

(19) 中华人民共和国国家知识产权局



# (12) 发明专利申请

(10) 申请公布号 CN 103595627 A

(43) 申请公布日 2014. 02. 19

(21) 申请号 201310628342. 3

H04L 12/935 (2013. 01)

(22) 申请日 2013. 11. 28

(71) 申请人 合肥工业大学

地址 230009 安徽省合肥市包河区屯溪路  
193 号

(72) 发明人 王伟 李润丰 陈田 刘军 方芳  
吴玺

(74) 专利代理机构 安徽省合肥新安专利代理有  
限责任公司 34101

代理人 何梅生 郭华俊

(51) Int. Cl.

H04L 12/701 (2013. 01)

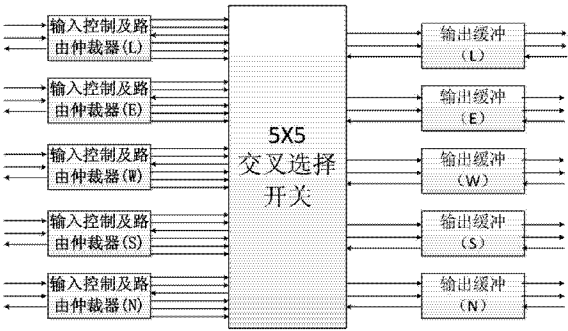
权利要求书2页 说明书8页 附图4页

## (54) 发明名称

基于多播维序路由算法的 NoC 路由器及其路由算法

## (57) 摘要

本发明公开了一种基于多播维序路由算法的 NoC 路由器及其路由算法。该路由器包含输入控制及路由仲裁器、交叉选择开关和输出缓冲及控制器；输入控制及路由仲裁器与交叉选择开关相连接，交叉选择开关与输出缓冲及控制器相连接；数据包进入输入控制及路由仲裁器后由仲裁器对该数据包进行解析，提取该数据包中包含的一个或几个目的地址，分别判断传送方向并发送给交叉选择开关；交叉选择开关复制数据包，并将它传送到相应的一个或几个输出端口，输出缓冲及控制器负责向下一个路由节点发送数据。本发明的 NoC 路由器及其路由算法，具有可解决对含有众核的 NoC 芯片进行多播测试过程中缺少路由器支持的问题、结构简单、适应性强、可实现多播功能等优点。



1. 基于多播维序路由算法的 NoC 路由器,其特征是,包括输入控制及路由仲裁器、交叉选择开关和输出缓冲及控制器;所述输入控制及路由仲裁器与所述交叉选择开关相连接,所述交叉选择开关与所述输出缓冲及控制器相连接;数据包进入输入控制及路由仲裁器后由仲裁器对该数据包进行解析,提取该数据包中包含的一个或多个目的地址,判断数据包的传送方向,并将数据包发送给交叉选择开关;交叉选择开关接受所述输入控制及路由仲裁器发来的数据包以及传送方向信号,将数据包传送至正确的输出端口,输出缓冲及控制器负责向下一个路由节点发送数据。

2. 根据权利要求1所述的基于多播维序路由算法的 NoC 路由器,其特征是,所述数据包采用数据微片格式,分为头微片、中间微片和尾微片;头微片主要用于提示新数据的到来,并记录要将数据传送到的前两个目的地址信息;中间微片主要用于存放传送的具体数据信息;尾微片用于主要用于提示数据包传送完毕。

3. 根据权利要求2所述的基于多播维序路由算法的 NoC 路由器,其特征是,所述头微片中,如果由于目的地址信息过多使得头微片存放不完目的地址信息,使用头微片后续的中间微片存放头微片尚未存放完的目的地址信息。

4. 一种基于多播维序路由算法的 NoC 路由器的路由算法,其特征是,数据传输过程如下:

步骤1:数据包的数据微片到达 NoC 路由器的一个输入控制及路由仲裁器,输入控制及路由仲裁器判断到达的数据微片是否为头微片;如果是头微片,将路径保持标志置为“真”,并判断该头微片是否包含了全部地址信息,如果不是,则等待后续包含地址信息的中间微片的到达。当所有包含地址信息的数据微片到达后,将这些数据微片内的地址信息与本地路由节点地址信息相比较,依据 X-Y 路由规则,判断数据包的发送方向,将数据包发送给交叉选择开关;

步骤2:数据微片被传送至交叉选择开关;交叉选择开关接收输入控制及路由仲裁器发送的数据包;如果步骤1中判断的数据传送方向是单一方向,则将数据微片传送至数据微片所要输送的输出缓冲及控制器;如果步骤1中判断的数据传送方向是多个方向,则根据不同的传送方向复制数据包,剔除跟该传送方向不同的目的地址,将重新打包的数据包传送至相应的输出缓冲及控制器;

步骤3:数据被传送到了某一个或某几个输出缓冲及控制器模块,此时 Req\_que=1,把数据写入缓存(Buffer\_write=1),向交叉开关发送应答信号,并向下游节点发送请求数据输出信号。此时给上个路由节点发送应答信号 Ack\_que=1,表示数据已经成功转发。

5. 根据权利要求3所述的基于多播维序路由算法的 NoC 路由器的路由算法,其特征是,所述步骤1中,判断数据包的发送方向的方法如下:

比较地址微片的第3到第6位与本地路由节点地址的前4位,如果前者大,表示目的节点在当前节点的右侧,置输入控制及路由仲裁器的 Select1 引脚为“001”;如果后者大,表示目的节点在当前节点的左侧,置输入控制及路由仲裁器的 Select1 引脚为“010”;

当地址微片的第3到第6位与本地路由节点地址的前4位同样大时,比较地址微片的第7到第10位与本地路由节点地址的后4位;若前者大,表示目的节点在当前节点的下侧,也就是南边,置输入控制及路由仲裁器的 Select1 引脚为“011”;如果后者大,表示目的节点在当前节点的上侧,置输入控制及路由仲裁器的 Select1 引脚为“100”;

若地址微片的第 7 到第 10 位与本地路由节点地址的后 4 位一样大,则表示当前节点即为目的节点,置输入控制及路由仲裁器的 Select1 引脚为“000”。

6. 根据权利要求 3 所述的基于多播维序路由算法的 NoC 路由器的路由算法,其特征是,所述步骤 2 中,数据包被传送至交叉选择开关的过程为:当其中一个输入控制及路由仲裁器的端口发来的请求信号时,若有虚电路保持信号,则首先判断发送虚电路保持信号的方向是否和请求信号的方向一致,即同一端口发送,若不是,则表示此路由已被其他端口占用,不能传输所请求的信号;然后检测该输入控制及路由仲裁器的端口方向有没有虚电路保持信号,如果没有,就将输入控制及路由仲裁器发来的路由选择信号存储在本地存储器中,将传送来的数据信息保存在本地缓存中,并向发来数据的输入控制及路由仲裁器发送应答信号。

## 基于多播维序路由算法的 NoC 路由器及其路由算法

### 技术领域

[0001] 本发明涉及一种基于多播维序路由算法的 NoC 路由器及其路由算法。

### 背景技术

[0002] 随着集成电路制造技术水平地不断提高以及 NoC (Network on Chip)设计技术和 3D 集成电路技术的不断发展,芯片向集成众多由第三方提供的 IP 核,形成具有众核的大规模电路的方向发展。如何对这些芯片中的众多 IP 核进行快速有效的测试,是个非常棘手的问题,包括如何处理测试时间过长与测试功耗过大的问题,而合理使用并行测试的方法则能有效的解决这些问题。

[0003] 将网络通信技术运用在 NoC 芯片测试中传输数据是目前芯片测试的主要手段。这些通信技术主要包含单播通信和多播通信两种:

[0004] 1、单播通信:是一种单一的源头发送到单一的目的接收者的一种网络服务。网络中绝大多数的通信都是以单播形式传输的。

[0005] 2、多播通信:又称组播通信,是同时向一组选定的目的地传输数据的一种网络服务。它在发送者和每一接收者之间实现一点对多点网络连接。如果一台发送者同时给多个的接收者传输相同的数据,也只需复制一份的相同数据包。它提高了数据传送效率。减少了骨干网络出现拥塞的可能性。

[0006] 单播通信是 NoC 的基本数据传送方式,而多播通信作为一种广泛运用的通信技术是多核并行处理的需求。多播通信即同时向一组选定的目的地传输数据的一种网络服务,1988 年由 Steve Deering 首次在其博士论文中提出 IP 多播的概念。但是将多播技术用于芯片测试的研究很少。

[0007] 多播测试即在 NoC 芯片上运用多播通信技术,对相同的 IP 核同时发送测试数据,这样只需发送一次测试数据,所有的同构 IP 核都可以接收到所需的测试数据。它旨在通过多播技术实现片上同构核之间测试数据(包括测试激励和参考响应)的重用来达到快速测试的目的。这样只要控制好每个阶段向哪些核发送数据,就可以很好的降低测试时间,并且减少乃至避免热斑的出现,做到测试迅速、准确。

[0008] 由于片上网络面积、功耗的受限,片上路由的结构和功能往往比较简单,一般由输入端口、路由仲裁器、交叉选择开关、输出端口等部分组成。大多数的片上路由器采用最简单的 X-Y 路由策略,向单一方向传送数据,有的还具有虚通道,可以有效的避免拥堵和死锁的发生。

[0009] 多播路由器即支持多播通信技术的路由器,它在总体结构上与普通片上路由器类似,不同的是它可以对一个数据包进行分析,找出数据包要到达的一个或多个目的地址,根据情况向一个或多个方向同时传输相同的数据包。对于需要传送大量相同测试数据的众核芯片来说,可以有效的减少测试时间,降低测试功耗。

[0010] 在国内外对于 3D 芯片测试的研究中,对于多播通信的研究还是比较多的,但是对于多播测试所要用到的多播路由器设计的研究不是很多。这些多播通信研究也给多播技术

应用于测试领域提供了技术支持。这些研究主要集中在如下几点：

[0011] 1、通过使用多播测试技术，将分布集中的同构芯核集合细分成测试时间错开的多个子集来消除“热斑”，并通过构建不引起“热斑”的多类同构芯核多播测试路径并行方案来降低测试时间，以达到测试并行与热量约束间的平衡。但是没有研究具体的用于多播测试的路由器结构。

[0012] 2、使用一种面向连接的路径多播机制来用于 NoC 测试。该方法提出的报文协议简单，实用性强，但是只是研究了具体的路由算法，没有说明路由器的设计细节。

[0013] 3、在 NoC 路由器中使用虚电路树状多播的结构，用一种特殊的单播报文（虚电路树建立报文）建立虚电路表，路由器依据此虚电路表转发相继通过的多播报文，但是硬件开销大。

[0014] 4、在 NoC 路由器中使用多播并行流水线技术，很好的实现了多播路由器的多播功能，但是不能解决死锁问题，路由器的开销相对较大。

## 发明内容

[0015] 本发明是为避免上述已有技术中存在的不足之处，提供了一种基于多播维序路由算法的 NoC 路由器及其路由算法，以解决对含有众核的 NoC 芯片进行多播测试过程中缺少路由器支持的问题。

[0016] 本发明为解决技术问题，提供了一种基于多播维序路由算法的 NoC 路由器。

[0017] 基于多播维序路由算法的 NoC 路由器，其结构特点是，包括输入控制及路由仲裁器、交叉选择开关和输出缓冲及控制器；所述输入控制及路由仲裁器与所述交叉选择开关相连接，所述交叉选择开关与所述输出缓冲及控制器相连接；数据包进入输入控制及路由仲裁器后由仲裁器对该数据包进行解析，提取该数据包中包含的一个或多个目的地址，判断数据包的传送方向，并将数据包发送给交叉选择开关；交叉选择开关接受所述输入控制及路由仲裁器发来的数据包以及传送方向信号，将数据包传送至正确的输出端口，输出缓冲及控制器负责向下一个路由节点发送数据。

[0018] 本发明的基于多播维序路由算法的 NoC 路由器还具有以下技术特点。

[0019] 所述数据包采用数据微片格式，分为头微片、中间微片和尾微片；头微片主要用于提示新数据的到来，并记录要将数据传送到的前两个目的地址信息；中间微片主要用于存放传送的具体数据信息；尾微片用于主要用于提示数据包传送完毕。

[0020] 所述头微片中，如果由于目的地址信息过多使得头微片存放不完目的地址信息，使用头微片后续的中间微片存放头微片尚未存放完的目的地址信息。

[0021] 本发明还提供了一种基于多播维序路由算法的 NoC 路由器的路由算法。

[0022] 一种基于多播维序路由算法的 NoC 路由器的路由算法，其数据传输过程如下：

[0023] 步骤 1：数据包的数据微片到达 NoC 路由器的一个输入控制及路由仲裁器，输入控制及路由仲裁器判断到达的数据微片是否为头微片；如果是头微片，将该头微片内的数据与本地路由节点地址数据相比较，判断数据包的发送方向，将数据包发送给交叉选择开关；

[0024] 步骤 2：数据包被传送至交叉选择开关；交叉选择开关接收输入控制及路由仲裁器发送的数据包；根据数据包的发送方向，交叉选择开关将数据包传送至数据包所要输送

的输出缓冲及控制器；

[0025] 步骤3:数据被传送到了某一个输出缓冲及控制器模块,此时 Req\_que=1,把数据写入缓存(Buffer\_write=1),向交叉开关发送应答信号,并向下游节点发送请求数据输出信号。此时给上个路由节点发送应答信号 Ack\_que=1,表示数据已经成功转发。

[0026] 所述步骤1中,判断数据包的发送方向的方法如下:

[0027] 比较头微片的第3到第6位与本地路由节点地址的前4位,如果前者大,表示目的节点在当前节点的右侧(也就是图1中的东边E),置输入控制及路由仲裁器的 Select1 引脚为“001”;如果后者大,表示目的节点在当前节点的左侧(也就是图1中的西边W),置输入控制及路由仲裁器的 Select1 引脚为“010”;

[0028] 当头微片的第3到第6位与本地路由节点地址的前4位同样大时,比较头微片的第7到第10位与本地路由节点地址的后4位;若前者大,表示目的节点在当前节点的下侧,也就是南边(也就是图1中的南边S),置输入控制及路由仲裁器的 Select1 引脚为“011”;如果后者大,表示目的节点在当前节点的上侧(也就是图1中的北边N),置输入控制及路由仲裁器的 Select1 引脚为“100”;

[0029] 若头微片的第7到第10位与本地路由节点地址的后4位一样大,则表示当前节点即为目的节点(也就是图1中的本地节点L),置输入控制及路由仲裁器的 Select1 引脚为“000”。

[0030] 所述步骤2中,数据包被传送至交叉选择开关的过程为:当其中一个输入控制及路由仲裁器的端口发来的请求信号时,若有虚电路保持信号,则首先判断发送虚电路保持信号的方向是否和请求信号的方向一致,即同一端口发送,若不是,则表示此路由已被其他端口占用,不能传输所请求的信号;然后检测该输入控制及路由仲裁器的端口方向有没有虚电路保持信号,如果没有,就将输入控制及路由仲裁器发来的路由选择信号存储在本地存储器中,将传送来的数据信息保存在本地缓存中,并向发来数据的输入控制及路由仲裁器发送应答信号。

[0031] 与已有技术相比,本发明有益效果体现在:

[0032] 本发明提出了一种基于多播维序路由算法的 NoC 路由器及其路由算法, NoC 路由器由输入控制及路由仲裁器、交叉选择开关、输出缓冲及控制器三大硬件部分组成。使用特殊的多播数据微片格式,运用特定的多播维序路由算法,采用虫孔交换技术,通过建立虚电路的方式进行数据的传送。采用的多播数据微片分为头微片、中间微片和尾微片三种;采用的路由算法是一种基于 X-Y 路由算法发展而来的可处理多个目的地址信息的路由算法。本发明的具有多播功能 NoC 路由器具有结构简单,功能完备,适应性强,可实现多播功能等优点。

[0033] 本发明的 NoC 路由器及其路由算法,具有可解决对含有众核的 NoC 芯片进行多播测试过程中缺少路由器支持的问题、结构简单、适应性强、可实现多播功能等优点。

#### 附图说明

[0034] 图1为本发明的 NoC 路由器的总体结构图。

[0035] 图2为本发明的 NoC 路由器的输入控制及路由仲裁器引脚示意图。

[0036] 图3为本发明的 NoC 路由器的交叉选择开关引脚示意图。

- [0037] 图 4 为本发明的 NoC 路由器的输出缓冲及控制器引脚示意图。
- [0038] 图 5 为本发明的 NoC 路由器的路由算法的特殊数据微片格式示意图。
- [0039] 图 6 为本发明的 NoC 路由器的多播维序路由算法的流程图。
- [0040] 以下通过具体实施方式,对本发明作进一步说明。

### 具体实施方式

[0041] 参加图 1 ~ 图 6, 基于多播维序路由算法的 NoC 路由器, 包括输入控制及路由仲裁器、交叉选择开关和输出缓冲及控制器; 所述输入控制及路由仲裁器与所述交叉选择开关相连接, 所述交叉选择开关与所述输出缓冲及控制器相连接; 数据包进入输入控制及路由仲裁器后由仲裁器对该数据包进行解析, 提取该数据包中包含的一个或多个目的地址, 判断数据包的传送方向, 并将数据包发送给交叉选择开关; 交叉选择开关接受所述输入控制及路由仲裁器发来的数据包以及传送方向信号, 将数据包传送至正确的输出端口, 输出缓冲及控制器负责向下一个路由节点发送数据。

[0042] 所述数据包采用数据微片格式, 分为头微片、中间微片和尾微片; 头微片主要用于提示新数据的到来, 并记录要将数据传送到的前两个目的地址信息; 中间微片主要用于存放传送的具体数据信息; 尾微片用于提示数据包传送完毕。

[0043] 所述头微片中, 如果由于目的地址信息过多使得头微片存放不完目的地址信息, 使用头微片后续的中间微片存放头微片尚未存放完的目的地址信息。

[0044] 一种基于多播维序路由算法的 NoC 路由器的路由算法, 其数据传输过程如下:

[0045] 步骤 1: 数据包的数据微片到达 NoC 路由器的一个输入控制及路由仲裁器, 输入控制及路由仲裁器判断到达的数据微片是否为头微片; 如果是头微片, 将路径保持标志置为“真”, 并判断该头微片是否包含了全部地址信息, 如果不是, 则等待后续包含地址信息的中间微片的到达; 当所有包含地址信息的数据微片到达后, 将这些微片内的地址信息与本地路由节点地址信息相比较, 依据 X-Y 路由规则, 判断数据包的发送方向, 将数据包发送给交叉选择开关;

[0046] 步骤 2: 数据微片被传送至交叉选择开关; 交叉选择开关接收输入控制及路由仲裁器发送的数据包; 如果步骤 1 中判断的数据传送方向是单一方向, 则将数据微片传送至数据微片所要输送的输出缓冲及控制器; 如果步骤 1 中判断的数据传送方向是多个方向, 则根据不同的传送方向复制数据包, 剔除跟该传送方向不同的目的地址, 将重新打包的数据包传送至相应的输出缓冲及控制器;

[0047] 步骤 3: 数据被传送到了某一个或某几个输出缓冲及控制器模块, 此时 Req\_que=1, 把数据写入缓存(Buffer\_write=1), 向交叉开关发送应答信号, 并向下游节点发送请求数据输出信号。此时给上个路由节点发送应答信号 Ack\_que=1, 表示数据已经成功转发。

[0048] 所述步骤 1 中, 判断数据包的发送方向的方法如下:

[0049] 比较地址微片的第 3 到第 6 位与本地路由节点地址的前 4 位, 如果前者大, 表示目的节点在当前节点的右侧(也就是图 1 中的东边 E), 置输入控制及路由仲裁器的 Select1 引脚为“001”; 如果后者大, 表示目的节点在当前节点的左侧(也就是图 1 中的西边 W), 置输入控制及路由仲裁器的 Select1 引脚为“010”;

[0050] 当地址微片的第 3 到第 6 位与本地路由节点地址的前 4 位同样大时,比较地址微片的第 7 到第 10 位与本地路由节点地址的后 4 位;若前者大,表示目的节点在当前节点的下侧,也就是南边(也就是图 1 中的南边 S),置输入控制及路由仲裁器的 Select1 引脚为“011”;如果后者大,表示目的节点在当前节点的上侧(也就是图 1 中的北边 N),置输入控制及路由仲裁器的 Select1 引脚为“100”;

[0051] 若地址微片的第 7 到第 10 位与本地路由节点地址的后 4 位一样大,则表示当前节点即为目的节点(也就是图 1 中的本地节点 L),置输入控制及路由仲裁器的 Select1 引脚为“000”。(查看头微片的后 8 位是否为全 1,如果为全 1,则表示该数据只向一个目的节点发送,置输入控制及路由仲裁器的 Select2 引脚为“111”。否则按照上述规则确定 Select2 的取值。查看 Select1 与 Select2 的取值,若两者相等,则表示向同一个方向发送数据,为了避免重复发送数据,置 Select2 为“111”,即无效。)

[0052] 所述步骤 2 中,数据包被传送至交叉选择开关的过程为:当其中一个输入控制及路由仲裁器的端口发来的请求信号时,若有虚电路保持信号,则首先判断发送虚电路保持信号的方向是否和请求信号的方向一致,即同一端口发送,若不是,则表示此路由已被其他端口占用,不能传输所请求的信号;然后检测该输入控制及路由仲裁器的端口方向有没有虚电路保持信号,如果没有,就将输入控制及路由仲裁器发来的路由选择信号存储在本地存储器中,将传送来的数据信息保存在本地缓存中,并向发来数据的输入控制及路由仲裁器发送应答信号。

[0053] 然后分析当前的头微片在此路由节点是否要分成多个方向传送,如果是,将他们各自的地址放在头微片的第 3 到第 10 位,后 8 位置“1”,根据路由选择信息跳转到不同的输出端口,并向输出缓冲器发送数据,接着判断是否还需要将数据发送到其他端口,如果需要,继续发送。

[0054] 本发明的基于多播维序路由算法的 NoC 路由器,可以由一个源地址同时向多个不同的目的地址发送消息,包括输入控制及路由仲裁器、交叉选择开关和输出缓冲及控制器三大部分硬件部分,使用可包含多个目的地址的特殊数据微片格式,运用多播维序路由算法进行数据包的转发,采用虫孔交换技术,通过建立虚电路的方式进行数据的传送。所述特定的多播维序路由算法是在普通维序路由算法的基础上,增加多地址判断功能和往多个输出端口同时发送功能。

[0055] 如图 5 所示,所述特殊的数据微片格式,分为头微片、中间微片和尾微片三种。头微片主要用于提示新数据的到来,并记录要将数据传送到的前两个目的地址信息;中间微片主要用于存放传送的具体数据信息,如果目的地址信息过多,头微片存放不完,也可使用头微片后续的中间微片存放头微片尚未存放完的目的地址信息;尾微片用于主要用于提示数据包传送完毕。

[0056] 所述多播维序路由算法是一种基于 X-Y 维序路由的多播路由算法。首先,路由器接收到传输来的头微片后,分析头微片,解析出其中的目的地址信息,并判断后续的中间微片是否包含剩余的目的地址信息。然后,分析这些地址信息,判断数据包的传送方向。对于提取到的若干个目的地址,依据特定的转发策略,判断数据包要发送的方向。后续的微片都按照前一个微片的传送方向依次传送到下一个路由端口,直到尾微片传送完毕。

[0057] 所述多播维序路由算法的特定转发规则,是根据 X 优先路由算法改变而来。对于



提取到的若干目的地址,分析这些目的地址相对于当前节点来说的方位,如果需要转发数据包的方向相同,则按普通片上路由那样,将数据微片转发到正确的方向,也就是向一个方向传送;如若不同,需要拆分数据微片地址信息,将其中同一方向的目的地址和后续数据打包成一个数据包,如此打包成若干个数据包后,再像普通片上路由那样,分别将这若干个数据包从若干个不同方向的端口输出。

[0058] 如图 1,所述输入控制及路由仲裁器分为东(East)、西(West)、南(South)、北(North)、本地(Local)五个,分别用来连接东西南北四个方向的临近路由器和本地 IP 核;所述输出缓冲及控制器也分为为东(East)、西(West)、南(South)、北(North)、本地(Local)五个;所述输入控制及路由仲裁器和输出缓冲及控制器分别与交叉选择开关相连。

[0059] 所述输入控制及路由仲裁器包含与上游节点连接的两个输入引脚和一条输出引脚,与交叉选择开关相连的五条输出引脚和一条输入引脚,以及与外界相连的时钟信号引脚(c1k)、复位信号引脚(rst)和地址注入引脚(ads);所述与上游节点连接的引脚包括数据包输入引脚(Data\_in),请求信号输入引脚(Req\_in),应答信号输出引脚(Ack\_out);所述与交叉选择开关相连的引脚包括数据输出引脚(Data\_sel),请求输出引脚(Req\_sel),应答输入引脚(Ack\_sel),路由选择信号引脚 1(Select1),路由选择信号引脚 2(Select2),路径保持信号(Keep\_sel);

[0060] 所述交叉选择开关包含与五个输入控制及路由仲裁器相连的各五条输入引脚和一条输出引脚,与五个输出缓冲及控制器相连的各两条输出引脚和一条输入引脚,以及与外界相连的时钟信号引脚(c1k)和复位信号引脚(rst);所述与其中一个输入控制及路由仲裁器连接的引脚包括数据输入引脚(Data\_sel),请求输入引脚(Req\_sel),应答输出引脚(Ack\_sel),路由选择信号引脚 1(Select1),路由选择信号引脚 2(Select2),路径保持信号(Keep\_sel);所述与输出缓冲及控制器连接的引脚包括数据输出引脚(Data\_que),请求输出引脚(Req\_que),应答输入引脚(Ack\_que);

[0061] 所述输出缓冲及控制器包含两个与交叉选择开关相连的输入引脚和一条输出引脚,与下游节点连接的两条输出引脚和一条输入引脚,以及与外界相连的时钟信号引脚(c1k)和复位信号引脚(rst);所述与交叉选择开关连接的引脚包括数据输入引脚(Data\_que),请求输入引脚(Req\_que),应答输出引脚(Ack\_que);所述与下游节点连接的引脚包括数据输出引脚(Data\_out),请求输出引脚(Req\_out),应答输入引脚(Ack\_in);

[0062] 如图 6,所述的多播维序路由算法是由普通的 X-Y 维序路由算法发展而来,当路由器接收到新的数据微片到达当前路由节点时,首先判断该微片是否为头微片,如果是头微片,分析头微片,解析出其中的目的地址信息,并判断后续的中间微片是否包含剩余的地址信息,如果还有,则保存接收到的地址信息,并等待后续微片的到达。如果后续没有地址信息了,就分析这些地址信息,判断数据包的传送方向。对于提取到的若干个目的地址,判断各个目的地址相对于当前节点的方向信息,确定数据包要发送的输出端口,并重新将相同方向的目的地址信息组合成新的头微片,分别发送到对应的目的端口,并保持相应的链路。下一个微片到达时,搜索各个链路的链路保持标志,复制该微片,分别发送到每个链路保持标志位真的方向。后续的微片都按照前一个微片的传送方向依次传送到下一个路由端口,直到尾微片传送完毕,置所有链路保持标志为假。

[0063] 数据包进入输入控制及路由仲裁器后由仲裁器对其进行解析,提取其中包含的一

个或多个目的地址,判断传送方向;然后交叉选择开关接受仲裁器发来的数据包以及传送方向信号,将数据包传送至正确的输出端口,输出缓冲及控制器负责向下一个路由节点发送数据。本发明涉及到五个方向的数据输入,以及五个方向的数据输出,分别是东、西、南、北、本地。图 1 所示即为本发明的总体结构框架及主要引脚图。在这张图中可以看到数据的输入端有五个方向,通过 5\*5 的交叉选择开关将输入的数据传送到不同的方向。

[0064] 如图 2 为输入控制及路由仲裁器:这个部分主要有 12 条信号线,其中“clk”是时钟信号,“rst”是复位信号。“ads”是发送数据前注入的当前路由器的地址信息,它是一个 8 位的信号线,用当前路由在网络中的 X、Y 坐标来表示地址,“ads”的前四位表示 X 方向坐标,后四位表示 Y 方向坐标。在注入“ads”信号后,当前的路由地址将不会再改变。“Data\_in”是上个路由发送来的数据信息,“Req\_in”是上个路由发来的请求发送数据信号,“Ack\_out”是当前路由器给上个路由器发送的应答信号,表示数据已经收到。“Data\_sel”是输入控制器向交叉选择开关发送的数据信息,“Req\_sel”是输入控制器向交叉选择开关发送的请求信号,“Ack\_sel”则是交叉开关回复的应答信号。“Select1”与“Select2”是路由仲裁器发给交叉开关的两个方向选择信号,此信号共 3 位,其中“000”代表发向本地(Local) IP 核,“001”代表发向东边(East),“010”代表发向西边(West),“011”代表发向南边(South),“100”代表发向北边(North)。“Keep\_sel”是地址保持信号,当头微片到达后就置“Keep\_sel”为 1,直至尾微片到达,起到的是保持虚电路的功能。

[0065] 如图 3 为交叉选择开关:这个部分主要负责将输入控制及路由仲裁器输送来的数据根据其路由信息转发到不同的端口。由于有五个输入端口和五个输出端口,这里介绍其中一个。“clk”为时钟信号,“rst”为复位信号。“Data\_sel”是输入控制及路由仲裁器发来的数据信号;“Req\_sel”为请求信号;“Select1”与“Select2”为路由选择信号,此模块就是根据这两个信号来进行输出端口的选择的;“Keep\_sel”为虚电路保持信号,如果此信号为 1,则不检测新的路由选择信号,还是按照之前的路径传送数据;“Ack\_sel”为应答信号;“Data\_que”为交叉开关发送给输出缓冲器的数据信息;“Req\_que”为请求信号;“Ack\_que”为应答信号。其它的四个输入端口和四输出端口与此图中的信号线及引脚大体一致。本模块的主要工作是将数据传送到正确的输出端口,它是通过接收路由仲裁器发来的路由选择信号(Select1 和 Select2)来控制数据输出方向的。

[0066] 如图 4 为输出缓冲及控制器:这个模块是用于控制数据输出、缓存数据的重要模块。此模块结构比较简单,“clk”为时钟信号,“rst”为复位信号,“Data\_que”为交叉开关发送来的数据信息,“Req\_que”为交叉开关发送来的请求信号,“Ack\_que”为应答信号,“Data\_out”为输出控制器向下个路由发送的数据信息,“Req\_out”为向下个路由节点发送的请求信号,“Ack\_in”为下个路由发节点来的应答信号。

[0067] 本发明的基于多播维序路由算法的 NoC 路由器,具有多播功能,可以同时向一个或多个目的节点发送相同的消息,为对含有众核的 NoC 芯片进行多播测试提供了硬件保障。数据包进入输入控制及路由仲裁器后由仲裁器对其进行解析,提取其中的目的地址,判断传送方向;然后交叉选择开关接受仲裁器发来的数据包以及传送方向信号,将数据包传送至正确的输出端口,输出缓冲及控制器负责向下个路由发送数据。本发明通过使用特殊的数据微片格式,运用特定的多播维序路由算法,并对一般的片上路由器中仲裁模块和交叉选择开关模块的改进,使仲裁器可以识别多个地址并判断不同的传输方向,然后由交叉

选择开关将数据包发送到不同的输出端口。

[0068] 本发明的多播路由器结构具有结构简单,功能完备,适应性强,可实现多播功能等优点。

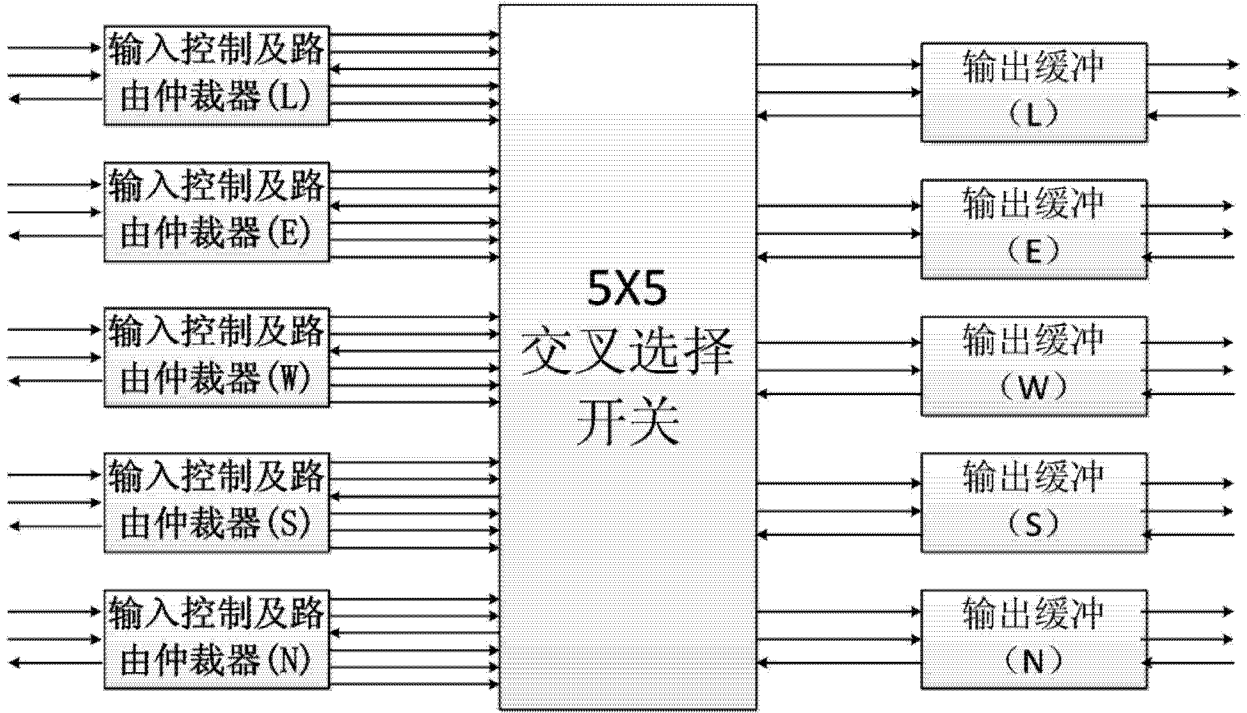


图 1

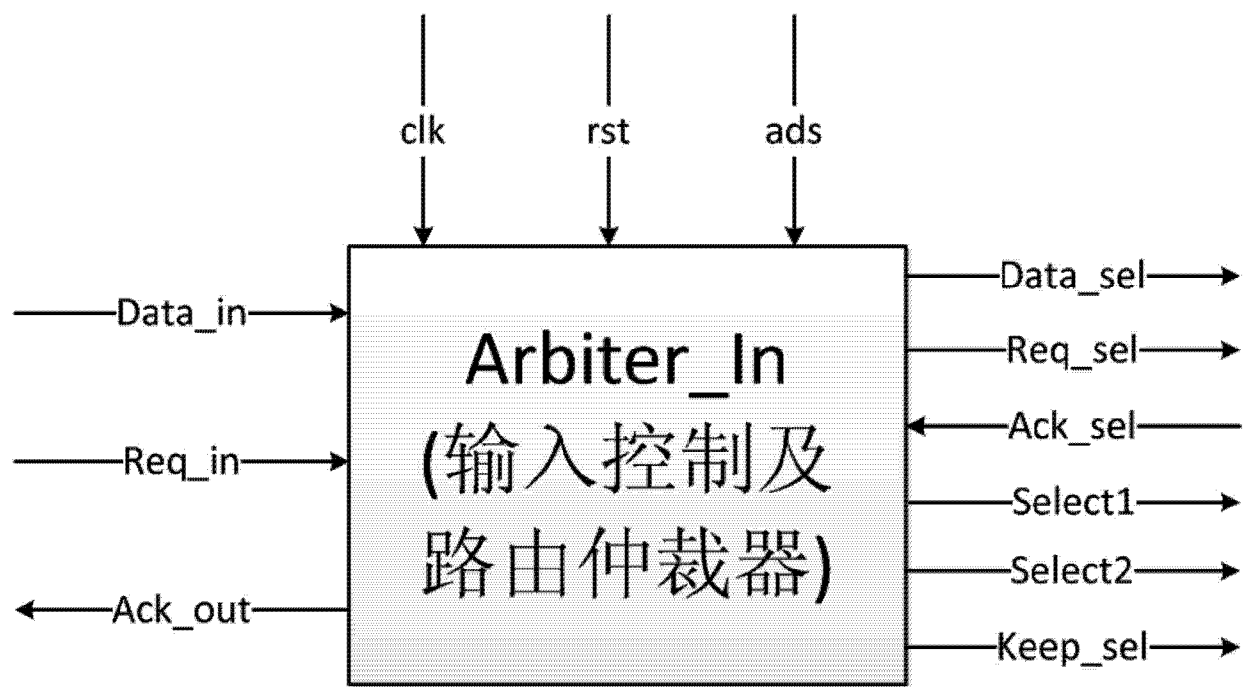


图 2

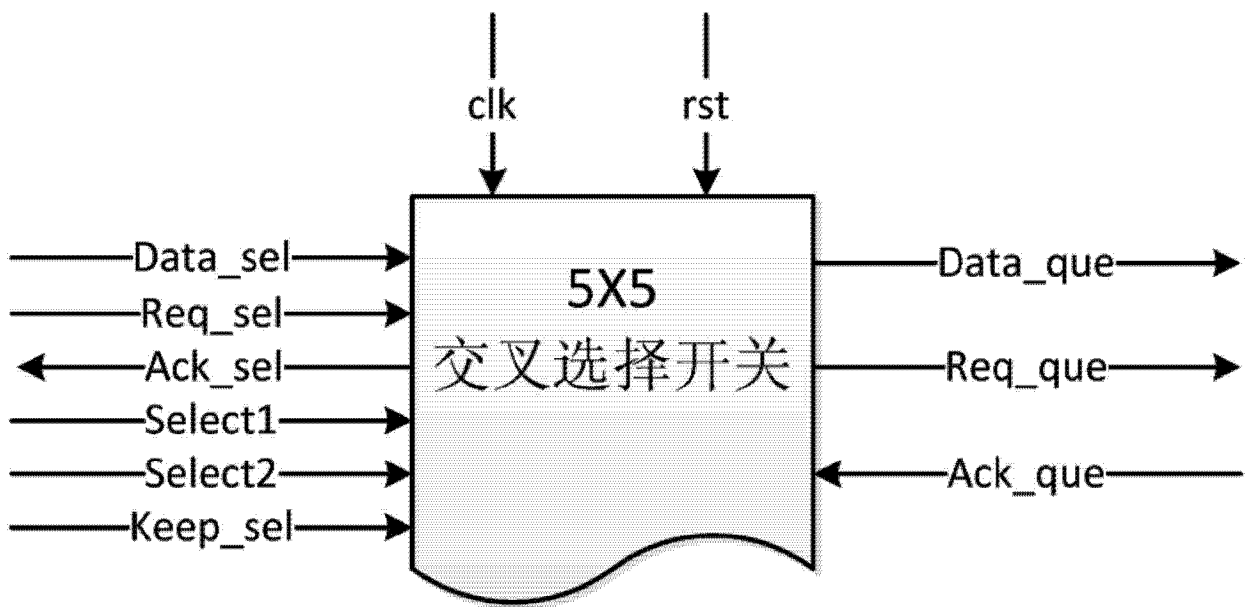


图 3

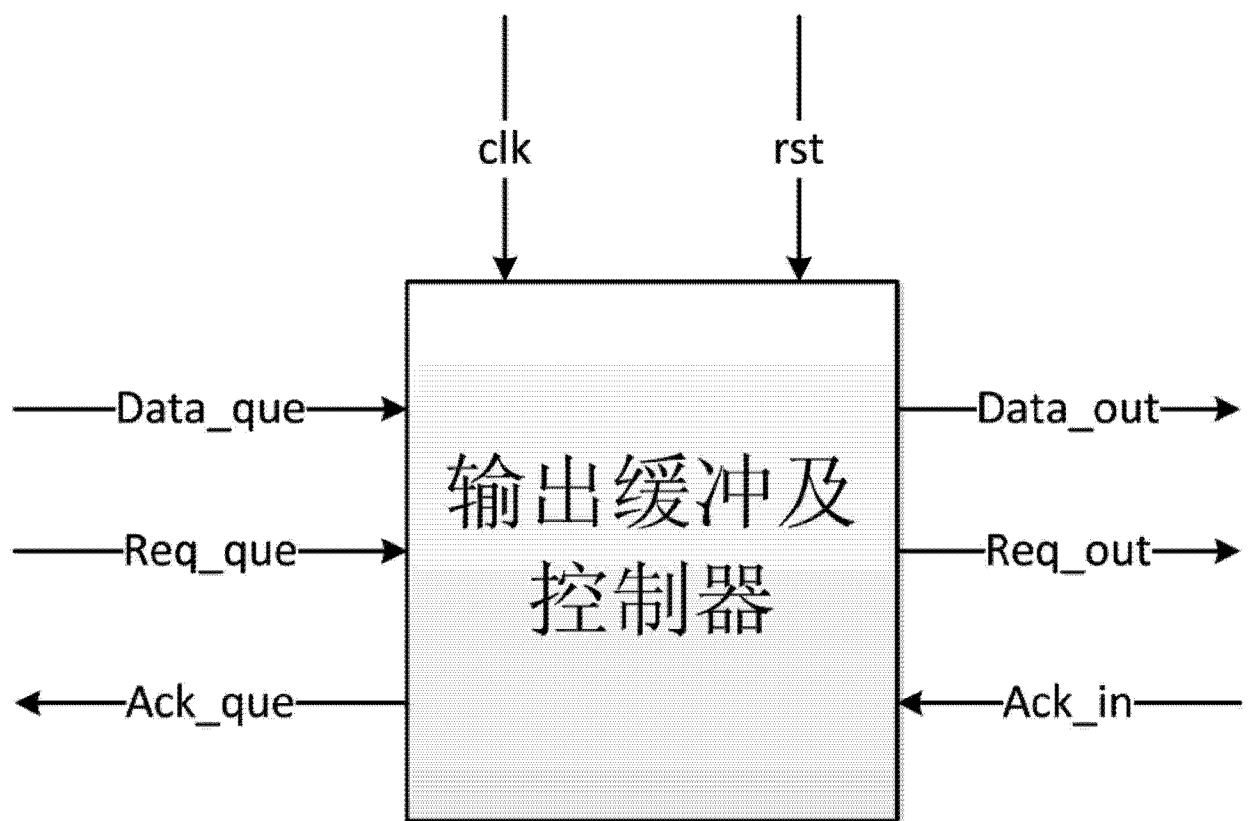


图 4

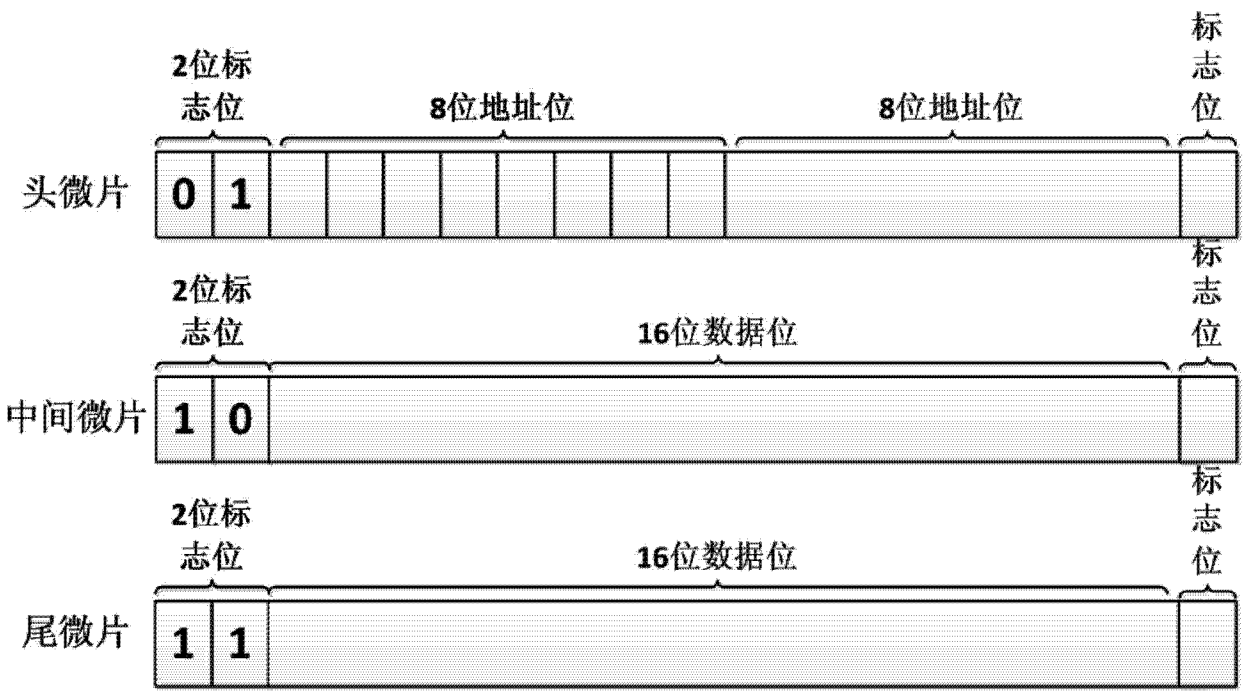


图 5

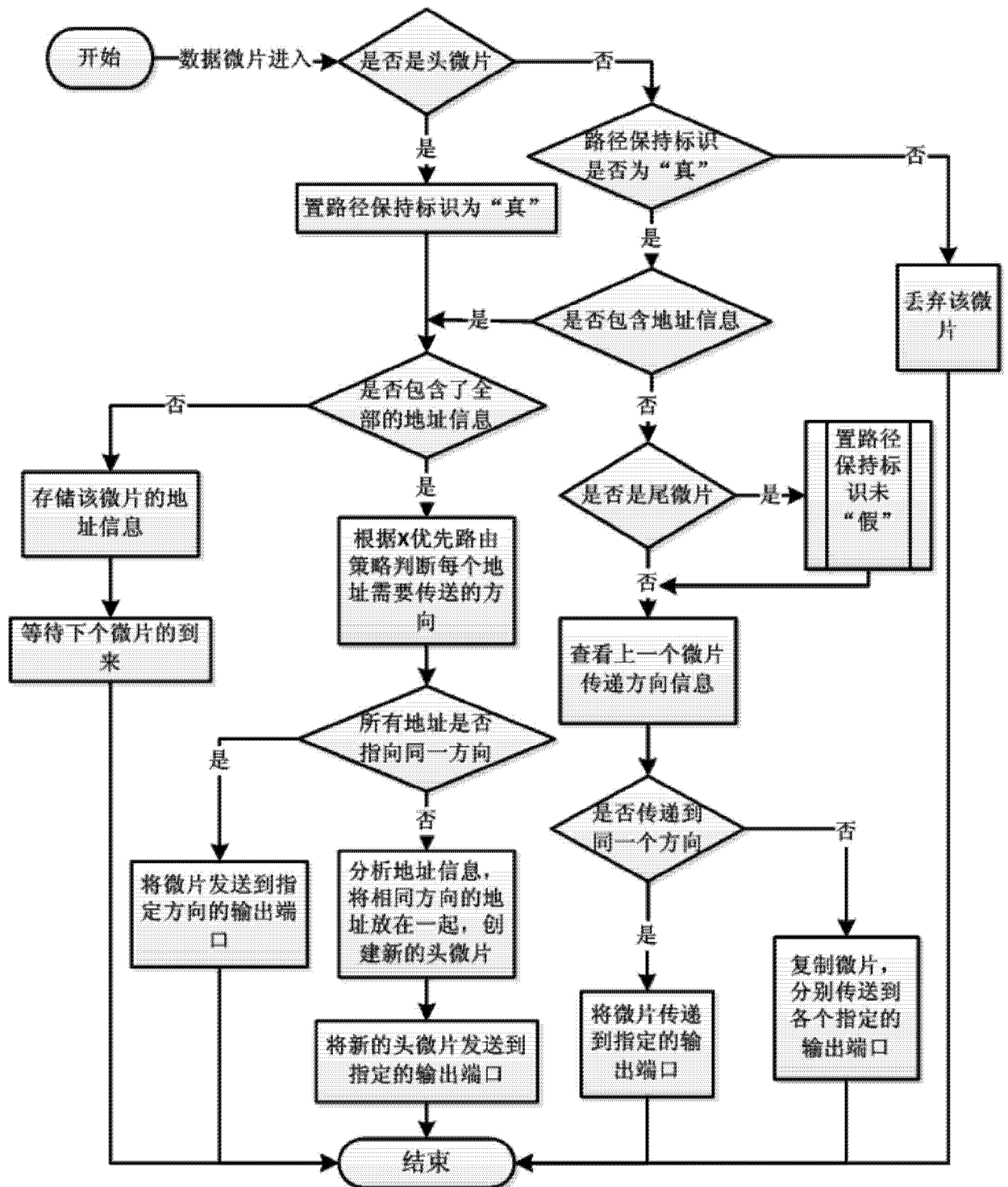


图 6