



(12) 发明专利

(10) 授权公告号 CN 108353036 B

(45) 授权公告日 2022.04.29

(21) 申请号 201580084428.1

(22) 申请日 2015.11.09

(65) 同一申请的已公布的文献号
申请公布号 CN 108353036 A

(43) 申请公布日 2018.07.31

(85) PCT国际申请进入国家阶段日
2018.05.08(86) PCT国际申请的申请数据
PCT/SE2015/051185 2015.11.09(87) PCT国际申请的公布数据
W02017/082778 EN 2017.05.18(73) 专利权人 瑞典爱立信有限公司
地址 瑞典斯德哥尔摩(72) 发明人 B·瓦尔加 J·法尔卡斯
S·鲁菲尼

(74) 专利代理机构 北京德琦知识产权代理有限公司 11018

代理人 史迎雪 王琦

(51) Int.Cl.

H04L 47/41 (2022.01)

H04L 47/32 (2022.01)

H04L 47/283 (2022.01)

H04L 45/42 (2022.01)

H04L 47/2483 (2022.01)

H04L 47/2441 (2022.01)

(56) 对比文件

US 2005058149 A1, 2005.03.17

CN 104641609 A, 2015.05.20

CN 101843140 A, 2010.09.22

CN 101459598 A, 2009.06.17

US 9179477 B2, 2015.11.03

审查员 姜佳轮

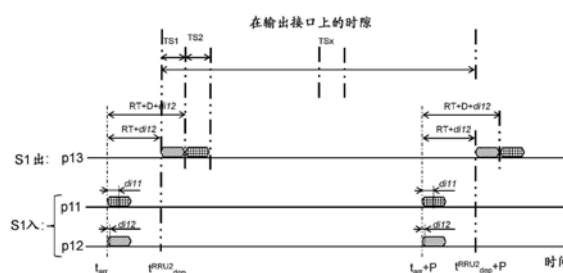
权利要求书4页 说明书12页 附图9页

(54) 发明名称

用于通信网络的分组处理技术

(57) 摘要

本公开一般涉及通过网络元件进行的分组处理,该网络元件传输相同通信量处理类(例如以太网快速通信类)的第一和第二分组流。通过网络元件执行的方法方面包括从网络控制器接收定义第一和第二分组流分组之间的相对转发顺序的信息的步骤。在网络元件的至少一个入口端口上接收到第一和第二分组之后,可以确定第一分组属于第一分组流,以及第二分组属于第二分组流。然后将第一和第二分组按照从网络控制器接收到的信息所定义的顺序转发到网络元件的至少一个出口端口。



1. 一种通过网络元件 (20) 处理分组的方法, 该网络元件传输相同通信量处理类的第一和第二分组流, 该方法包括:

从网络控制器 (10) 接收定义第一和第二分组流分组之间的相对转发顺序的信息;

在所述网络元件 (20) 的至少一个入口端口 (28) 上接收第一和第二分组;

确定所述第一分组属于所述第一分组流和所述第二分组属于所述第二分组流; 和

按照接收到的信息所定义的顺序向所述网络元件 (20) 的至少一个出口端口 (30) 转发所述第一和第二分组,

其中所述相对转发顺序被定义为, 使得它根据定时方案控制所述第一和第二分组的转发, 其中所述定时方案被用于定义所述第一和第二分组的有序时隙, 其中所述方法由通信网络 (100) 中的多个网络元件 (20) 来执行, 并且其中所述多个网络元件 (20) 全部应用于所述第一和第二分组的相同时隙顺序。

2. 根据权利要求1所述的方法, 其中

所述相对转发顺序通过用于所述第一和第二分组流中的至少一个的至少一个延迟来定义; 并且进一步包括

在所述转发步骤之前将所述至少一个延迟应用于所述第一和第二分组中的至少一个。

3. 根据权利要求2所述的方法, 其中

所述网络元件 (20) 位于传输网络域的边缘, 其中所述延迟被应用于控制所述第一和第二分组进入所述传输网络域的进入顺序。

4. 根据权利要求1所述的方法, 其中

所述第一和第二分组流中的每一个是恒定比特率流。

5. 根据权利要求4所述的方法, 其中

所述第一和第二分组流具有等于或数倍于公共基本比特率的相应比特率。

6. 根据权利要求1所述的方法, 其中

从所述网络元件和所述网络控制器 (10) 中的至少一者的角度来看, 所述第一和第二分组在所述网络元件 (20) 上的到达时间是确定性的。

7. 根据权利要求1所述的方法, 其中

所述第一和第二分组都在比至少一个所述第一和第二分组的时间扩展小的时间间隔内被接收。

8. 根据权利要求1所述的方法, 其中

所述网络元件 (20) 具有至少两个入口端口 (28); 并且其中所述第一和第二分组通过不同的入口端口 (28) 来接收。

9. 根据权利要求1所述的方法, 其中

所述第一和第二分组通过单个入口端口 (28) 来接收, 并且被封装在单个数据传输实体中和/或具有公共标识符。

10. 根据权利要求1所述的方法, 其中

所述第一和第二分组被转发给相同的出口端口 (30)。

11. 根据权利要求1所述的方法, 其中

两个或更多个不同的通信量处理类级被定义, 其中所述第一和第二分组流在较低类级上属于不同的通信量处理类, 以及在较高类级上属于相同的通信量处理类。

12. 根据权利要求1所述的方法, 其中

至少一个第三分组流, 其不具有通信量处理类或具有比所述第一和第二分组流更低的通信量处理类, 通过所述网络元件 (20) 来传输, 并且进一步包括:

在所述至少一个入口端口 (28) 上接收第三分组;

确认所述第三分组属于所述第三分组流; 并且执行以下步骤的至少一个:

阻止所述第三分组转发到所述出口端口 (30) 直到所述第一和第二分组已经被转发到所述出口端口 (30); 和

阻止通过所述出口端口 (30) 发送所述第三分组直到所述第一和第二分组已经通过所述出口端口 (30) 被发送。

13. 根据权利要求12所述的方法, 其中

通过根据在IEEE 802.1Qbv、IEEE 802.1Qbu和IEEE 802.3br中的至少一者中定义的技术优先化所述第一和第二分组的转发或发送, 阻止所述第三分组被转发到所述出口端口 (30) 或通过所述出口端口 (30) 来发送。

14. 根据前述权利要求1所述的方法, 其中

所述分组流构成以太网层通信量。

15. 一种通过网络元件 (20) 控制分组处理的方法, 该网络元件传输相同通信量处理类的第一和第二分组流, 所述方法通过网络控制器 (10) 来执行, 并且所述方法包括:

确定定义所述第一和第二分组流分组之间的相对转发顺序的信息; 和

发送所述信息给具有至少一个分组入口端口 (28) 和至少一个分组出口端口 (30) 的至少一个网络元件 (20), 其中所述信息被配置为对该相对转发顺序进行编程, 所述网络元件 (20) 以该相对转发顺序将通过所述入口端口 (28) 接收的所述第一和第二分组流的各个分组转发给所述出口端口 (30),

其中所述相对转发顺序被定义为, 使得它根据定时方案控制所述第一和第二分组的转发, 其中所述定时方案被用于定义所述第一和第二分组的有序时隙, 并且其中多个网络元件 (20) 全部应用于所述第一和第二分组的相同时隙顺序。

16. 根据权利要求15所述的方法, 其中

定义所述相对转发顺序的信息被发送到多个网络元件 (20), 以在整个通信网络 (100) 中定义相同的分组顺序。

17. 根据权利要求15或16所述的方法, 其中

所述相对转发顺序通过用于所述第一和第二分组流分组中的至少一个的至少一个延迟来定义。

18. 根据权利要求17所述的方法, 进一步包括

基于以下项的至少一个计算所述至少一个延迟:

所述第一和第二分组流分组在所述网络元件 (20) 中的驻留时间; 和

在发送所述分组的一个或更多个上游网络元件与所述网络元件 (20) 之间的一个或更多个链路延迟。

19. 根据权利要求15所述的方法, 进一步包括

基于以下项的至少一个计算所述至少一个延迟:

以所述第一和第二分组流中的至少一个为基础的比特率; 和

发送所述分组的一个或更多个上游网络元件的分组发送时间。

20. 根据权利要求19所述的方法, 其中

所述网络元件(20)位于传输网络域的边缘, 以及所述一个或更多个上游网络元件位于所述传输网络域之外并且直接连接所述网络元件(20)。

21. 一种网络元件(20), 其被配置为传输相同通信量处理类的第一和第二分组流, 该网络元件包括:

接口(24), 其被配置为从网络控制器(10)接收定义第一和第二分组流分组之间的相对转发顺序的信息;

至少一个入口端口(28), 其被配置为接收第一和第二分组;

处理器(22), 其被配置为确定所述第一分组属于所述第一分组流和所述第二分组属于所述第二分组流, 并按照接收到的信息所定义的顺序向所述网络元件的所述至少一个出口端口(30)转发所述第一和第二分组,

其中所述相对转发顺序被定义为, 使得它根据定时方案控制所述第一和第二分组的转发, 其中所述定时方案被用于定义所述第一和第二分组的有序时隙, 其中多个网络元件(20)全部应用于所述第一和第二分组的相同时隙顺序。

22. 根据权利要求21所述的网络元件, 其中

所述网络元件(20)是第二层网络桥。

23. 根据权利要求21所述的网络元件, 其中

所述网络元件(20)是传输网络域的边缘节点。

24. 根据权利要求21所述的网络元件, 其中

所述网络元件(20)被配置为执行权利要求1至17中任一项所述的方法。

25. 一种网络控制器(10), 其被配置为通过网络元件(20)控制分组处理, 该网络元件传输相同通信量处理类的第一和第二分组流, 该网络控制器包括:

处理器(12), 其被配置为确定定义所述第一和第二分组流分组之间的相对转发顺序的信息; 和

接口(24), 其被配置为发送所述信息给至少一个具有至少一个分组入口端口(28)和至少一个分组出口端口(30)的网络元件(20), 其中该信息被配置为对该相对转发顺序进行编程, 所述网络元件(20)将通过所述入口端口(28)接收的所述第一和第二分组流的各个分组转发给所述出口端口(30),

其中所述相对转发顺序被定义为, 使得它根据定时方案控制所述第一和第二分组的转发, 其中所述定时方案被用于定义所述第一和第二分组的有序时隙, 其中多个网络元件(20)全部应用于所述第一和第二分组的相同时隙顺序。

26. 根据权利要求25所述的网络控制器, 其中

所述网络控制器(10)被实现为软件定义网络SDNc的控制实体。

27. 根据权利要求25所述的网络控制器, 其中

所述网络控制器(10)被配置为执行权利要求15至20中任一项所述的方法。

28. 一种通信网络(100), 其包括根据权利要求21至24中任一项所述的一个或更多个的网络元件(20), 以及根据权利要求25至27中任一项所述的网络控制器(10)。

29. 根据权利要求28所述的通信网络(100), 其中一个或更多个网络元件(20)被前传网

络域包括或连接前传网络域。

用于通信网络的分组处理技术

技术领域

[0001] 本公开一般涉及通信网络。特别地，一种用于通信网络的分组处理技术被描述，其中多个分组流被传输。该技术可以方法、设备、网络 and 计算机程序的形式来实现。

背景技术

[0002] 时间关键(time critical)的应用产生的通信量(traffic)通常在专有专用网络(如Cobranet, ISO/IEC 62439, 61158或61784)和航空电子全双工交换以太网(AFDX)中传输。最近的发展尝试将时间关键的通信量转移到分组网络，例如第二层桥接网络，其也可以处理其他要求不太严格的通信量。由于这些发展，带有例如音频视频桥接(AVB)的商用以太网正被引入并用于工业自动化、专业音频处理和车载网络。

[0003] 正在考虑引入第二层桥接网络的另一个领域是移动通信网络的前传领域。该前传领域连接移动运营商的无线设备和无线设备控制器。该连接在技术上具有挑战性的延迟和分组延迟变化(PDV)要求，例如正如由通用公共无线接口(CPRI)所定义的那样。鉴于这些要求，目前前传领域不是由桥接网络提供的，而是由光纤网络中的专用lambda连接提供的。在另一方面，在延迟和PDV可以保持较低的情况下，对于前传领域，由时间关键的通信量和其它通信量共享网络资源的好处也将是有趣的选择。

[0004] 一种降低时间关键的通信量的延迟和PDV的方法是不同通信量类型的分类，以使时间关键的通信量可以在通信网络中以比其它通信量更高的优先级来传输。在以太网标准中，例如，定义八个通信量类别，以允许按优先次序处理各个通信量类别。

[0005] 以太网标准中的更新发展允许八个通信量类别之一声明为在以太网桥中接收特定处理的快速通信量类别，以选择性地降低相关联的通信量的分组延迟。例如，IEEE标准草案802.1Qbu“帧抢占”和802.3br“散布式快速通信量”建议暂停较少时间关键的(“可抢占”)通信量的传输，以利于时间关键的快速通信量。另一方面，IEEE标准草案802.1Qbv“调度通信量的增强”目标是通过针对明确定义的定时排空传输队列来减少分组延迟。

[0006] 术语“出口”和“输出”以及“入口”和“输入”分别在本文中可互换使用。此外，术语“帧”和“分组”在本文中可互换使用。

[0007] 已经发现，尽管在通信网络中减少延迟和PDV的各种方法，单个分组仍可能遭受显著的延迟，以及分组流可能遭受相当大的PDV。作为一个示例，在网络元件端口处两个同时接收到的用于服务的不同分组流的快速通信量分组的竞争可能导致输掉该竞争的分组的显著的排队延迟，因为它们共享相同的资源(即专用于快速通信量的资源)。除了总分组延迟的增加之外，输掉的分组流额外地遭受PDV累积，因为该输掉的分组流实质上“继承”了获胜分组流的PDV。

[0008] 因此，快速通信量本身的声明不能保证每个快速通信量分组流的低延迟和低PDV。例如，帧抢占不能帮助解决所有属于快速通信量类的时间关键的分组之间的竞争条件。然而，如果无法控制这种分组流的竞争情况，则需要额外的缓冲来减少PDV。另一方面，这种额外的缓冲将进一步增加由所缓冲的分组经历的延迟。类似的问题会发生在其它情况中，其

中相同的通信量处理类别的不同分组流需要在通信网络中被处理。

[0009] 因为上述原因,特别时间关键的应用仍然必须依靠专有专用网络。这是令人不满意的情况,因为第二层以太网桥接网络和类似的网络是容易获得的。

发明内容

[0010] 需要这样的一种分组处理技术:其避免上述现有技术中的一个或更多个的缺点,或其它的缺点。

[0011] 根据第一方面,通过网络元件处理分组的方法被提供,该网络元件传输相同通信量处理类的第一和第二分组流。该方法包括从网络控制器接收定义第一和第二分组流分组之间的相对转发顺序的信息,并在网络元件的至少一个入口端口上接收第一和第二分组。该方法进一步包括确定第一分组属于第一分组流和第二分组属于第二分组流,并且按照由所接收的信息定义的顺序向网络元件的至少一个出口端口转发第一和第二分组。

[0012] 通信量处理类可以指示时间关键的通信量。第一和第二分组流可以具有相同的定时约束(例如就在网络元件上优先化的入口和/或出口处理而言),因为它们属于相同的通信量处理类。

[0013] 在某些情况下,相对转发顺序可以通过用于第一和第二分组流中的至少一个的至少一个延迟来定义。因此,信息可以包括与第一分组流的分组有关的延迟信息,与第二分组流的分组有关的延迟信息,或它们的组合。可以理解的是,在某些变型中,第一和第二分组流分组之间的相对转发顺序可以通过选择性地只延迟第一分组流分组或只延迟第二分组流分组来定义。

[0014] 由网络元件执行的方法可以进一步包括将至少一个延迟应用到第一和第二分组中的至少一个。该延迟可以在转发步骤之前被应用。因此,在一些实现中,第一和第二分组流分组中的一个或两个的出口处理和出口传输中的至少一者可以选择性地被延迟。

[0015] 网络元件可以位于通信网络中的任何位置。作为示例,该网络元件可以位于通信网络的传输网络域的边缘(例如在朝着发送分组的上游网络元件的边缘上)。在这样的一种情况下,延迟可以被应用于控制第一和第二分组进入传输网络域的进入顺序。传输网络域进而可以包括负责在传输网络域中传输第一和第二分组流的一个或更多个核心元件。本文中提到的网络元件也可以采取这样的核心元件的形式。

[0016] 相对转发顺序可以被定义为,使得它控制第一和第二分组的绝对或相对转发时间。该转发可以根据定时方案在某些变型中被控制。该定时方案可以在网络元件本地应用。或者,该定时方案可以被全局地应用在传输网络域中或更大的通信网络中。该定时方案一般可以通过本地或全局计时机制来定义。

[0017] 在一些情况下,该定时方案可以被用于定义用于第一和第二分组的有序时隙。该时隙可以在网络元件本地或以全局方式(例如在传输网络域级别或在更大的通信网络级别)来应用。在一个示例中,该时隙包括至少一个较早的时隙和至少一个较晚的时隙,其中这两个时隙中的第一个被一致地分配给第一分组流分组,并且第二时隙被一致地分配给第二分组流分组。时隙可以被用于定义有序分组序列。

[0018] 在一个变型中,时隙彼此紧密跟随。在另一个变型中,保护间隔被插在两个连续的时隙之间。时隙可以具有相同的大小或不同的大小。

[0019] 本文所提供的方法可以由通信网络中的多个网络元件来执行。在这种情况下，多个网络元件可以针对第一和第二分组都应用相同的时隙顺序。应该注意的是，各个网络元件可能但并非必须彼此同步。在这种情况下，一个网络元件可以相对于另一网络元件以非同步的方式应用其定时方案。

[0020] 第一和第二分组流中的每一个可以是恒定的比特率流。对于每个恒定的比特率流，一个或多个不同的比特率可以被定义。第一和第二分组流可以具有相应的比特率，其等于公共基本比特率或是公共基本比特率的数倍。

[0021] 从网络元件和网络控制器中的至少一者的角度来看，第一和第二分组在网络元件上的(绝对或相对)到达时间可以是确定性的。作为示例，网络元件和网络控制器中的至少一个可以知道影响到达时间的基本因素。这样的因素可以包括链路延迟、分组流比特率等中的一个或多个。

[0022] 第一和第二分组都可以在小于第一和第二分组中的至少一个的时间扩展的时间间隔内被接收(例如，在通过至少一个入口端口接收时)。在更一般的术语中，第一和第二分组可以由网络元件来接收，使得竞争条件可能发生，特别是关于网络元件上的入口处理。

[0023] 网络元件可以具有至少两个入口端口。在这种情况下，第一和第二分组可以通过不同的入口端口来接收。或者，第一和第二分组可以通过单个入口端口来接收。在后一种情况下，第一和第二分组可以被接收封装在单个数据传输实体(例如更大的高级别的分组)中。附加地，或者替代地，第一和第二分组可以通过单个入口端口来接收，并携带公共标识符，例如公共虚拟局域网标识符(VID)。

[0024] 第一和第二分组可以被转发到网络元件的相同出口端口。或者，第一和第二分组可以被转发到不同的出口端口。

[0025] 在一些变型中，两个或多个不同的通信量处理类级可以被定义。在这种情况下，第一和第二分组流在较低类级上可以属于不同的通信量处理类，但在较高类级上属于相同的通信量处理类。通常，一个、两个或多个较低级通信量处理类可以被映射到单个较高层通信量处理类上。作为示例，八个以太网通信量类中的一个或多个可以被映射在以太网快速通信量类上，其进而构成较高类级上的(单个)通信量处理类。

[0026] 在一个实现中，至少一个不具有通信量处理类(例如不属于以太网快速通信量类)或具有比第一和第二流低的通信量处理类的第三分组流通过网络元件来传输。在这样的情况下，该方法可以进一步包括在至少一个入口端口上接收第三分组，并且确定该第三分组是否属于第三分组流。在第三分组属于第三分组流的情况下，它可以被阻止转发到出口端口，直到第一和第二分组已经被转发到出口端口。附加地，或者替代地，通过出口端口发送第三分组(“出口发送”)可以被阻止，直到第一和第二分组已经通过该出口端口被发送。可以通过根据在IEEE 802.1Qbv、IEEE 802.1Qbu 和IEEE 802.3br中的至少一个所定义的技术优先化第一和第二分组的转发和/或发送来阻止第三分组的转发和/或发送。

[0027] 通过网络元件来传输的分组流一般可以构成以太网层通信量。附加地，或者替代地，一个或多个的分组流，特别是第一和第二分组流，可以构成CPRI分组流。

[0028] 根据另一方面，一种通过传输相同通信量处理类的第一和第二分组流的网络元件控制分组处理的方法被提出。该方法通过网络控制器来执行，并包括确定定义第一和第二分组流分组之间的相对转发顺序的信息。该方法进一步包括发送该信息给至少一个具有至

少一个分组入口端口和至少一个分组入口端口的网络元件,其中该信息被配置为对该相对顺序进行编程,网络元件以该相对顺序将通过入口端口所接收的第一和第二分组流的各个分组转发到出口端口。

[0029] 定义该相对转发顺序的信息可以通过网络控制器被发送给多个网络元件,以在整个通信网络中定义相同的分组顺序。相应的信息对于所有的网络元件可以是相同的,或对于不同的网络元件是不同的。

[0030] 如上所述,该相对转发顺序可以通过用于第一和第二分组流分组中的至少一个的至少一个延迟来定义。网络控制器可以计算至少一个延迟。至此,网络控制器或与网络控制器通信的任何网络元件能够测量网络延迟参数。

[0031] 作为示例,基于第一和第二分组流分组在网络元件中的驻留时间,至少一个延迟可以被计算。附加地,或者替代地,基于在一个或多个发送分组的上游网络元件(不一定是分组源)和接收网络元件之间一个或多个的链路延迟,至少一个延迟可以被计算。附加地,或者替代地,基于以第一和第二分组流中的至少一个为基础的比特率,至少一个延迟可以被计算。附加地,或者替代地,基于一个或多个发送分组的上游网络元件的分组发送时间,至少一个延迟可以被计算。接收网络元件一般可以位于传输网络域的边缘。上游网络元件可以位于传输网络域之外,并且直接连接接收网络元件。

[0032] 还提供了一种计算机程序产品,其包括程序代码部分,以当该计算机程序产品通过一个或多个处理器来执行时,执行本文所提及的方法和方法步骤。该一个或多个处理器可以被定位在单独的网络节点上,或可以由分布式计算系统包括。该计算机程序产品可以被存储在计算机可读记录介质上,例如半导体存储器、DVD-ROM、CD-ROM等。该计算机程序产品也可以被提供用于通过通信连接来下载。

[0033] 根据又一方面,一种被配置为传输相同通信量处理类的第一和第二分组流的网络元件被提供。该网络元件包括被配置为从网络控制器接收定义第一和第二分组流分组之间的相对转发顺序的信息的接口。该网络元件进一步包括至少一个被配置为接收第一和第二分组的入口端口。并且,该网络元件包括被配置为确定第一分组属于第一分组流和第二分组属于第二分组流,并且按照由接收到的信息所定义的顺序向网络元件的至少一个出口端口转发第一和第二分组的处理器。

[0034] 该网络元件可以是第二层网络桥(例如第二层以太网网络桥)。附加地,或者替代地,该网络元件可以是传输网络域的边缘节点。该网络元件一般可以被配置为执行本文所提及的任何方法和方法方面的步骤。

[0035] 根据另一个方面,一种被配置为通过传输相同通信量处理类的第一和第二分组流的网络元件控制分组处理的网络控制器被提供。该网络控制器包括被配置为确定定义第一和第二分组流分组之间的相对转发顺序的信息的处理器。该网络控制器进一步包括被配置为发送信息给至少一个具有至少一个分组入口端口和至少一个分组出口端口的网络元件的接口,其中该信息被配置为对相对顺序进行编程,网络元件以该相对顺序将通过入口端口所接收的第一和第二分组流的各个分组转发到出口端口。

[0036] 该网络控制器可以被实现为软件定义网络(SDNc)的控制实体。通常,网络控制器可以被配置为执行本文所提及的任何方法和方法方面的步骤。

[0037] 又一方面涉及一种包括一个或多个本文所提及的网络元件和本文所提及的控

制器的通信网络。一个或更多个网络元件可以由前传领域来包括或连接前传领域。

附图说明

[0038] 在下文中,将参照附图更详细地描述本公开的实施例和示例性方面,其中:

[0039] 图1示出了根据本公开的具有网络控制器实施例和一个或更多个网络元件实施例的通信网络的实施例。

[0040] 图2A和2B示出了根据本公开的另一个网络控制器实施例和另一个网络元件实施例。

[0041] 图2C示出了根据本公开的通过网络控制器执行的方法实施例和通过网络元件执行的方法实施例的流程图。

[0042] 图2D示出了根据本公开的通过一个或更多个网络元件执行的另一个方法实施例的流程图。

[0043] 图3示出了根据本公开的通信网络的另一个实施例。

[0044] 图4是示出在图3的网络中不同分组流的两个分组之间的竞争情况的示意性时间图。

[0045] 图5是示出本公开的另一个方法实施例的流程图。

[0046] 图6是示出本公开的又一个方法实施例的流程图。

[0047] 图7和8是示出根据本公开的分组处理实施例的示意性时间图。

具体实施方式

[0048] 在以下描述中,出于解释而非限制的目的,阐述了特定细节,诸如特定网络协议和特定网络元件,以便提供对本公开的透彻理解。对于本领域技术人员来说显而易见的是,本公开可以在偏离这些具体细节的其他实施例中实施。

[0049] 本领域的技术人员将进一步意识到,本文所描述的功能、步骤和服务可以使用单独的硬件电路,使用与编程的处理器或通用计算机相结合的软件功能,使用专用集成电路(ASIC),和/或使用一个或更多个的数字信号处理器(DSP)来实现。还将意识到,虽然本公开主要被描述在方法和设备的上下文中,它也可以被体现在一个或更多个处理器,以及耦合到一个或更多个处理器的一个或更多个存储器中,其中一个或更多个存储器使用一个或更多个程序来编码,当程序在一个或更多个处理器上被执行时,它执行本文所描述的功能、步骤和服务。

[0050] 图1示出了通信网络100的一些组件的实施例。如图1所示,通信网络100包括网络控制器10,以及与网络控制10通信的一个或更多个网络元件20。可以理解的是通信网络100包括在图1中未示出的额外组件。

[0051] 图1的网络控制器10被配置为通过一个或更多个网络元件20来控制分组处理。至此,网络控制器10包括处理器12和被配置为与一个或更多个网络元件20通信的接口14。网络控制器10进一步包括具有程序代码的存储器16,程序代码当通过处理器12来执行时,将网络控制器10配置为实现本公开的方法和方面。

[0052] 以类似的方式,每个网络元件20包括处理器22、接口24和存储器26。接口24被配置用于与网络控制器10通信。存储器26存储程序代码,程序代码当通过处理器22来执行时,将

网络元件20配置为实现本公开的方法和方面。

[0053] 如图1所示,网络元件20进一步包括一个或多个入口端口28,以及一个或多个出口端口30。每个入口端口28被配置为接收一个或多个入口分组流,并且每个出口端口30被配置为发送一个或多个出口分组流。处理器22被配置为执行入口和出口处理操作。这些处理操作例如可以包括第二层处理(例如从网络元件20的角度来看,入口和出口分组流可以构成以太网层通信量)。在一些变型中,网络元件20因此可以被配置为通信网络100内的第二层以太网桥或交换机。

[0054] 图2A和2B分别显示了网络控制器10的功能性实施例和网络元件20 之一的功能性实施例。这些功能性实施例可以是基于上述参照图1所讨论的结构实施例。

[0055] 如图2A所示,网络控制器10包括信息确定模块140和信息发送模块 150。网络元件20包括信息接收模块160、分组接收模块170、确定模块 180、以及转发模块190。

[0056] 图2C在两个流程图200A、200B中示出了分别由网络控制器10和单个网络元件20执行的示例性方法实施例。网络控制器10和网络元件20 可以分别如上参照图1、2A和2B所述来配置。

[0057] 方法实施例目标一般是避免相同通信量处理类的入口分组流之间的竞争条件。特别地,该通信量处理类可以优先的或快速的类别,其将通信网络100中的时间关键的通信量优先于其它的(“后台”)具有较少时间关键限制的通信量。作为示例,时间关键的通信量可以是一项或更多项无线设备与一个或多个无线设备控制器之间的前传通信量。在这种情况下,图 1所示的一个或多个网络元件20可以位于一项或更多项无线设备与一个或多个无线设备控制器之间,并构成一部分或完整的前传连接(例如根据CPRI)。另一方面,后台通信量可以是任何其它的较少时间关键的通信量。可以理解的是,前传连接只是一个示例,并且本公开也可以在其它场景中被实现,例如工业自动化、车载网络、航空电子设备等。

[0058] 如图2C所示,通过网络控制器10来执行的方法包括步骤202:确定定义相同通信量处理类的两个分组流之间的相对转发顺序的信息。该转发顺序可以根据任何给定的要求来确定,但是也可以被任意定义。最重要的往往不是这样的顺序,而是该顺序由单独网络元件20并且在某些情况下跨多个网络元件20来一致地实现。

[0059] 相对转发顺序可以通过用于至少一个分组流的延迟来定义。在由单独的网络元件20处理的n个分组流的情况中,至少n-1个延迟被定义。定义相对转发顺序的信息可以针对每个网络元件20单独确定。在这种情况下,可以针对不同的网络元件定义相同分组流的不同延迟。

[0060] 可以由网络控制器100基于从通信网络100获得的信息(例如,通过测量)来计算定义相对转发顺序的信息。该计算可以基于以下项中的一项或更多项:单个网络元件20中的单个分组的驻留时间,在单个网络元件 20与例如发送分组的网络元件20上游的网络元件之间的一个或多个链路延迟,以分组流为基础的比特率,以及一个或多个上游网络元件的绝对或相对分组发送时间。

[0061] 一旦该信息在步骤202中已经被确定,它在步骤204中由网络控制器 10发送给一个或多个网络元件20以对其编程。关于图1,该信息通过处理器12来确定,并且该信息通过网络控制器10的接口14来发送。

[0062] 现在参考由网络元件20所执行的方法,在步骤204中由网络控制器 10发送的信息

在步骤206中由单独的网络元件20来接收。如图1所示,该信息将通过接口24来接收,然后被转发给网络元件20的处理器22,用于进一步处理和/或被存储在存储器26中。

[0063] 在进一步的步骤208中,属于不同分组流的分组通过网络元件20的一个或更多个入口端口26来接收。不同分组流的分组可以被接收封装在单个数据传输实体(例如更高级分组)中。替代地,或附加地,不同分组流的分组可以通过公共标识符来识别,例如公共入口端口标识符或VID。

[0064] 步骤208和步骤206可以以任何顺序来执行。典型地,步骤206先于步骤208,但是步骤206可以被重复多次,而步骤208被连续地执行。

[0065] 在进一步的步骤210中,网络元件20的处理器22确定接收到的分组中的第一个属于第一分组流,以及接收到的分组中的第二个属于第二分组流。步骤210可以以多种不同的方法来执行。例如,每个分组可以携带特定分组流的标识符(例如,在报头部分中)。因此,在接收到的分组与相关联的分组流之间的关联因此可以基于这样的分组流标识符来执行。

[0066] 在进一步的步骤212中,网络元件20的处理器22按照通过在步骤206中接收到的信息所定义的顺序转发第一和第二分组给一个或更多个出口端口30。步骤212中的相应排序可以通过根据在步骤206中接收到的信息选择性地延迟第一和第二分组来执行。因此,在步骤206中接收到的信息中传达的延迟可以被应用到第一和第二分组中的至少一个,以一致地实现预定义的转发顺序。以这种方式,具有不可预测的分组转发顺序的竞争条件可以被避免。

[0067] 在可选择的进一步的步骤214中,处理器22对转发到一个或更多个出口端口30的分组应用出口处理(例如,分组排队),并且控制分组(例如,来自一个或更多个队列)经由一个或更多个出口端口30到通信网络100中的下一个组件的传输。

[0068] 图2D示出了网络元件20的操作的另一种模式的流程图200C。在图2D中所示的步骤可以与在图2C中所示的步骤并行执行。

[0069] 在步骤216中,网络元件20接收相同的(例如较高级别)通信量处理类的第一和第二分组流的第一和第二分组。在步骤218中,网络元件20接收非专用通信量处理类或较低级别通信量处理类的第三分组流的第三分组。可以理解的是,步骤216和步骤218可以被并行执行。

[0070] 然后,在步骤220中,第三分组的传输被阻止,直到第一和第二分组已经被传输。在图2D中所示的操作例如可以被实现,以确保在步骤216中所接收的优先分组流的分组优先于诸如在步骤218中所接收的第三分组流之类的后台通信量。步骤220因此可以根据IEEE 802.1Qbv、IEEE 802.1Qbu和IEEE 802.3br中一个或更多个,特别是在网络元件20的出口端口处来执行。

[0071] 另一方面,在步骤218中接收到的第三分组流的第三分组可以绕过图2C的步骤212中针对第一和第二分组执行的任何分组延迟操作。因此,第三分组例如可以被立即从入口端口28转发到出口端口30(参见图1),而没有任何入口侧延迟,但是随后可能受到如上面提及的IEEE草案标准中的任一个所定义的分组抢占或类似的技术。

[0072] 在某些变型中,图1、2B、2C和2D中所示的网络元件20可位于传输网络域的边缘(例如,可以构成边缘节点),朝向位于传输网络域的外侧并发送分组的一个或更多个网络元件。在这种情况下,网络元件20因此可以通过步骤212中的排序操作来控制不同分组流的分

组进入传输网络域的进入顺序。例如通过选择性地延迟单个分组流的分组进入已进入的传输网络域,可以促进通过传输网络域的无竞争传输。此外,通过在步骤204 中由网络元件10 发送的信息,通过网络控制器10,通过网络元件20的“编程”,分组延迟和PDV变得更好的可跟踪和确定性的。当然,在某些变型中,传输网络域中的核心元件也可以已这种方式来编程。

[0073] 此外,全通信网络或其中的相关分组传输组件可以通过网络控制器10 来编程,使得当时间关键的分组流的分组由给定的网络元件来接收时,它将被服务和传输,而没有不必要的延迟(例如,没有遭受排队延迟)。换句话说,时间关键的分组流可以优先于后台分组流。如上文参考图2D所讨论的,根据如IEEE 802.1Qbv、IEEE 802.1Qbu和IEEE 802.3br中的一个或多个所定义的技术,属于这样的后台分组流的分组可以通过单独的网络元件20来阻止被转发到出口端口或通过该出口端口被传输。

[0074] 已经发现,此处提及的技术对于具有等于或数倍于公共基本比特率的比特率的恒定比特率流是特别有益的。在这种情况下,单个分组流的分组到达时间和跨多个分组流的相对分组到达时间变得确定。因此,在特定的分组流的分组上的专用延迟的应用可以极大地避免竞争条件。当然,相应的延迟不仅可以基于单个分组流的当前(恒定)的比特率,而且可以额外地考虑如上所述的在网络元件处的分组驻留时间、链路延迟和分组发送时间中的一项或更多项。

[0075] 相对转发顺序可以被定义(例如,通过延迟),使得它根据被本地应用在网络元件20上或在更大范围上(例如,跨较大的一组网络元件20) 的定时方案来控制分组的转发。该定时方案可以被用于定义用于来自不同分组流的分组的传输的有序时隙。特别地,对于特定的通信量处理类(例如,以太网快速通信量类)的每个分组流,专用的时隙可以被每个网络元件20保留。在某些变型中,跨多个网络元件20应用用于相同通信量处理类的分组流的相同时隙顺序。这样,该时隙概念特别受益于分组流,该分组流是从如上所述的公共基本比特率获得的恒定比特率流。

[0076] 可以理解的是,分组排序操作可导致单个分组流的分组的某个附加延迟。另一方面,所有分组流的PDV可以被极大地减小,并且时间关键的应用的PDV要求典型地比相关联的延迟要求更严格。因此,就抖动(如典型地要求减少PDV)和缓冲而言,网络元件20可以用最小的努力来处理传入的时间关键的通信量(例如,CPRI通信量)。因此,没有进一步的延迟被添加到分组传输。

[0077] 此外,本文所提及的技术可以容易地与诸如IEEE 802.1Qbv、IEEE 802.1Qbu和IEEE 802.3br中的一个或多个之类的通信量优先化方案(本身不能保护一个时间关键的分组流不受到另一个时间关键的分组流的影响)相结合。因此,还可以减小后台分组流对时间关键的分组流的冲击。

[0078] 在本公开的部分基于上述参照图1和2A至2D讨论的实施例的进一步实施例的以下描述中,无线网络前传场景将参照图3更详细地被讨论。

[0079] 图3显示了通信网络300的实施例,其中所谓的远程无线电单元(RRU)形式的无线电设备和位于两个站点A、B处的所谓的数字单元(DU)形式的无线电设备控制器在以太网交换传输网络域上通过CPRI通信。传输网络域的网络元件被表示为S1到S7,并且可以被实现为以太网桥或交换机。在图3中,每个RRU被用图案标记,并且完全相同的图案被用于由相

应RRU产生的分组。作为示例,图案是用于RRU1和由RRU1 产生的分组流的“网”,以及用于RRU6和由RRU6产生的分组流的“方格”。图3示例性示出了上行链路专利流,但是本文所提及的原理同样适用于下行链路分组流。

[0080] 在图3所示的场景中,或者在类似的场景中,因为以下原因,可以被实现有序分组转发机制(例如,通过分组延迟)的实施和(可选择的)时隙的创建。首先,分组流由类似的源产生,这意味着在CPRI的情况下,分组流是恒定比特率流,其分组速率等于或数倍于公共基本比特率(例如, 614.4Mbps)。第二,在不同RRU的分组流之间的相对分组发送时间是已知的,这是因为它们通常利用相同的同步参考,并因此利用相同的定时方案,并且典型地同时发送它们的分组。因此,可以假定所有RRU都可以访问相同的时间同步参考(例如因为它们属于相同的运营商,或者因为不同的无线电运营商可以访问相同的同步服务)。在传输网络(例如,以太网)域中的RRU(一般来说,无线接入网络域)和网络元件可以使用不同的时钟源。第三,穿过传输网络域中的每个链路和网络元件的延迟是已知的(即,可以被测量)。在本场景中,链路延迟可以包括传播延迟,并且典型地通过特定链路的长度来确定。穿过网络元件的延迟对应于在单个网络元件上转发分组所花费的时间,并且在本文中也称为驻留时间(RT)。

[0081] 通过网络元件S1至S7进行的分组处理由中央网络控制器来控制,在本实施例中,中央网络控制器采取软件定义网络控制器(SDNc) 10的形式。SDNc 10以及网络元件S1至S7(在图1和2B至2D中用参考数字20 表示)一般可以被配置为执行上述参照图2C和2D所讨论的方法步骤。因此,SDNc 10可以确定定义分组流之间的相对转发顺序的信息,并且可以使用该信息对网络元件S1至S7上的每个分组流的可调延迟参数和/或其它的参数进行编程。

[0082] 根据使用情况,可能只需要编程传输网络域边缘的网络元件,或者传输网络域的每个网络元件。在以下参照图3所描述的示例中,将假设入口延迟被应用在每个网络元件S1至S7(即每一跳)上,以确保用于相同的通信量处理类的多个分组流的一致分组转发顺序,并避免在网络元件S1 至S7的输出队列中的竞争。下一跳的竞争情况以同样的方式通过适当选择的入口延迟来解决。

[0083] 每个RRU产生类似的CPRI分组流(例如,它们具有相同的恒定比特率)。而且,将假设每个分组流被封装在具有相同大小的以太网帧中。这些假设简化了时隙的使用,因为它们也可以具有相同的大小。然而,这些是可选择的特征。

[0084] 应该注意的是,以太网网络是异步网络,因此,例如,特定时隙的开始时间不被绑定到明确的时间点,而是相对于其它的时隙而言。正如这里所理解的,时隙对应于用于朝向一个或更多个出口端口的分组转发的特定时间间隔。一个时隙可以被另一个时隙紧跟,或者可以在两个相继的时隙之间提供保护间隔(例如,防止通信网络300中的定时不确定)。

[0085] 图3中所显示的网络拓扑代表了典型的前传部署。在这样的部署中,由于CPRI的严格的同步和延迟要求,通常相对简单的拓扑(例如树形结构、部分网状结构、有限的跳数等)将被使用。图3中的传输网络域的拓扑包括构成边缘节点的三个网络元件S1、S2和S7,以及构成传输网络域的核心节点的四个网络元件S3、S4、S5和S6。

[0086] 网络元件S1至S7中的每一个包括一个或更多个入口端口,以及一个或更多个出口端口(也参见图1)。传输网络域中的各个网络元件S1至 S7的端口被连续编号。例如,网络元件S1包括端口p11、p12和p13,其中在图3所示的上行链路处理场景中,端口p11和p12被配置

为入口端口,并且p13被配置为出口端口。SDNc 10被配置为通过定义如上所解释的不同分组流的分组之间的相对转发顺序的信息(即,延迟值)对网络元件S1至S7中的每一个进行编程。用于特定的网络元件S1至S7的由SDNc 10设置的入口延迟值通过 d_{ikm} 来表示,其中 k 是端口编号(例如,用于网络元件S1的入口端口p11的 d_{i11})。因此,可以在每个入口端口的基础上和/或每个分组流的基础上确定和设置入口延迟。

[0087] 如果链路将RRU连接到传输网络域的边缘上的网络元件S1、S2和S7 中的一个,则如图3所示的链路具有其由 d_i 表示的传输延迟,其中, i 标识特定的RRU。针对各个网络元件S1至S7之间的链路,链路具有其由 d_{ij} 所表示的传输延迟,其中, i 和 j 标识两个互连网络元件S1至S7。链路延迟测量可以使用诸如IEEE 1588对等延迟测量之类的标准化方法来执行。

[0088] 在下文中,在网络元件S1上分别来自RRU1和RRU2的两个不同的分组流的分组之间示例性竞争情况的发生将参照图4中的示意性时间图来解释。因此,图4显示了其中不应用本文提供的有序分组转发机制的场景。

[0089] 在图4的场景中,将假设RRU以特定分组流的两个后续分组之间的时间周期 P 创建具有相同的恒定比特率的周期性通信量。示例性地假设 d_1 等于 d_2 ,RRU1和RRU2的分组在相同的时间 t_{arr} 到达网络元件S1的入口端口p11和p12。入口端口p11和p12是网络元件S1的用户网络接口(UNI) 端口。来自RRU2的分组在驻留时间 RT 之后由出口端口p13来发出,在分组服务时间(表示为 D)后,来自RRU1的分组被发出。因此,来自 RRU2的分组的离开时间 t_{dep}^{RRU2} 在来自RRU1的分组的离开时间 t_{dep}^{RRU1} 之前。然而,来自RRU1和RRU2的分组的该顺序是不可预测的,如在 $t_{arr}+P$ 到达的随后分组的所示,其中, t_{dep}^{RRU1} 在 t_{dep}^{RRU2} 之前。该竞争情况导致不可预测的网络行为,以及对于诸如图3中的示例性CPRI分组流之类的时间关键的分组流而言不允许的PDV增加。

[0090] 为了避免图4中所示的竞争情况,网络元件S1至S7被配置为考虑定义来自不同RRU的分组流之间的相对转发顺序的信息,如上述参照图2C 一般地解释的那样。在本示例性场景中,不同分组流分组之间的相对转发顺序由网络元件S1至S7的入口端口处的可编程入口延迟来定义。

[0091] 入口延迟通过SDNc 10在每个入口端口的基础上被定义,正如在图5 的流程图500中所显示的那样。

[0092] 在初始步骤502,SDNc 10等待图3中的通信网络的变化,包括初始网络设置。然后,在步骤504,通信量和网络信息被收集,例如包括参数 d_i 、 d_{ikm} 、 RT 、 P 等中的一个或多个。

[0093] 在步骤506中,要应用在网络元件S1至S7的入口端口上的入口延迟按通信量流来计算。步骤506一般对应于图2C中的步骤202。在进一步的步骤508中,在步骤506确定的入口延迟在网络元件S1至S7上被编程。步骤508一般对应于图2C中的步骤204。

[0094] 步骤508中的编程被执行,以使针对特定分组流的特定入口端口的入口延迟被设置,使得保证相同通信量处理类的不同分组流的分组顺序(并且可选地,相对时间差)在特定的网络元件S1至S7上始终保持相同。例如,RRU2的“阴影”分组被转发,使得它们总是在网络元件S1和S3之间的链路上通过图3所示的网络元件S1的出口端口p13在RRU1的“网状”分组之前被传输。

[0095] 入口延迟可以在传输网络域的边缘处的网络元件S1、S2和S7(即第一跳节点)上被编程,使得如果两个或多个RRU的流被聚合,则“分组序列”被形成(参见由网络元件S1的

出口端口p13所传输的分组序列)。通过使用有序的时隙,能够保证特定分组流的分组(来自一个RRU或DU)总是在通过特定出口端口输出的序列中占据相同的位置,这可以应用于传输网络域的每个出口端口。

[0096] 图6显示了流程图60,其更详细地示出了将入口延迟应用于网络元件 S1至S7之一上的各个分组的方法。

[0097] 如图6所示,最初,特定的网络元件S1至S7在入口端口上等待新分组(步骤602)。当新分组被接收时,它在步骤604中对照一个或更多个选择标准来检测。该选择标准可以包括特定的流标识符、特定的通信量处理类等中的一个或更多个。步骤604对应于图2C中的步骤210的方面。

[0098] 如果发现一个或更多个选择标准被满足,该方法进行到步骤606,并在继续进行步骤608之前,将接收到的分组延迟预定义的时间间隔 t_d 。入口延迟 t_d 一般意味着如果分组被接收到,然后它在网络元件S1至S7的每一个中的进一步处理仅在它已经延迟 t_d 之后开始。步骤606一般对应于图2C中的步骤212的方面。如果选择标准不被满足,该方法从步骤604 直接进行至步骤608。

[0099] 在步骤608中,分组被朝着出口端口转发。作为示例,该分组可以被置于相应网络元件S1至S7的出口侧的传输队列中。步骤608一般对应于图2C中的步骤212和可选的步骤214的方面。

[0100] 入口延迟可以被用于所有入口端口(UNI端口和以太网传输内部中继线)上。诸如IEEE 802.1Qbv、IEEE 802.1Qbu和IEEE 802.3br之类的额外的方法可以被用于保证时间关键的分组能够在相应的网络元件S1至S7 上传输,而没有任何(进一步的)延迟,并且保证属于时间关键的通信量的分组优先于后台通信量。

[0101] 如上所述,入口延迟可以被用于将每个接收到的分组“推”到其在有序时隙集内的专用位置。因此,分组在下一个网络元件(即图3中的S3 和S5)上的到达时间变得确定。

[0102] 在下文中,参考图3中的示例性通信网络和图7和8中示出的时间图,将更详细地解释使用上述的入口延迟机制可实现的优点。

[0103] 在图7所示的场景中,RRU1和RRU2的分组再次分别在相同的时间 t_{arr} 到达网络元件S1的入口端口t11和p12。与图4所示的情况不同,网络元件S1的入口端口p11和p12实现了上面讨论的入口延迟机制。如图7 所示,入口延迟在入口端口p11上被设置成di11,在入口端口p12上被设置成di12。在某些变型中,对于至少一个入口端口(或者,如果多个分组流到达相同的入口端口,分组流之一),该延迟也可以是零。

[0104] 如上所述,考虑到分组流的特性和某些网络参数,延迟通过SDNc10 来设置。例如,如果RRU具有恒定比特率分组流,并且它们同时发送分组,那么在特定的入口端口上被用于入口延迟的值取决于——在图7的场景中——链路延迟d1和d2,以及所期望的分组顺序。如果链路延迟相同(即,如果d1等于d2),并且如果目标是在RRU1的分组之前发送出RRU2 的分组,那么di11必须被设置为大于di12(即,di11>di12,其中di11<D),这总是保证图7所示的分组顺序和时隙安排。

[0105] 因此,延迟值di11和di12可以被设置为实现如图4的上部所示的顺序时隙。具体地,时隙被设置为,使得来自RRU1的分组总是在第一时隙 TS1中被传输,而来自RRU2的分组总是在随后的第二时隙TS2中被传输。

[0106] 图8示例性地说明了网络元件S3的操作。如图3所示,网络元件S3 是传输网络域的核心节点。它在其两个入口接口p31和p32上实现了本文提及的入口延迟机制。

[0107] 网络元件S3聚合了六个时间关键的分组流,但将分组流转发到不同的出口端口。也就是说,五个分组流被转发到出口端口p34,而一个分组流被转发到出口端口p33。

[0108] 最初,网络元件S3在入口端口p31接收RRU2和RRU1的已排序的分组(参见图7),而RRU3、RRU4和RRU5的已排序的分组在入口端口p32被接收。网络元件S3必须保证这两个分组序列之间的顺序被保持,使得RRU2和RRU1的分组总是在出口端口p34上的其余分组之前被发出。因为这个理由,入口端口p32的入口延迟 d_{i32} 必须被设置为,使得入口端口p32的分组被延迟,直到入口端口p31的第二分组已经被相应的出口端口服务。如果两个分组序列在相同的时间 t_{arr} 到达网络元件S3,那么,例如, d_{i31} 可以被设置为零,并且 d_{i32} 可以被设置为, $D < d_{i32} < 2 * D$ 。如上所述,SDNc 10知道整个通信网络中的链路延迟,包括 d_{13} 和 d_{32} 。因此,SDNc 10可以如上所述确定并设置 d_{i31} 和 d_{i32} 。

[0109] 由于参照图8所讨论的分组排序,时间关键的分组流的分组形成分组序列,其中每个分组流的每个分组总是占据相同的位置,这可以被解释为分组具有专用的时隙。将被理解的是,时隙是“虚拟”实体,本质上描述来自不同分组流的分组之间的特定顺序。在图8所示的示例中,时隙TS1 属于RRU2,等等。

[0110] 应该被注意的是,根据通信量情况,分组不一定需要在通信网络内建立序列。例如,在图8的场景中,时隙TS4留空,以表示RRU4的“黑色”分组采用不同的出口端口p33。这可能会导致传出分组序列中的空位。如果需要,这种空位可以通过改变分组流到时隙的映射来消除(例如,通过改变到达入口端口p32的分组序列中的分组的顺序,使得RRU4的“黑色”分组是序列中的最后分组)。

[0111] 如从上述示例性实施例的描述已经变得显而易见的,此处提供的公开可以用于通过定义各个分组流分组之间的相对顺序来使相同的通信量处理类的分组流的PDV最小化。如果需要,分组流可以使用入口延迟机制而映射到时隙。就相关的分组流而言,时隙的使用可有助于创建具有一致的分组顺序的分组序列。

[0112] 尽管已经关于示例性实施例描述了本公开,但是应该理解,本公开仅是说明性的。因此,旨在本发明仅由所附权利要求的范围来限制。

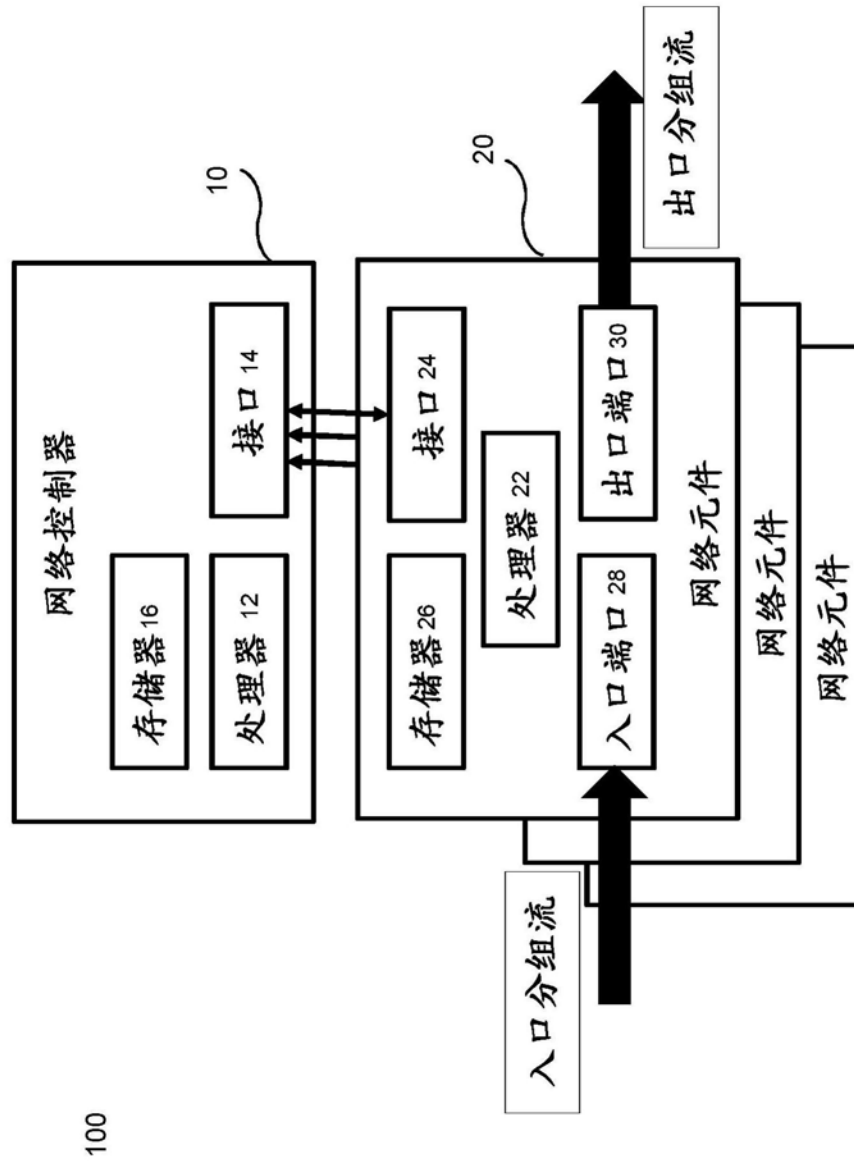


图1

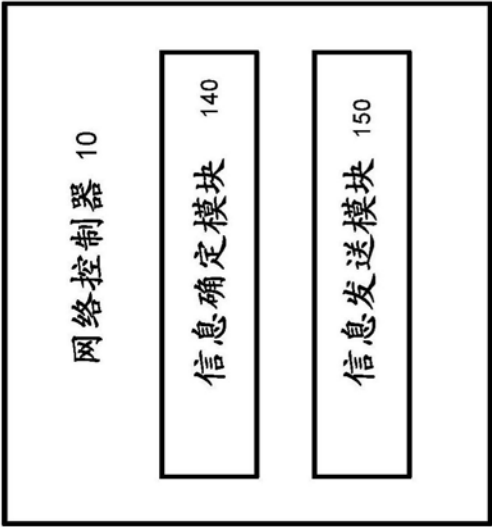


图2A

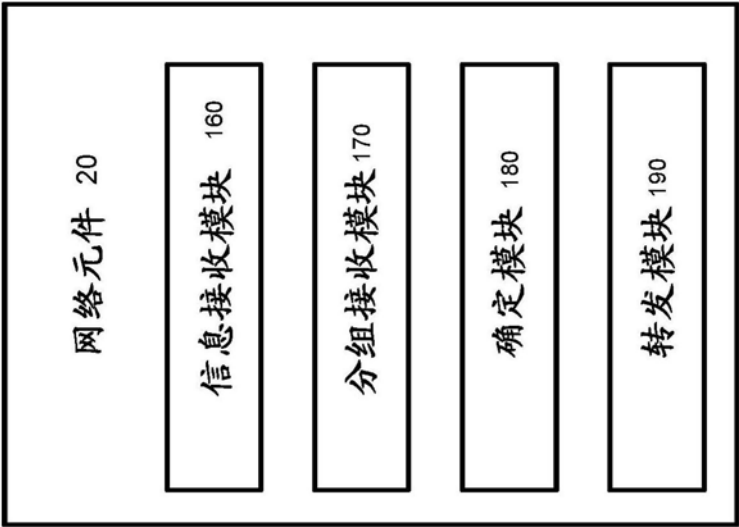


图2B

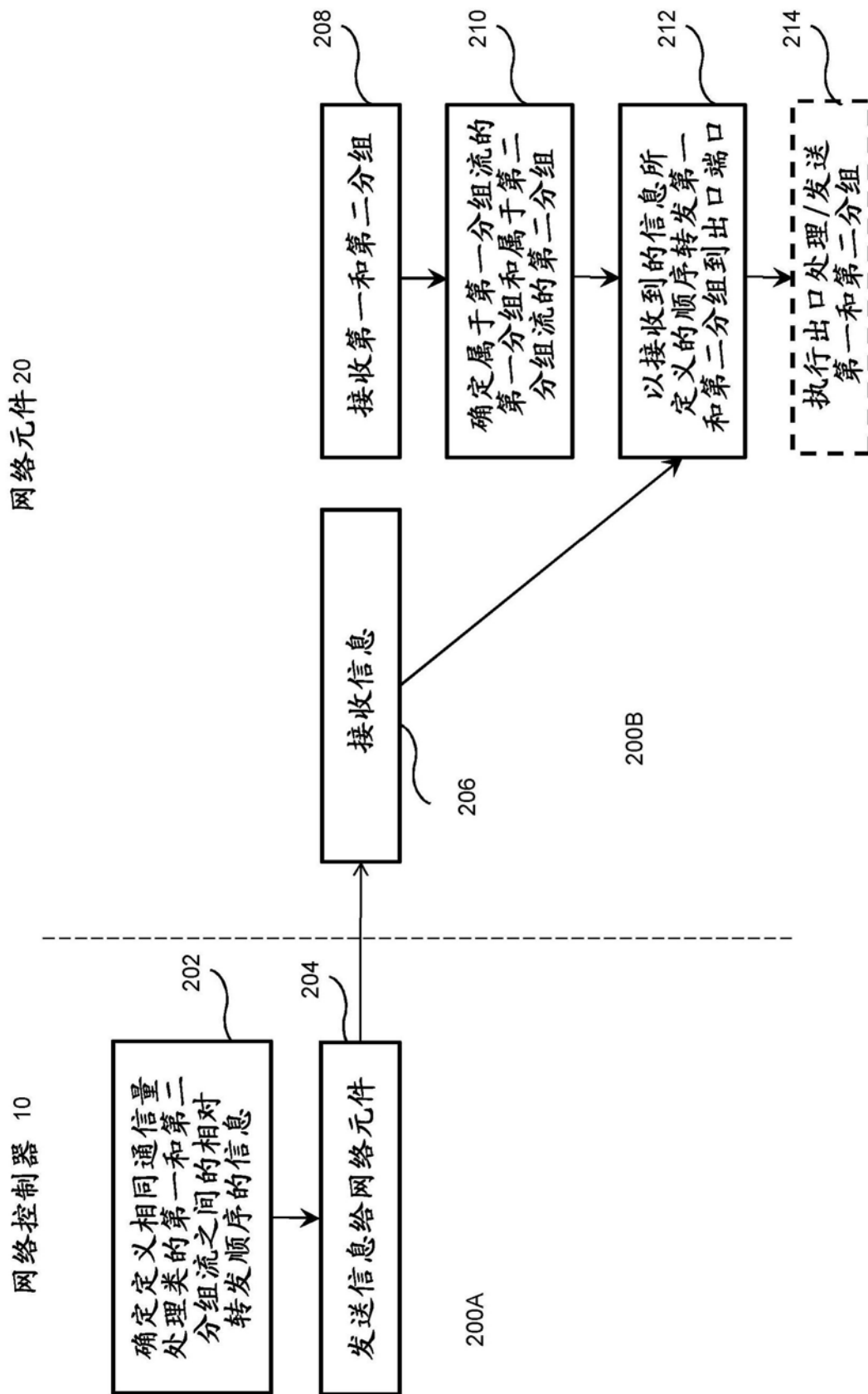


图2C

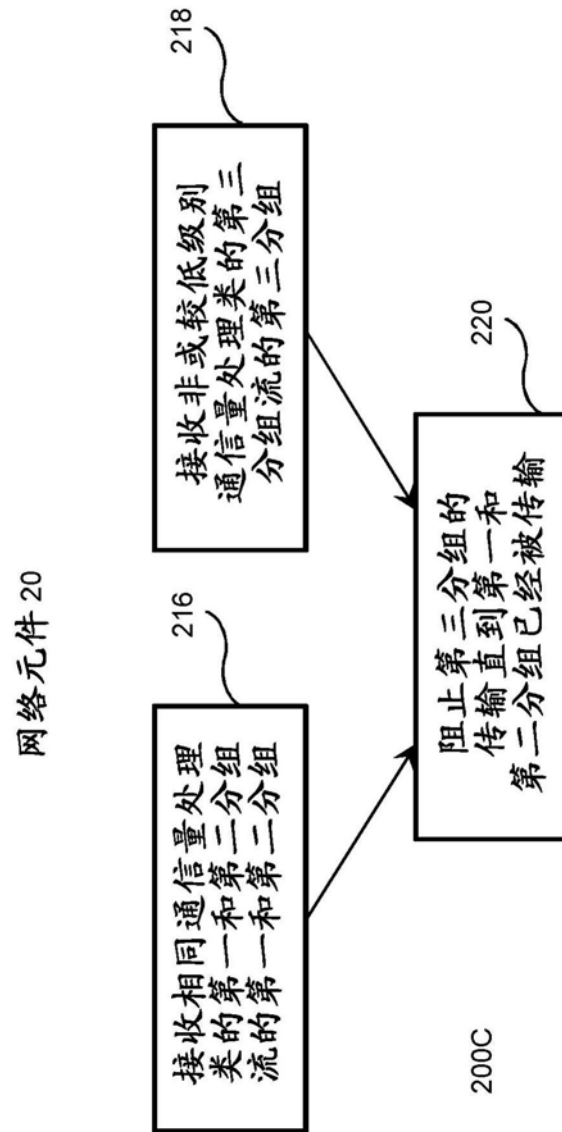


图2D

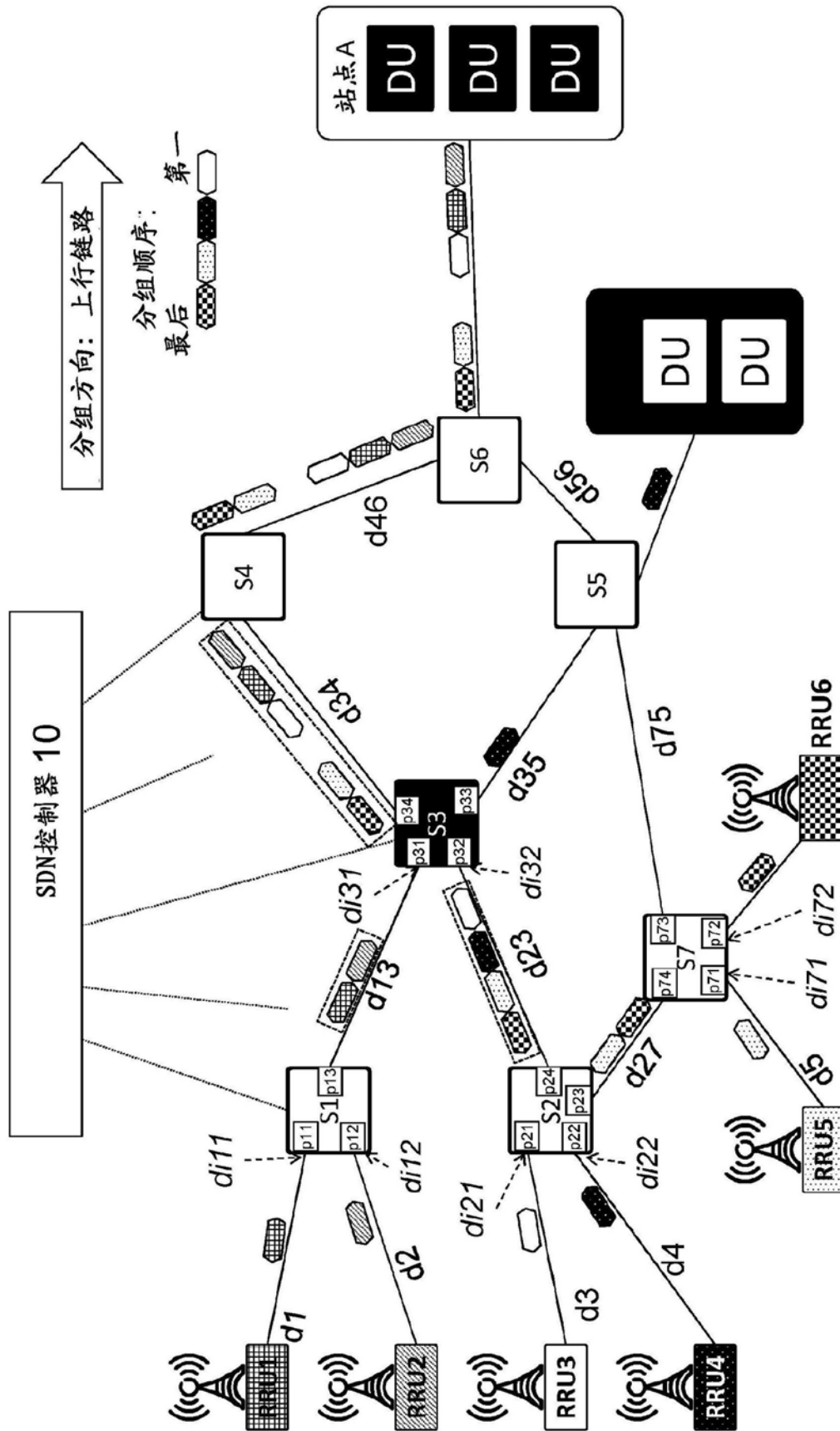


图3

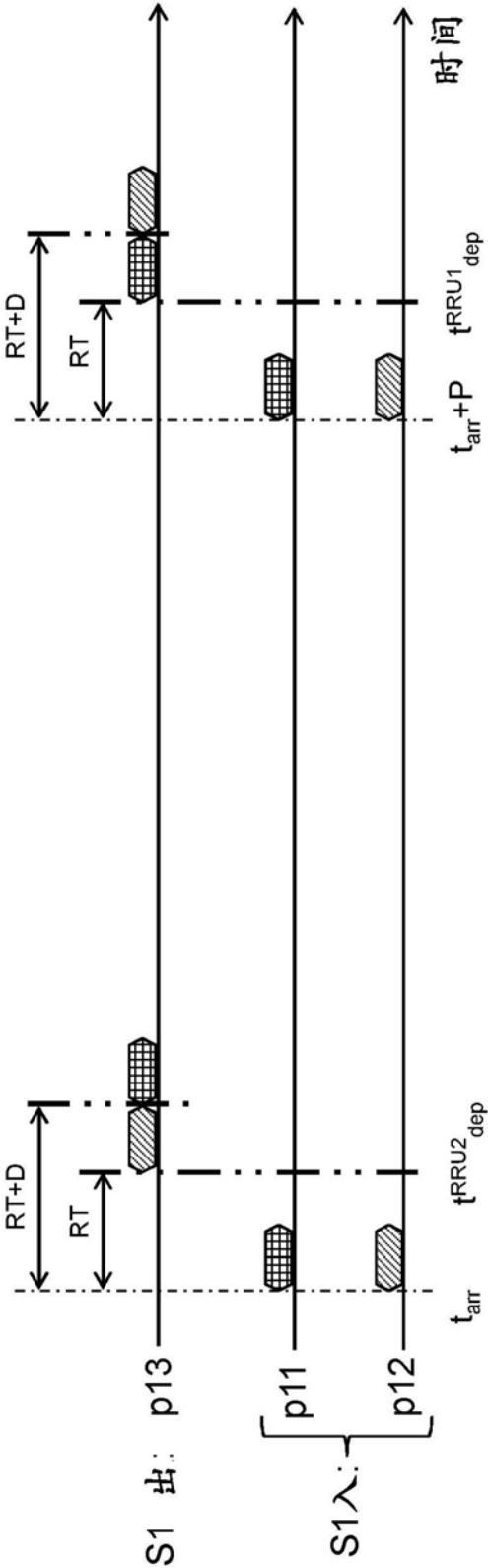


图4

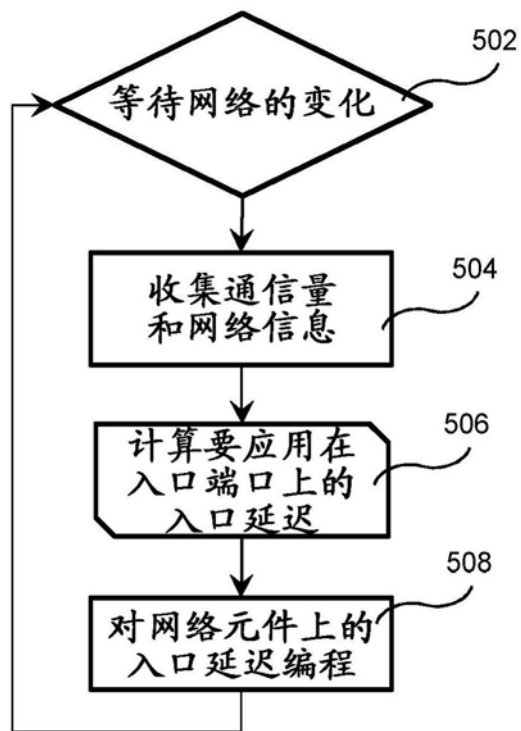


图5

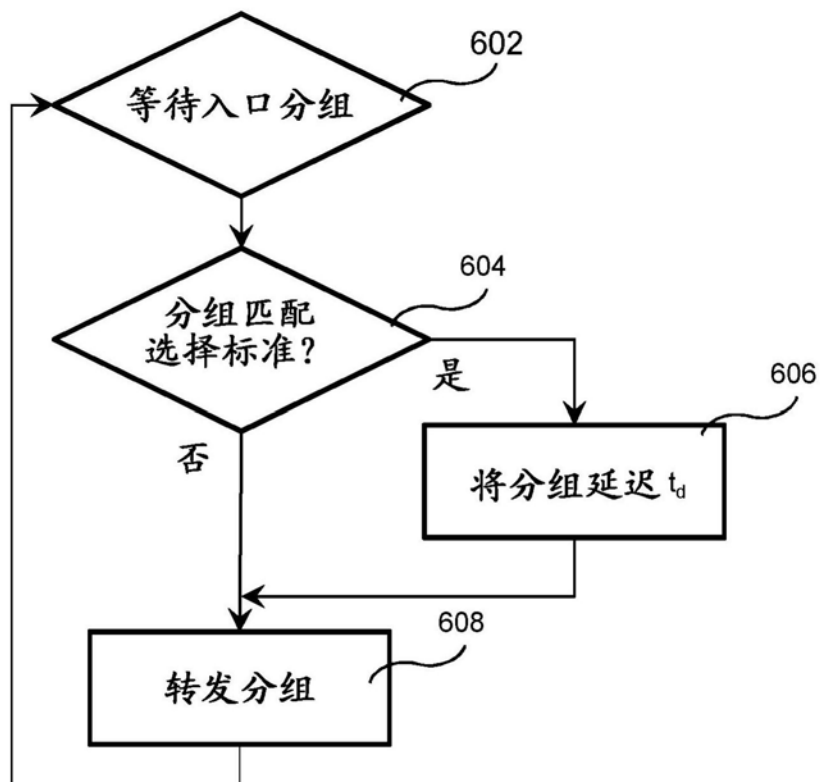


图6

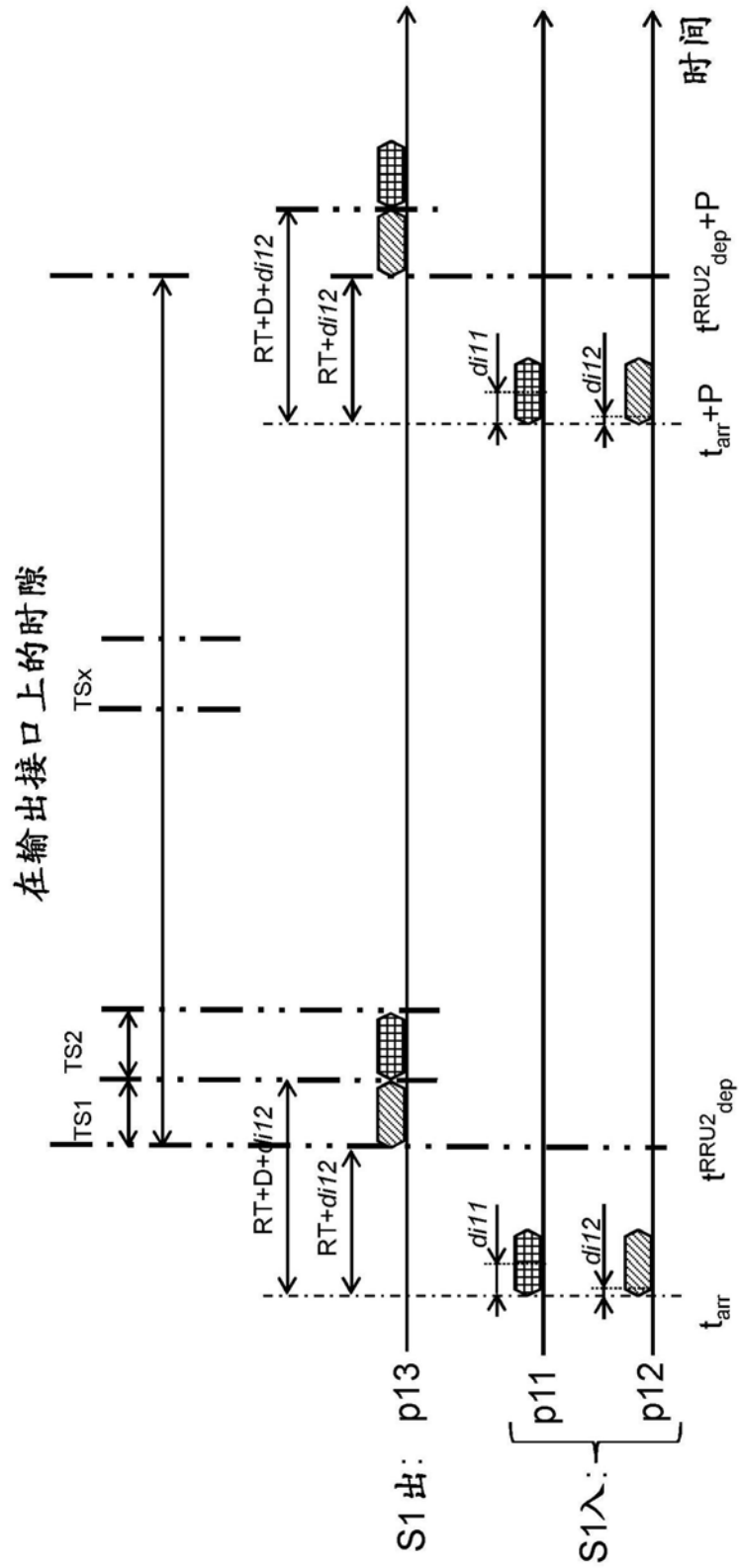


图7

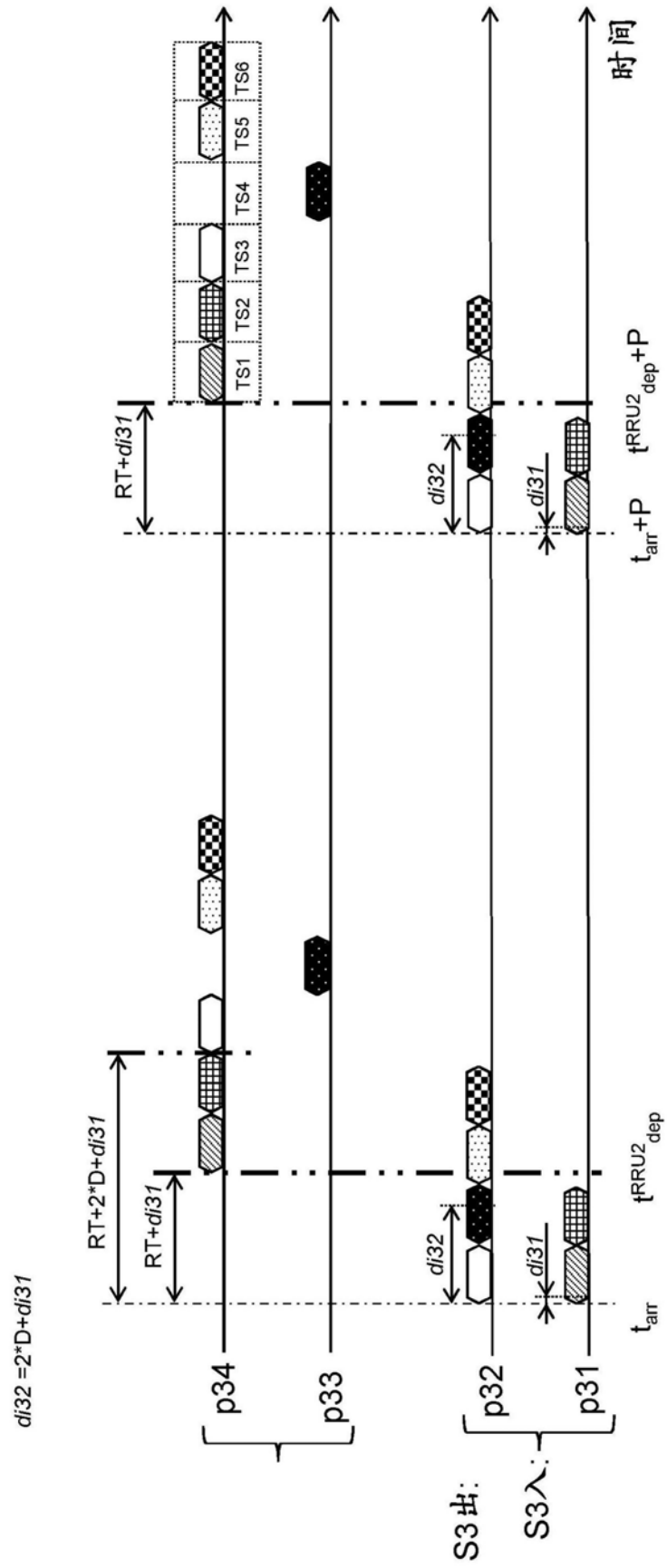


图8