、西、北、本地5个方向。每个方向都与其他4个方向连接。本文使用的是VOQ和单输出缓冲共存的形式。在目前NOC路由节点的仲裁器中，都采用基于RR的仲裁机制。该方法实现简单，对所有的通信平等对待，但在面临具体应用的时候会缺乏灵活性，且不能对不同的通信设置优先程度。本文通过使用基于Lottery算法的仲裁设备和TUM机制实现了可设置优先程度的高效NOC路由节点。该节点一组输入端口和输出端口的结构框较长如图1所示。

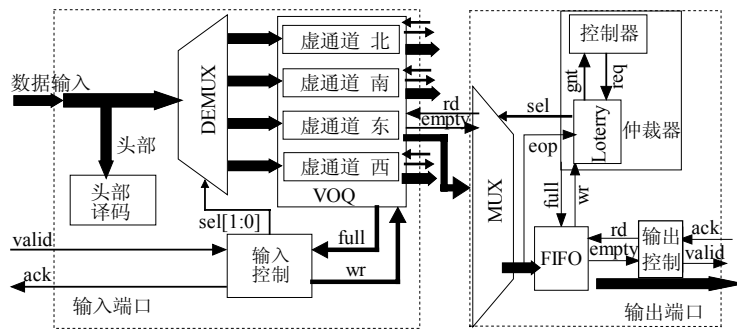


图1 NOC路由节点微结构

1.1 Lottery仲裁器的实现

在NOC的应用中，仲裁器主要实现的是对不同请求的响应。本文采用基于Lottery算法的仲裁器取代普遍采用的RR仲裁器，主要原因有：(1) 具体应用的NOC通信中，通信需求很不平衡，需要仲裁器具有区分优先级的能力。(2) Lottery仲裁器只需通过设置优先程度来降低热点影响，不需消耗额外的缓存资源。(3) 能避免固定优先级机制中，低优先级设备“饥饿”的问题。(4) 能根据NOC的通信状况改变，方便地实现各通信链路的优先程度的可重配置或动态配置^[8]。图2表示了4个输入端口申请一个输出端口的Lottery仲裁器的结构。

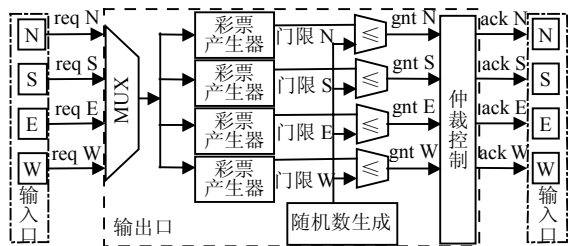


图2 彩票仲裁器结构

在Lottery方法中，每个请求根据其优先级别的

不同获得一定数量的“彩票”，优先级高的“彩票”数量多。当有请求的时候，根据事先设置好的彩票比例通过“彩票产生器”产生一个彩票数量值，即门限。再生成一个随机数，即中奖的彩票，将其与门限进行比较，就可以确定是哪一个方向中奖，则响应该方向请求。然后将这组gnt信号输入到仲裁器中，再结合输出缓冲的情况向输入端口发出ack信号，完成数据传输。然后重复前面的过程，进行“二次开奖”。如果输出缓冲不允许数据接收，则本次中奖无效，重新产生随机数，进行“二次开奖”。

本文用 r_1, r_2, \dots, r_n 代表各端口对输出口的请求，有请求则令该端口对应的 $r_i=1$ ；反之 $r_i=0$ 。每个端口可以拥有的彩票数量根据通信情况来确定，记为 l_1, l_2, \dots, l_n ，那么，各端口获得响应的概率为：

$$P(C_i) = r_i l_i / \sum_{j=1}^n r_j l_j$$

设彩票总数为 L ，则各端口的门限为：

$$\text{threshold} = P(C_i)L + \sum_{j=1}^{i-1} P(C_j)L$$

当请求到来时，仲裁器产生一个在 $[1, L]$ 之间的

随机数, 根据该数据与各个门限的比较值来确定响应的端口。设彩票总数为100, N、S、E、W各个端口对应的优先程度分别为1:2:3:4, 当 $r_1 r_2 r_3 r_4=1011$ 的时候, 对应的S端口没有请求, 则N、E、W的门限值分别为13、50、100。如果产生的随机数为47, 则在(13,50]之间, 响应端口E的请求。依此类推。

1.2 TUM机制

在基于虫洞路由的片上网络通信系统中, 数据是以flit为单位进行传输的, 若干个flit组成一个数据包。对于一个OB, 一个周期只能接收一个flit, 若发生两个输入端口同时竞争一个输出端口的情况, 则必然会使有一个端口的数据被阻塞一个周期, 这就是内部阻塞。从而引起竞争阻塞, 严重会导致网络性能急剧下降。本文采用了TUM的方式来解决内部阻塞的问题, 这种方式与传统的时分复用方式不同。在传统的时分复用方式中, 端口占用指定的时隙, 若对应端口无数据传送, 则该时隙空闲。而TUM方式将一个周期划分成若干个时间单元, 并不固定某个时隙给某个端口使用, 只要有空闲单元任何端口均可占用该时间单元。这有点类似于ATM的方式, 但是要简单得多。图3为具体的实现方法。在具体实现中, 根据端口的数量, 将一个外部周期分成了4个内部时间单元, 即使节点内部数据交换的速度是外部的4倍, 可视为一种简单的GALS方式。但是该方法也有缺陷:

- (1) 该方法要求路由节点在输入输出端口都具有缓冲器, 单一的输入缓冲和输出缓冲的路由节点不能使用该方法。
- (2) 当输出缓冲存储满以后, 阻塞仍然会发生, 但是通过增加输出缓冲的大小能降低阻塞概率, 提高性能。

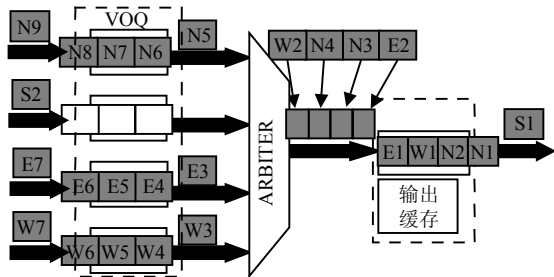


图3 TUM机制

2 针对具体应用的仿真

本文基于两种具体应用MPEG4解码器^[9]和VOPD(video object plane decoder)^[10], 对所设计的路由节点进行了性能验证和评估, 并与使用RR仲裁算

法的路由节点进行了比较。其片上网络结构的IP关系如图5所示。从图中可以很明显看到具体应用中存在的通信不均匀的问题。图中数字表示通信重量(通信量和权重的综合)。从具体的通信中可以看到, 一个路由节点的端口可能同时被几个输入端口申请, 并且各个请求端口的通信重量的大小很不均匀。如图5a中, UP SAMP节点的本地、西、东都会通过北输出端口传送数据, 但是其通信重量分别为910、0.5、670, 极不均衡, 不应该均匀对待。在整个网络中, 并不需要对每个节点的每个端口的仲裁器都进行优先程度的设置, 只需要针对通信比较集中、冲突比较严重的节点设置即可。如图5a的节点(0,0)就不需要设置, 而节点(0,1)也只需要设置其南输出端口的仲裁器对L、E、W 3个方向的响应程度。在彩票总数100的情况下, 其彩票数分别为75、24和1。其余节点的情况可以根据图4类推出来。

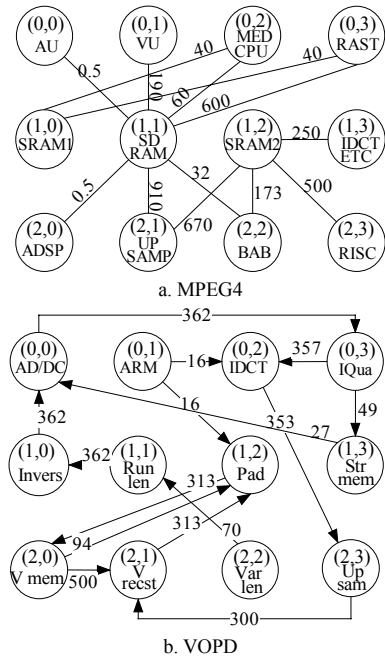


图4 特殊应用中各IP之间的通信情况

本文分别用两种不同的路由节点构建了用于MPEG4和VOPD两种具体应用的NOC。在图5和图6中, 对本文所设计的路由节点和RR路由节点的性能进行了比较。图中统计了在一段时间内(10 000 ns) NOC中各个节点收到的数据包数量以及收发包比。从图中可以看到, 在GT和BE两种不同的通信模式中, 本文的路由节点都具有较大的优势。如MPEG4的GT模式下, 本文路由节点在相同的时间长度和时间段内, 具有更大的收包量。在BE模式中, 由于网络尽最大努力在传送数据, 对传输没有了保证机制, 因此本文的路由节点性能优势能更大地表现出来,

不仅在收包的数量上,而且在收发比上也远远高于RR路由节点。这是因为,RR路由节点对所有的信道都平等对待,而本文的路由节点则根据具体情况对信道分配了优先级别,这样就能使较繁忙的信道得到较多的响应,使用到更多的资源,从而提高网络利用率。因此,使用本文的路由节点所构成的NOC网络能更好适应特定应用的NOC的需求,并有助于提高网络性能。

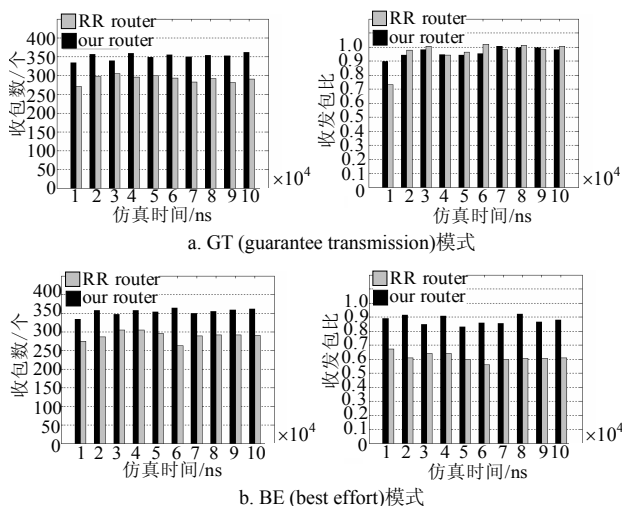


图5 MPEG4 NOC性能比较

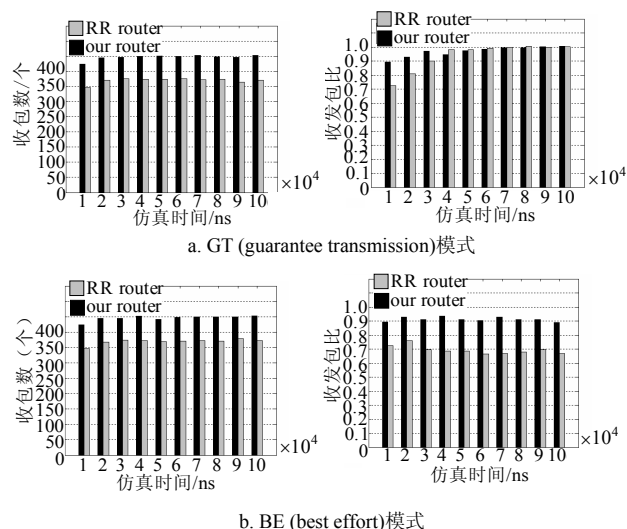


图6 VOPD NOC性能比较

3 总结

本文提出了一种基于彩票仲裁器和TUM内部交换机制的NOC路由节点。该节点针对特定应用使用了彩票仲裁算法来配置不同通信的仲裁优先程度,从而实现对不同通信的区分。另外,为了缓解

内部阻塞带来的影响,本文使用TUM机制来提高节点内部数据交换的速度。在具体应用中,本文在MEPG4和VOPD的基础上构建了基于2D Mesh拓扑结构和XY路由算法的NOC,并将特定应用的IP映射到了网络中进行仿真。与RR路由节点的比较结果表明,本文的路由节点在GT和BE模式中都具有更好的性能。由于该路由节点能实现仲裁优先程度可配置,因此在今后的研究中,将把这种路由节点使用到可重配置和动态配置的片上网络系统中去。

参考文献

- [1] HENKEL J, WOLF W, CHAKRADHAR S. On-chip networks a scalable, communication-centric embedded system design paradigm[C]//Proceedings of the 17th International Conference on VLSI Design. Piscataway, NJ, USA: IEEE Press, 2004: 845-851.
- [2] LIU Jian, ZHENG Li-Rong, TENHUNEN H. Interconnect intellectual property for Network-on-Chip[J]. Journal of Systems Architecture, 2004, 50: 65-79.
- [3] DALLY W J, TOWLES B. Route packets, not wires: on-chip interconnection networks[C]// Design Automation Conference. Las Vegas, USA: ACM Press, 2001: 684-689.
- [4] KUMAR S, JANTSCH A, SOININEN J P, et al. A Network on chip architecture and design methodology[C]//IEEE Computer Society Annual Symposium on VLSI (ISVLSI.02). Pittsburgh, Pennsylvania, USA: IEEE Press, 2002: 105-112.
- [5] SALEH R, WILTON S, MIRABBASI S, et al. System-on-Chip: reuse and integration[J]. Proceedings of the IEEE. 2006, 94(6): 1050-1069.
- [6] NARASIMHAN A, SRINIVASAN K, Sridhar R. A high-performance router design for VDSM NoCs[C]//IEEE International SOC Conference. Herndon, VA: IEEE Press, 2005: 301-304.
- [7] LAHIRI K, RAGHUNATHAN A, LAKSHMINARAYANA G. The Lotterybus on-chip communication architecture[J]. Very Large Scale Integration (VLSI) Systems IEEE Transactions, 2006, 14(6): 596-608.
- [8] BARTIC T A, MIGNOLET J Y, NOLLET V. Highly scalable network on chip for reconfigurable systems[C]// International Symposium on System-on-Chip 2003. Tampere, Finland: IEEE Press. 2003: 79-82.
- [9] BERTOZZI D, BENINI L. Xpipes: A network-on-chip architecture for gigascale systems-on-chip[J]. Circuits and Systems Magazine, IEEE, 2004, 4(2): 18-31.
- [10] NGUYEN H N, NGO V D, Choi H W. Assessing routing behavior on On-Chip Network[C]//The 2006 International Conference on Computer Engineering and Systems. Cairo, Egypt: IEEE Press, 2006: 62-65.

编辑 张俊