

[12] 发明专利申请公开说明书

[21] 申请号 99810536.8

[43] 公开日 2001 年 10 月 10 日

[11] 公开号 CN 1317189A

[22] 申请日 1999.7.1 [21] 申请号 99810536.8

[30] 优先权

[32] 1998.7.2 [33] US [31] 09/108,771

[86] 国际申请 PCT/US99/15028 1999.7.1

[87] 国际公布 WO00/02347 英 2000.1.13

[85] 进入国家阶段日期 2001.2.28

[71] 申请人 铁桥网络股份有限公司

地址 美国马萨诸塞州

[72] 发明人 S·J·施瓦茨 J·D·卡尔森

Y·佩杜艾尔

M·哈撒韦

[74] 专利代理机构 上海专利商标事务所

代理人 赵国华

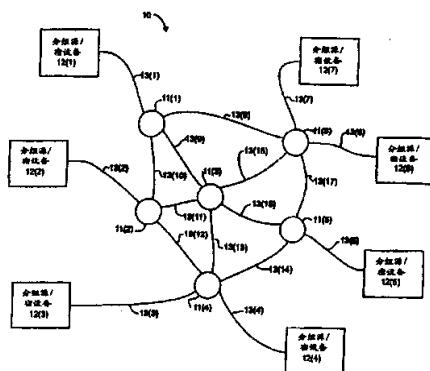
权利要求书 9 页 说明书 27 页 附图页数 8 页

[54] 发明名称 网络分组交换系统和方法

[57] 摘要

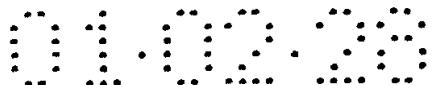
一种用于在网络中传送每一个均包含宿地址的分组的交换节点，包括多个输入端口模块、多个输出端口模块以及一包含分组元数据处理器和分组交换机在内的交换网路。每一输入端口模块与一通信链路连接用于通过其接收分组。每一输入端口模块一旦接收到一分组，便缓存该分组并生成一元数据分组，从而识别将要发送该分组和分组识别符信息的输出端口模块，并将该元数据分组提供给分组元数据处理器。该分组元数据处理器接收全部输入端口模块所生成的元数据分组和全部输出端口模块的工作状态信息，并对每一输出端口模块，结合其工作状态信息处理从全部输入端口模块接收到的元数据分组，来判定应传递还是丢掉该分组。若分组元数据处理器判定与元数据分组相关联的分组将被丢掉的话，便通知其中缓存该分组的输入端口模块，其进而将舍弃该分组。反之，若该分组元数据处理器判定与该元数据分组相关联的该分组不会被丢掉的话，就会针对相关联的输出端口模块对该元数据分组进行排队。每一输出端口模块从分组元数据处理器为此保存的其相应的元数据分组队列当中检索出元数据分组。对输出端口模块

检索出的每一元数据分组，该输出端口模块都会请求该元数据分组中标识的输入端口模块将该输入端口模块中所标识的分组经过该分组交换机传送。当该输出端口模块接收到该分组时，便通过与其连接的通信链路发送它。



权 利 要 求 书

1. 一种网络上传送数据的交换节点，其特征在于，该交换节点包括：
 - 一用于从该网络接收并缓存数据分组的输入端口模块，该输入端口模块生成与一所要传送的接收到的分组相关联的分组元数据，该分组元数据包括识别从该网络接收到该数据分组的该输入端口模块的信息；
 - 一用于从该输入端口接收该分组元数据的分组元数据处理器；
 - 一用于将一数据分组发送到该网络上的输出端口模块，该输出端口模块（i）从该分组元数据处理器接收该分组元数据，（ii）对该分组元数据进行排队，（iii）读出经过排队的分组元数据，以及（iv）向输入端口模块发送一请求信号来启动从该输入端口模块对与该分组元数据相关联的数据分组的发送；以及
 - 一在该输入端口模块和输出端口模块间用于将数据分组从该输入端口模块传送至该输出端口模块的交换模块。
2. 如权利要求1所述的交换节点，其特征在于，交换模块包括多个用于对经过该交换模块传送的数据分组中相应的多个部分进行传送的交换机构。
3. 如权利要求2所述的交换节点，其特征在于，多个交换机构按循环轮流方式传送数据分组的多个部分。
4. 如权利要求1所述的交换节点，其特征在于，输出端口模块向元数据处理器提供有关该输出端口模块工作状态的信息。
5. 如权利要求4所述的交换节点，其特征在于，元数据处理器利用有关输出端口模块工作状态的信息，来判定分组元数据是否将传送至该输出端口模块。
6. 如权利要求4所述的交换节点，其特征在于，有关输出端口模块工作状态的信息包括有关从输入端口模块接收数据分组的该输出端口模块可供利用度的信息。
7. 如权利要求6所述的交换节点，其特征在于，若有关从输入端口模块接收数



据分组的输出端口模块可供利用度的信息表明该输出端口模块可供利用，元数据处理器便将分组元数据传送至该输出端口模块。

8. 如权利要求6所述的交换节点，其特征在于，若有关从输入端口模块接收数据分组的输出端口模块可供利用度的信息表明该输出端口模块无法利用，便丢掉数据分组。

9. 如权利要求4所述的交换节点，其特征在于，有关输出端口模块工作状态的信息包括有关在该输出端口模块处数据拥塞的信息。

10. 如权利要求1所述的交换节点，其特征在于，输入端口模块包括一用于缓存从网络接收到的数据分组的分组存储器。

11. 如权利要求1所述的交换节点，其特征在于，输入端口模块包括多个用于从网络接收数据分组的输入端口。

12. 如权利要求1所述的交换节点，其特征在于，输出端口模块包括多个用于将数据分组发送至网络上的输出端口。

13. 如权利要求1所述的交换节点，其特征在于，交换模块可在多个输入端口模块和多个输出端口模块之间传送数据分组。

14. 一种网络上传送数据的方法，其特征在于，包括：

在一输入端口模块处，(i) 从该网络接收数据分组，(ii) 缓存从该网络接收到的数据分组，以及(iii) 生成与一所要传送的接收到的分组相关联的分组元数据，该分组元数据包括识别从该网络接收到该数据分组的该输入端口模块的信息；提供一用于从该输入端口模块接收该分组元数据的分组元数据处理器；

在一用于将数据分组发送到该网络上的输出端口模块处，(i) 从该分组元数据处理器接收该分组元数据，(ii) 对该分组元数据进行排队，(iii) 读出该经过排队的分组元数据，以及(iv) 向输入端口模块发送一请求信号来启动从该输入端口模块对与该分组元数据相关联的数据分组的发送；以及

在该输入端口模块和输出端口模块间提供一用于将数据分组从该输入端口模块传送至该输出端口模块的交换模块。

15. 如权利要求14所述的方法，其特征在于，提供交换模块包括提供多个用于对经过该交换模块传送的数据分组中相应的多个部分进行传送的交换机构。

16. 如权利要求15所述的方法，其特征在于，多个交换机构按循环轮流方式传送数据分组的多个部分。

17. 如权利要求14所述的方法，其特征在于，输出端口模块向元数据处理器提供有关该输出端口模块工作状态的信息。

18. 如权利要求17所述的方法，其特征在于，元数据处理器利用有关输出端口模块工作状态的信息，来判定分组元数据是否将传送至该输出端口模块。

19. 如权利要求17所述的方法，其特征在于，有关输出端口模块工作状态的信息包括有关从输入端口模块接收数据分组的该输出端口模块可供利用度的信息。

20. 如权利要求19所述的方法，其特征在于，若有关从输入端口模块接收数据分组的输出端口模块可供利用度的信息表明该输出端口模块可供利用，元数据处理器便将分组元数据传送至该输出端口模块。

21. 如权利要求19所述的方法，其特征在于，若有关从输入端口模块接收数据分组的输出端口模块可供利用度的信息表明该输出端口模块无法利用，便丢掉数据分组。

22. 如权利要求17所述的方法，其特征在于，有关输出端口模块工作状态的信息包括有关在该输出端口模块处数据拥塞的信息。

23. 如权利要求14所述的方法，其特征在于，输入端口模块包括一用于缓存从网络接收到的数据分组的分组存储器。

24. 如权利要求14所述的方法，其特征在于，输入端口模块包括多个用于从网络接收数据分组的输入端口。

25. 如权利要求14所述的方法，其特征在于，输出端口模块包括多个用于将数据分组发送至网络上的输出端口。

26. 如权利要求14所述的方法，其特征在于，交换模块可在多个输入端口模块和输出端口模块之间传送数据分组。

27. 一种通过网络上交换节点递送数据分组的方法，其特征在于，所述数据分组包括一该数据分组将要沿该交换节点出发的输出路径递送至的、识别该网络上一宿节点的宿地址，该宿地址可分为具有相应的多个分地址的多个分地址字段，所述方法包括：

在存储器中存储一将多个宿地址中的每一个与一交换节点出发的输出路径相关联的数据表，通过其可将分组递送至该宿地址所标识的该宿节点，该数据表中多个部分中的每一部分与相应的分地址字段相关联，并可通过利用该相关的分地址字段所具有的分地址来访问；

提供一系列分地址处理器，每一个分地址处理器与相应的分地址字段相关联，接收其相关联的相应分地址字段所具有的分地址，并利用该分地址来访问与该分地址字段相关联的数据表部分以接收有关与该宿地址相关联的输出路径的信息；

从将要通过交换节点递送的相应系列的数据分组当中接收一系列宿地址；

利用该分地址处理器系列按流水线方式访问该数据表，就每一接收到的数据分组检索一输出路径，这样，

在第一寻址期间，与第一分地址字段相关联的第一分地址处理器，利用第一接收到的宿地址的第一分地址字段所具有的分地址来访问与第一分地址字段相关联的数据表的第一部分；以及

在第二寻址期间，(i) 与第二分地址字段相关联的第二分地址处理器，利用第一宿地址的第二分地址字段所具有的分地址来访问与第二分地址字段相关联的数据表的第二部分，以及(ii) 第一分地址处理器利用第二接收到的宿地址的第一分地址字段所具有的分地址来访问与第一分地址字段相关联的数据表的第一部分。

28. 如权利要求27所述的方法，其特征在于，在寻址期间，第一分地址处理器向第二分地址处理器提供一信息项，由该第二分地址处理器利用该信息项访问数据表的第二部分。

29. 如权利要求28所述的方法，其特征在于，信息项是在数据表中的由第二分地址处理器用来访问该数据表中第二部分的一起始地址。

30. 如权利要求29所述的方法，其特征在于，
第二分地址字段所具有的分地址是一地址偏移量；以及
第二分地址处理器将起始地址和地址偏移量组合在一起访问该数据表的第二部分。

31. 如权利要求27所述的方法，其特征在于，该数据表是一分组路由选择数据表。

32. 如权利要求27所述的方法，其特征在于，该数据表是一与互联网协议兼容的分组路由选择数据表。

33. 一种通过网络上交换节点递送数据分组的装置，其特征在于，所述数据分组包括一该数据分组将要沿该交换节点出发的输出路径递送至的、识别该网络上一宿节点的宿地址，该宿地址可分为具有相应的多个分地址的多个分地址字段，所述装置包括：

一存储器，用于存储一将多个宿地址中的每一个与一交换节点出发的输出路径相关联的数据表，通过其可将分组递送至该宿地址所标识的该宿节点，该数据表中多个部分中的每一部分与相应的分地址字段相关联，并可通过该相关的分地址字段所具有的分地址来访问；

一系列分地址处理器，每一个分地址处理器与相应的分地址字段相关联，接收其相关联的相应分地址字段所具有的分地址，并利用该分地址来访问与该分地址字段相关联的数据表部分来接收有关与该宿地址相关联的输出路径的信息；

一接收机，用于从将要通过交换节点递送的一相应系列的数据分组当中接收一

系列宿地址；其中

分地址处理器系列按流水线方式访问该数据表，就每一接收到的数据分组检索一输出路径，这样，

在第一寻址期间，与第一分地址字段相关联的第一分地址处理器，利用第一接收到的宿地址的第一分地址字段所具有的分地址来访问与第一分地址字段相关联的数据表的第一部分；以及

在第二寻址期间，(i) 与第二分地址字段相关联的第二分地址处理器，利用第一宿地址的第二分地址字段所具有的分地址来访问与第二分地址字段相关联的数据表的第二部分，以及(ii) 第一分地址处理器利用第二接收到的宿地址的第一分地址字段所具有的分地址来访问与第一分地址字段相关联的数据表的第一部分。

34. 如权利要求33所述的装置，其特征在于，在寻址期间，第一分地址处理器向第二分地址处理器提供一信息项，由该第二分地址处理器利用该信息项访问数据表的第二部分。

35. 如权利要求34所述的装置，其特征在于，信息项是在数据表中的由第二分地址处理器用来访问该数据表中第二部分的一起始地址。

36. 如权利要求35所述的装置，其特征在于，

第二分地址字段所具有的分地址是一地址偏移量；以及

第二分地址处理器将起始地址和地址偏移量组合在一起访问该数据表的第二部分。

37. 如权利要求33所述的装置，其特征在于，该数据表是一分组路由选择数据表。

38. 如权利要求33所述的装置，其特征在于，该数据表是一与互联网协议兼容的分组路由选择数据表。

39. 一种网络上从输入模块至输出模块的数据传送的控制方法，其特征在于，所述方法包括：



提供一控制模块用于控制该输入模块和输出模块之间的通信；

在输入模块处接收一数据分组，该输入模块响应数据分组生成一元数据分组，该元数据分组包括对将要向其传送该数据分组的输出模块的识别信息，该输入模块将该元数据分组传送给控制模块；以及

在控制模块处，(i) 从输入模块接收该元数据分组，(ii) 从所识别的输出模块接收表明该所识别的输出模块的状态的第一信息项，以及(iii) 根据该第一信息项判定该数据分组是否将传送至该识别的输出模块。

40. 如权利要求39所述的方法，其特征在于，还包括：将输入模块经过一交换模块耦合至输出模块，从而通过该交换模块将数据分组从输入模块传送至输出模块。

41. 如权利要求39所述的方法，其特征在于，第一信息项表明数据在所识别的输出模块处拥塞。

42. 如权利要求39所述的方法，其特征在于，第一信息项表明所识别的输出模块接收数据分组的可供利用度。

43. 如权利要求39所述的方法，其特征在于，还包括：将第二信息项从控制模块传送给输入模块，所述第二信息项表明数据分组将不传送给所识别的输出模块。

44. 如权利要求39所述的方法，其特征在于，还包括：将数据分组存储在至少一个与输入模块相关联的缓存器中。

45. 如权利要求44所述的方法，其特征在于，还包括：响应第二信息项，从控制模块向输入模块传送一给该输入模块的命令，令该输入模块对存储数据分组的至少一个缓存器进行清零，而不向所识别的输出模块传送数据分组。

46. 如权利要求44所述的方法，其特征在于，还包括：响应第二信息项，从控制模块向输入模块传送一给该输入模块的命令，令该输入模块对存储数据分组的至少一个缓存器进行复位，而不向所识别的输出模块传送数据分组。

47. 一种网络上传送数据的装置，其特征在于，所述装置包括：

一用于从该网络接收数据分组的输入模块；

一用于从该输入模块接收数据分组并将数据分组传送到该网络上的输出模块；

以及

一用于控制该输入模块和输出模块间通信的控制模块；其中

该输入模块适合（i）响应所接收到的数据分组生成一元数据分组，该元数据分组包括对将要向其传送该数据分组的输出模块的识别信息，以及（ii）将该元数据分组传送给该控制模块；以及

该控制模块适合（i）从该输入模块接收该元数据分组，（ii）从所识别的输出模块接收表明该识别的输出模块的状态的第一信息项，以及（iii）根据该第一信息项判定该数据分组是否将传送至该识别的输出模块。

48. 如权利要求47所述的装置，其特征在于，还包括：一通过其将数据分组从输入模块传送至输出模块的交换模块。

49. 如权利要求47所述的装置，其特征在于，第一信息项表明数据在所识别的输出模块处拥塞。

50. 如权利要求47所述的装置，其特征在于，第一信息项表明所识别的输出模块接收数据分组的可供利用度。

51. 如权利要求47所述的装置，其特征在于，控制装置适合将第二信息项从控制模块传送给输入模块，所述第二信息项表明数据分组将不传送给所识别的输出模块。

52. 如权利要求47所述的装置，其特征在于，输入模块与至少一个用于存储所接收数据分组的缓存器相关联。

53. 如权利要求52所述的装置，其特征在于，控制模块适合响应第二信息项，向输入模块传送一给该输入模块的命令，令该输入模块对存储数据分组的至少一个

01·02·08

缓存器进行清零，而不向所识别的输出模块传送数据分组。

54. 如权利要求52所述的装置，其特征在于，控制模块适合响应第二信息项，从该控制模块向输入模块传送一给该输入模块的命令，令该输入模块对存储数据分组的至少一个缓存器进行清零，而不向所识别的输出模块传送数据分组。

网络分组交换系统和方法

发明领域

本发明总体涉及数字通信领域，具体来说，涉及一种数字数据网络中所用的交换节点对数字数据进行分组交换的系统和方法。

发明背景

已经开发出种种数字网络，来方便包括数据和程序在内的信息在数字计算机系统和许多其他类型设备之间传送。已开发实施了采用多样的数据传送法的多种类型的网络。现代网络中，经过通信链路按多种模式互联的交换节点网传送信息。该互联成网的模式，允许可从作为源设备发送信息的各个计算机系统或其他设备至将作为宿设备接收信息的另一计算机系统或其他设备间存在若干个可经该网络提供的路径，这样一旦特定区域的网络发生拥塞或网络部件失去作用，便可使信息绕过网络中拥塞或失去作用的部分。

从源设备传送至宿设备的信息总体上按固定或可变长度的分组形式传送，一交换节点通过与其连接的一条通信链路接收这些分组，并通过另一通信链路发送这些分组，以便该分组沿一至宿设备的路径传送到宿设备或另一交换节点。每一分组通常包括地址信息，其中包括识别生成该分组的特定设备的源地址和识别接收该分组的各特定设备的宿地址。

通常，一交换节点包括一个或多个输入端口、多个输出端口和一“交换网路”，各个输入端口与一通信链路连接来接收各分组，各个输出端口则与一通信链路连接来发送各分组，“交换网路”将各输入端口的各分组耦合至相应的用于传输的输出端口。一输入端口接收到一分组后，通常会缓存该分组，从宿地址当中识别出将发送该分组的那个特定输出端口，并经交换网路将该分组传送至该输出端口。输出端口接收到该分组后，其（即输出端口）通常会将该分组缓存在一队列中，用于通过与之连接的通信链路进行传输。虽然输出端口的缓存和调度可便于输出端口提供高效的分组传输，但由于输出端口可能保持连续的繁忙状态，因而会随输出端口的缓存出现几个问题。通常，各个输出端口会有效地对各个输入端口提供一个队

列，在这种情况下，该交换节点所提供的队列总数将为 N^2 量级，其中N为输入端口数目，若如常规那样每一通信链路用于分组的双向传输，N便进而与输出端口数目相对应。这样，随着输入/输出端口数目N的增加，该输出端口所维持的队列其数目便按平方的速率急剧增加，从而输出的排队无法较好地成比例增加。

不同于对所要发送的分组使用输出排队，已开发了可提供输入排队的种种交换节点，其中在输入端口处对分组进行缓存和排队。每一输入端口仅需要一个队列，故随着输入（输出）端口数目的增加，队列数按一线性斜率增加，避免了随输出排队而来的成平方增加。但输入排队造成交换网路低得多的使用效率，这是因为将所接收的分组缓存后，输入端口必须进行利用交换网路所必需的竞争和仲裁，以便分组传送至相应的用于传输的输出端口。

发明概述

本发明提供一种新改进的交换节点，用于提供对互联输入和输出端口的交换网路的高效率利用，其特征在于，交换节点提供对该交换节点所传送的分组进行输出排队，同时避免分组队列相对于输入/输出端口数目的增加而成平方增加，其特征在于提供进行输出排队。同样，本发明提供一种新改进的交换节点，用于相对于输入/输出端口数目的增加使分组队列线性增加，其特征在于，交换节点提供进行输入排队，同时避免对互联输入和输出端口的交换网路进行的相对来说不够有效的利用，其特征在于，交换节点提供对该交换节点所传送分组进行输入排队。

简要归纳一下，本发明提供的交换节点，包括在网络中进行分组传送的多个输入端口模块、多个输出端口模块和一交换网路，每一分组包括宿地址信息。每一输入端口模块与一通信链路连接用于通过其接收分组，每一输出端口模块则与一通信链路连接用于通过其发送分组。每一输入端口模块一旦接收到与其连接的通信链路来的分组，便缓存该分组并生成用于其的元数据分组，该元数据分组标识将发送该分组的输出端口模块，并生成该分组的识别符信息，具体来说，生成对其中缓存该分组的输入端口模块的标识，以及输入端口模块中缓存该分组的位置的指针。生成元数据分组之后，输入端口模块将它提供给交换网路，具体来说，提供给其分组元数据处理器部分。

交换网路包括分组元数据处理器部分和分组交换机部分这两者。分组元数据处理器部分接收全部输入端口模块生成的元数据分组和全部输出端口模块的工作状态信息。每一输出端口模块的工作状态信息，包括对各个相应的输出端口模块来说有

益于就那些将会由相应输出端口模块发送的分组是否被忽略或被丢弃这一情况形成一判断的信息。对每一输出端口模块来说，分组元数据处理器与工作状态信息相联系来处理从全部输入端口模块接收到的元数据分组。若元数据分组处理过程中分组元数据处理器判定与元数据分组相关联的分组将被丢弃，便会通知其中缓存该分组的输入端口模块，它因而会忽略该分组。另一方面，若该分组元数据处理器判定与元数据分组相关联的分组未被丢弃，便对该相关联的输出端口模块的元数据分组进行排队。

每一输出端口模块根据分组元数据处理器为此保存的其相应的元数据分组队列检索元数据分组。对于一输出端口模块检索出的每一元数据分组，输出端口模块都会向元数据分组中标识的输入端口模块发出一请求，要求将输入端口模块中标识的分组传送给它（即传送给发出该请求的输出端口模块）。输入端口模块接收到该输出端口模块的请求后，便会经过该交换网路的分组交换机部分传送该请求所要求的分组。分组交换机部分进而会将该分组传送至输出端口模块，由此通过与其连接的通信链路进行传输。

按照本发明构成的交换节点提供一输入排队的交换节点的伸缩性，同时基本上保持一输出排队的交换节点的交换网路利用效率。由于输入端口模块处缓存该分组，直到分组元数据处理器判定它们将由相应的输出端口模块发送为止，进而到它们被相应的输出端口模块所请求发送到此进行传输为止，所以仅需要“N”个缓存或队列，每一输入端口模块有一个，而输出排队的交换节点中则需要有 N^2 个缓存。但对于为此建立的队列中的每一输出端口模块来说，是很有效地分开对哪些分组应忽略（即哪些分组应丢弃）和哪些不应忽略（即哪些分组应通过）这些情况作出判断的，从而对于一输出排队的交换节点来说，是很有效地按与那相类似方式作出通过/丢弃判断的。这样，该交换网路的分组交换机部分所实现的效率与输出排队的交换节点所实现的相类似。

附图简要说明

所附权利要求中针对的是具有特殊性的本发明。可参照下面结合附图给出的说明更好地理解本发明上述以及进一步的优点。其中，

图1示意性示出一包括至少一个按照本发明构成的交换节点的计算机网络；

图2是一按照本发明构成的用于图1所示网络的交换节点的功能框图；

图3是图2所示交换节点所用的一输入端口模块的功能框图；



图4是图2所示交换节点所用的端口间分组交换机中一个交换平面的功能框图；

图5是图2所示交换节点中一用于处理分组元数据的元数据分组处理器的功能框图；

图6是图1所示交换节点所用的一输出端口模块的功能框图；

图7是图3所示输入端口模块中有用的帧递送引擎一个实施例的功能框图；以及
图8是图3所示输入端口模块中有用的帧递送引擎第二实施例的功能框图。

示范性实施例的详细说明

图1示意性示出包括多个交换节点11（1）至11（N）（通常由标号11（n）标识）的计算机网络10，用于在若干个设备间传送表示数据的信号，在图1中这些设备由广域网（“WAN”）中的分组源/宿设备12（1）至12（M）（通常由标号12（m）标识）表示。分组源/宿设备12（m）如常规的那样，包括一特定设备，诸如存储、生成、处理或利用数字数据的计算机系统或其他设备，这些设备的局域网等（未分开图示）乃至广域网10。每一分组源/宿设备12（m）经过一通常由标号13（p）标识的通信链路与一交换节点11（n）连接，以便于其对数据的发送和接收。交换节点11（n）经过通常也由标号13（p）所标识的通信链路互联，以便于在相应的交换节点11（n）之间传送信息。通信链路13（p）可利用任何适宜的信息传输介质，例如包括用于载送电信号的导线，用于载送光信号的光纤链路等。每一通信链路13（p）最好是双向的，允许交换节点11（n）互相间和与经过相同链路同其连接的用户住宅设备12（m）间发送接收信号；根据为相应通信链路13（p）选定的特定介质类型，可提供多介质用于在相反方向上传送信号，由此提供双向链路。

网络10中数据以分组形式传送。通常，一分组包括报头部分和数据部分。报头部分包括辅助经过网络对分组选择路由的信息，以及取决于经过该网络对分组选择路由过程中所用的对协议选择路由的特定分组的信息。与网络10相联系，可以采用若干个公知的分组路由选择协议中的任何协议；一个实施例中采用了众所周知的互联网协议（“IP”）。在任何情况下，报头通常包括地址信息，其中包含对生成分组的特定源设备12（_{m_s}）进行识别的源地址和对要接收该分组的特定宿设备12（_{m_d}）进行识别的宿地址。IP协议中，分组可以是可变长度的，报头通常还会包括长度信息以识别该分组长度。报头通常还包括其他信息，例如包括对规定分组结构的特定协议进行识别的协议识别符信息。数据部分包含分组的数据有效负荷。分组还可以包括作为数据部分等其中一部分的检错信息，可用于确定在分组传送过程中是否发

生过差错。

源设备12 (m_s) 在生成一传送给宿设备12 (m_b) 的分组后，将向与其连接的交换节点11 (n) 提供分组。交换节点11 (n) 将在分组中利用宿地址，来尝试识别一将宿地址同要经过其传送分组、与交换节点连接的其中一个通信链路13 (p) 相关联的“路由”，将其（即分组）递送至宿设备12 (m_b) （若交换节点11 (n) 与宿设备12 (m_b) 连接），或送至沿一路径至宿设备12 (m_b) 的另一交换节点11 (n') ($n' \neq n$)。若交换节点可对所接收到的分组识别一路由，便经过该路由所识别的通信链路递送该分组。接收该分组的每一交换节点11 (n)，11 (n")，…都会执行一类似操作。若全部交换节点均具有面向宿地址的相应路由，该分组便会最终到达宿设备12 (m_b)。

本发明提供一种新交换节点11 (n)，图2示出其功能框图，提供经过网络对分组的高效传送。参照图2，交换节点11 (n) 包括：由包含端口间分组交换机22、分组元数据处理器23和交换节点管理处理器27在内的交换网路互联的若干个输入端口模块20 (1) 至20 (N) (通常由标号20 (n) 标识) 和同样个数的输出端口模块21 (1) 至21 (N) (通常由标号21 (n) 标识)。每一输入端口模块20 (n) 包括与相应的一个通信链路13 (p) 连接用于经过其接收 (n) (m) 信号中PKT_IN (n) (m) 分组所表示分组的一个或多个输入端口25 (n) (1) 至25 (n) (M) (通常由标号25 (n) (m) 标识)。对于所接收的每一分组，输入端口模块20 (n) 缓存该分组，并在后面述及的识别处理中从该分组报头所包含的宿地址当中识别用于其的合适路由，识别处理中识别的是该分组将要发送给的特定输出端口模块21 (n)，以及该输出端口模块21 (n) 上将要通过其传输该分组以便将该分组递送给宿设备12 (m_b) 或递送给沿该路径的下一交换节点11 (n') 来方便该分组路由选择至宿设备12 (m_b) 的一个或多个输出端口26 (n) (1) 至26 (n) (M) (通常由标号26 (n) (m) 标识) 当中的一个。通常，输入端口模块20 (n) 如INP (n) _PKT_DATA输入 (n) 分组数据信号（索引“n”是1至N中的任一整数）所示，会将分组传送给交换网路，尤其传送给端口间分组交换机22。该端口间交换机22进而将该分组如OP (n) _PKT_DATA输出 (n) 分组数据信号（索引“n”是1至N中的任一整数）所示，耦合至所识别的输出端口模块用于传输。

如上所述，每一输出端口模块21 (n) 包括一个或多个输出端口26 (n) (m)，其中每一个与一个或多个通信链路13 (p) 连接。该输出端口模块21 (n) 进而如PKT_OUT分组输出信号所示，将交换网路提供给其的相应的某些分组通过通信链路13

(p) 发送。

将来自输入端口模块20 (n) 的分组耦合至相应输出端口模块21 (n) 的端口间分组交换机22是交叉点接线器形式的。更为具体地来说，该端口间分组交换机22是多个交换平面22 (1) 至22 (P) (通常由标号22 (p) 标识) 形式的，每一平面是交叉点接线器形式的。通常，每一输入端口模块20 (n) 当其将一分组传送给端口间分组交换机22时，便将该分组分成按一循环轮流方式传送给端口间分组交换机22中连续的交换平面22 (m) 的一系列分段。同样，每一输出端口模块21 (n) 当其从端口间分组交换机接收一分组时，便会按一循环轮流方式接收来自连续的交换平面22 (m) 的连续分组分段，并会在经过其相应的输出端口26 (n) (m) 传送之前将这些分段重组成为一分组。在端口间分组交换机22中提供多交换平面22 (p)，允许分组从输入端口模块20 (n) 至输出端口模块21 (n) 有更高的吞吐量。另外，若交换平面22 (p) 故障或另外失去作用，可从循环轮流圈中剔出，端口间分组交换机22便会允许用其他交换平面22 (p') ($p' \neq p$) 来继续进行令人满意的交换。在一个实施例中，对交换平面22 (p) 是否发生过故障或另外失去作用、进而是否该从循环轮流圈中剔出所进行的判断，取决于交换平面22 (p) 将分组分段从输入端口模块20 (n) 传送至输出端口模块21 (n) 过程中的位差错率。

交换节点管理处理器27对交换节点11 (n) 执行若干管理功能，对本领域技术人员来说很清楚，这些管理功能例如包括：对输入端口模块20 (n) 在就相应分组识别将要经过其发送这些分组的合适输出端口模块21 (n) 和输出端口26 (n) (m) 过程中所用的路由信息进行保存和更新。另外，交换节点管理处理器27就相应的交换平面22 (p) 接收位差错率信息，并对其进行响应来控制交换平面22 (p) 切换进入或撤出该循环轮流圈。

按照本发明，与对经过交换网路从每一输入端口模块20 (n) 至相应输出端口模块21 (n) 的分组传送所进行的控制相联系，输入端口模块20 (n) 经过与其连接的通信链路13 (p) 接收一分组后，便会缓存该分组，并生成用于此且说明该分组的元数据分组，包括指向输入端口模块20 (n) 中的分组和路由信息的指针，其中包含将要接收该分组的输出端口模块21 (n) 和将要通过其发送该分组的输出端口26 (n) (m) 的识别符，也许还包括对分组长度的指示。输入端口模块20 (n) 就该分组生成一元数据分组后，将会如INP (n) _META-DATA/RESP输入 (n) 元数据/应答信号所示，将该元数据分组提供给交换网路中的分组元数据处理器23用于处理。

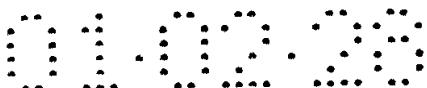
分组元数据处理器23还从相应的输出端口模块21 (n) 接收OP (n) _CTRL

/STATUS输出端口模块21（n）控制/状态信号所示的关于其工作状态的信息，尤其是其在任意时刻从输入端口模块20（n）接收另外的分组用于传输的各自能力的信息。输出端口模块21（n）状态信息可反映若干因数的状况，例如包括：输出端口模块21（n）所具有的可用于传输前从输入端口模块20（n）接收到的分组的缓存量（或反之，当前被所要发送的分组所占据的缓存量）是否可供利用的缓存量正下降、升高或基本不变等状况，所有这些将提供一与相应的元数据分组相关联的输出端口模块其分组接收能力的指示。

从相应的输入端口模块20（n）接收每一分组的元数据分组后，该分组元数据处理器23将会判断与将要发送该分组的输出端口模块21（n）相关联的当前状态信息，是否表明该输出端口模块21（n）具有足够的能力从该输入端口模块20（n）接收该分组，并通过相应的通信链路13（p）发送它（即该分组）。若该分组元数据处理器23作出肯定的判断，即判定与将要发送该分组的输出端口模块21（n）相关联的当前状态信息表明其（即该输出端口模块21（n））具有足够的能力接收并发送该分组的话，该分组元数据处理器23便向相应的输出端口模块21（n）提供OP（n）_CTRL/STATUS信号所示的元数据分组。

该输出端口模块21（n），从同其（即该输出端口模块21（n））将要发送的分组相关联的分组元数据处理器23接收到一元数据分组后，便会因此提供一由OP（n）/INP（n）_PKT_REQ输出（n）/输入（n）分组请求信号（两种情形的索引“n”均为整数，其数值可以不同）所示的分组请求，来启动正缓存该分组以经过该端口间分组交换机22向输出端口模块21（n）传送该分组的输入端口模块20（n）。从该输出端口模块21（n）接收该分组请求后，输入端口模块20（n）便将INP（n）_PKT_DATA输入（n）分组数据信号所示的该分组发送给端口间分组交换机22，其进而将OP（n）_PKT_DATA输出（n）分组数据信号所示的该分组，耦合至该输出端口模块21（n）用于如上所述的传输。

另一方面，若分组元数据处理器23在从一输入端口模块20（n）接收到一分组的元数据分组后，判定将要发送该分组的输出端口模块21（n）其工作状态表明该输出端口模块21（n）其能力使得该输出端口模块21（n）将无法发送该分组的话，其（即分组元数据处理器23）会向接收到其元数据分组的输入端口模块20（n）提供一由INP（n）_META-DATA/RESP输入（n）元数据/应答信号所示的通知。例如，如果与输出端口模块21（n）相关联的状态信息如分组元数据处理器23所保存的那样，表明当输入端口模块20（n）向分组元数据处理器23提供一会另外发送给输出端口模块21

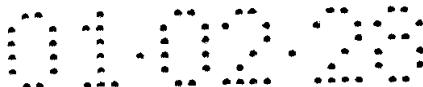


(n) 的分组的元数据分组时该输出端口模块21 (n) 便拥塞的话，这种情况便会发生。在此情况下，由于输出端口模块21 (n) 将无法发送该分组，输入端口模块20 (n) 便可忽略它（即该分组）来释放缓存空间用于会通过与其连接的通信链路接收的额外分组。另外，分组元数据处理器23将忽略与所要忽略的分组相关联的元数据分组，而不将该元数据分组递送给输出端口模块21 (n)。除非输入端口模块20 (n) 从分组元数据处理器23接收到该分组将被忽略的通知，否则该输出端口模块21 (n) 通常将请求其通过其中一条与其连接的通信链路13 (p) 进行传输。

因而，将会理解交换节点11 (n) 内对于通过该网络接收到的分组是否将由交换节点发送这种决定，是分组元数据处理器23响应输入端口模块20 (n) 提供给其的元数据分组并根据输出端口模块21 (n) 提供给其的工作状态信息作出的。若分组元数据处理器23接收到一分组的元数据分组后判定将要发送该分组的输出端口模块21 (n) 具有能力接收并发送该分组的话，该分组元数据处理器23便会将该分组的元数据分组提供给输出端口模块21 (n)，它进而将启动输入端口模块20 (n) 将该分组递送给其（即输出端口模块21 (n)）来便于输出端口模块21 (n) 对该分组的传输。反之，若分组元数据处理器23接收到一分组的元数据分组后判定将要发送该分组的输出端口模块21 (n) 没有能力接收并发送该分组的话，该输出端口模块21 (n) 便会不接收该分组的元数据分组，该分组元数据处理器23则会启动该输入端口模块20 (n) 来忽略该分组。该输入端口模块20 (n) 会一直缓存该分组，直到它从该输出端口模块21 (n) 接收到通知将该分组递送（“传递”）给其（即该输出端口模块21 (n)）用于传输为止，或直到它从该分组元数据处理器23接收到通知以忽略（“丢掉”）该分组为止，因而该输入端口模块20 (n) 提供对其所接收的分组进行输入排队。

另一方面，分组元数据处理器23有效地提供在作出传递/丢掉决定过程中所用的对信息进行的输出排队。但由于分组元数据处理器23可对这样一种元数据分组作出分组“传递/丢掉”决定，该元数据分组通常会比分组规模小很多，只需要包括一接收该分组的输入端口模块20 (n) 中指向该分组的指针、对将要接收该分组用于传输的输出端口模块21 (n) 的识别、也许还包括该分组长度，所以传送给分组元数据处理器23的数据量显著小于通常为整个分组的常规输出排队方案中传送的典型数据量。因而，按照本发明构成的交换节点11 (n) 避免了利用分组交换带宽将属于输出排队的交换节点中较典型的最终将被其忽略的分组传送给输出端口模块21 (n)。

而且，由于是将要发送一分组的输出端口模块21 (n) 启动正缓存该分组的输入



端口模块20 (n) , 在其要发送该分组时经过端口间分组交换机22将它 (即该分组) 传送给相应的输出端口模块21 (n) , 所以端口间交换机22能够以高得多的效率工作, 这是输入排队的交换节点的特点。

图3、图4、图5和图6分别示出图2所示交换节点11 (n) 中所用的输入端口模块20 (n) 、交换平面22 (m) 、分组元数据处理器23以及输出端口模块21 (n) 的功能框图。参照图3, 交换节点11 (n) 中所用的输入端口模块20 (n) 包括网络输入接口30、分组存储器31、元数据分组生成器32, 全部均由输入端口模块20 (n) 控制电路33来控制。网络输入接口30包括输入端口25 (n) (m) , 它们连接后接收表示相应通信链路13 (p) 上各分组的PKT_IN (n) (1) 至PKT_IN (n) (M) (通常为PKT_IN (n) (m)) 分组输入信号。尽管会理解可用其他类型的输入端口, 但在一个实施例中, 每一输入端口25 (n) (m) 形成为按照公知的SONET规范所构成的加入/丢掉多路复用器中的一部分。网络输入接口30将作为PKT_DATA分组数据信号的所接收的分组耦合至输入端口模块20 (n) 控制电路33。该输入端口模块20 (n) 控制电路33进而再在分组存储器31中缓存从网络接口30接收到的分组。

另外, 对于每一缓存分组, 输入端口模块20 (n) 控制电路33将信息提供给元数据分组生成器32来使之 (即元数据分组生成器32) 能够对该分组生成元数据分组。提供给元数据分组生成器的信息包括诸如分组的宿地址、分组存储器31中指向分组的指针、或许关于分组长度的信息这类信息。元数据分组生成器32存储交换节点管理处理器27提供给其的路由信息, 利用分组的宿地址来识别与将要通过其发送该分组的通信链路13 (p) 连接的输出端口模块21 (n) 及其输出端口26 (n) (m) 。若元数据分组生成器32能从该路由信息识别出合适的输出端口模块21 (n) 和输出端口26 (n) (m) , 元数据分组生成器32便由此并根据分组存储器中指向分组的指针、或许分组长度信息对该分组生成一元数据分组, 并将该元数据分组如INP (n) _M-D_ENQ输入 (n) 元数据分组排队信号所示那样提供给分组元数据处理器23。如下面所说明的那样, 元数据分组生成器32具体来说, 根据输入端口模块20 (n) 控制模块53提供的宿地址, 判定对将要接收该分组的输出端口模块21 (n) 和将要通过其发送该分组的输出端口模块26 (n) (m) 的识别, 将该识别与输入端口模块20 (n) 的识别符、分组存储器中指向该分组的指针、或许长度信息一起作为元数据分组提供给分组元数据处理器23。

如上所述, 交换节点可以没有某些宿地址的合适路由信息。若该元数据分组生成器32判定对输入端口模块20 (n) 控制电路33提供给其的一宿地址没有路由信息的

话，它（即元数据分组生成器）便可将此情形通知输入端口模块20（n）控制电路33。该输入端口模块20（n）控制电路33进而可执行本领域技术人员所清楚的规定操作，其中可包括例如从分组存储器当中忽略与该宿地址相关联的分组。另外，输入端口模块20（n）控制电路33可本身对该分组生成一表明分组没有送达其应到达的宿设备12（m_b）的分组，用于传输给源设备12（m_s）。该输入端口模块20（n）控制电路33可使该生成的分组能够按与其接收到的分组同样的方式传送。

元数据分组生成器32还从分组元数据处理器接收2个控制信号，包括INP（n）_ENQ_DRP输入（n）排队丢掉信号和INP（n）_PKT_DRP输入（n）分组丢掉信号。若分组元数据处理器23无法从元数据分组生成器32接收元数据分组，这种情形在元数据分组生成器32试图向其提供一元数据分组时，若分组元数据处理器23如下面结合图5所说明的那样拥塞的话便会发生，该分组元数据处理器23便会发出INP（n）_ENQ_DRP输入（n）排队丢掉信号以表明其无法受理元数据分组。若元数据分组生成器32当它（即元数据分组生成器32）向其提供元数据分组时从该分组元数据处理器23接收所发出的INP（n）_ENQ_DRP输入（n）排队丢掉信号的话，该元数据生成器32便使得输入端口模块20（n）控制电路33能够忽略该分组存储器31的分组。反之，若分组元数据处理器23能够从元数据分组生成器32接收元数据分组的话，就将不发出INP（n）_ENQ_DRP输入（n）排队丢掉信号，而是受理该元数据分组用于处理。

如上所述，分组元数据处理器23对由其受理的元数据分组进行处理，来判定与将要发送该分组的输出端口模块21（n）相关联的当前状态信息是否表明该输出端口模块21（n）具有足够的能力从输入端口模块20（n）接收该分组并通过相应的通信链路13（p）将它（即该分组）发送。若分组元数据处理器23受理元数据分组用于处理后作出一否定判断，即判定该输出端口模块21（n）没有足够能力来接收并发送该分组的话，它（即分组元数据处理器23）将在对输入端口模块20（n）所缓存的将被丢掉的特定分组提供识别的过程中，发出INP（n）_PKT_DRP输入（n）分组丢掉信号。若元数据分组生成器32从分组元数据处理器23接收所发出的INP（n）_PKT_DRP输入（n）分组丢掉信号的话，它便会通知输入端口模块20（n）控制电路33，它进而会从分组存储器当中忽略由INP（n）_PKT_DRP输入（n）分组丢掉信号所识别的分组。

输入端口模块20（n）控制电路33还从输出端口模块21（n）接收分组传送请求，并利用一分组分段生成器34控制从分组存储器31当中对分组的检索和将分组从分组存储器31传送至端口间分组交换机22以便传送至输出端口模块21（n）。分组传



送请求由OUT (1) /INP (n) _PKT_REQ输出 (1) /输入 (n) 分组请求至OUT (N) /INP (n) _PKT_REQ输出 (N) /输入 (n) 分组请求信号（通常为OUT (n') /INP (n) _PKT_REQ，索引“n'”是1至N的整数，可以但无需等于索引“n”的数值）所示，而输入端口模块20 (n) 提供的分组数据则由INP (n) _PKT_DATA输入 (n) 分组数据信号所示。输入端口模块20 (n) 控制电路33从一输出端口模块21 (n') 接收一分组传送请求后，该分组传送请求包括将响应该请求提供的指向该分组的指针，输入端口模块20 (n) 控制电路33使得分组分段生成器34能够从分组存储器31当中检索所请求的分组，将该分组分成相应的分段，将它们作为INP (n) _PKT_DATA_PLN (1) 输入分组数据平面 (1) 至INP (n) _PKT_DATA_PLN (P) 输入分组数据平面 (P) 信号（它们均来自于INP (n) _PKT_DATA输入 (n) 分组数据信号），与对将要发送该分组的输出端口模块21 (n') 的识别一起传送给端口间分组交换机22。如上面结合图2所述的那样，该端口间分组交换机22进而会将从输入端口模块20 (n) 接收到的分组传送至输出端口模块21 (n') 用于传输。

图4示出端口间分组交换机22中所用的交换平面22 (p) 的功能框图。参照图4，该交换平面22 (p) 包括多个交换模块60 (p) (1) 至60 (p) (N) （通常由标号60 (p) (n) 标识），这其中每一个与带相应索引的输出端口模块21 (n) 相关联，并从将要传送给带相应索引的输出端口模块21 (n) 的分组当中提供各分组分段。每一交换模块60 (p) (n) 包括一FIFO (先进先出缓存) 模块61 (p) (1) 至61 (p) (N) （通常由标号61 (p) (n) 标识），这其中每一个包括多个FIFO 62 (p) (n) (1) 至62 (p) (n) (F) （通常由标号62 (p) (n) (f) 标识）。每一交换模块60 (p) (n) 还包括多个输入多路复用器63 (p) (n) (1) 至63 (p) (n) (F) （通常由标号63 (p) (n) (f) 标识），这其中每一个提供一至带相应索引的FIFO 621 (p) (n) (f) 的输入端，此外还包括一个输出多路复用器64 (p) (n)。每一输入多路复用器62 (p) (n) (f) 由交换节点管理处理器27 (图2) 控制使之能够将INPUT (n) _PKT_DATA_ (PLN_p) 输入 (n) 分组数据 (平面 (p)) 信号从输入端口模块20 (n) 耦合到与其连接的FIFO 62 (p) (n) (f)，这时那些信号所表示分组分段将传送至相应索引的输出端口模块21 (n) 用于在网络上传送。每一交换模块60 (p) (n) 中的输出多路复用器64 (p) (n) 由输出端口模块21 (n) 控制来使之（即输出端口模块21 (n)）能够接收与其相关联的交换模块60 (p) (n) 中不同FIFO 62 (p) (n) (f) 输出的分组分段。通常，每一输出端口模块21 (n) 将控制该多路复用器64 (p) (n) 使之（即输出端口模块21 (n)）能够接收



其中输入端口模块20 (n) 是按循环轮流方式加载分组分段的全部不同FIFO 62 (p) (n) (f) 所输出的分组分段，以避免若输出端口模块21 (n) 一次从一个FIFO 62 (p) (n) (f) 接收整个分组的分组分段会发生的使一个或多个FIFO存满。

将会理解，若每一交换模块60 (p) (1) 中有“N”个FIFO (即“F”等于“N”) 的话，端口间分组交换机22将有效地成为非堵塞的，全部输入分组模块20 (n) 将能够在任何时刻传送分组至任何输出端口模块。

图5示出图2所示交换节点11 (n) 中所用的分组元数据处理器23的功能框图。参照图5，分组元数据处理器23包括多个处理器模块40 (1) 至40 (N) (通常由标号40 (n) 标识)，这其中每一个与带相应索引的输出端口模块21 (n) 相关联，并对带相应索引的输出端口模块21 (n) 作出分组传递/丢掉判定。该处理器模块40 (n) 总体上均类似。每一处理器模块40 (n) 均包括一输入队列41 (n)、分组传递/丢掉电路42 (n)、输出端口模块 (n) 状态信息存储43 (n) 以及输出FIFO (先进/先出缓存器) 44 (n)。输入队列41 (n) 从全部输入端口模块20 (n) 接收元数据分组，并将它们组织在单一队列中用来由分组传递/丢掉电路42 (n) 处理。输入队列41 (n) 接收INP (1) _M-D_ENQ输入 (1) 元数据分组排队至INP (N) _M-D_ENQ输入 (N) 元数据分组排队信号 (通常为INP (n) _M-D_ENQ) 所示的元数据分组排队请求。每一元数据分组排队请求包括如元数据生成器32所生成那样的元数据分组，如所注出的那样，它包括输入端口模块20 (n) 的识别符，分组存储器中指向分组的指针，或许标识与该元数据分组相关联的分组其长度的信息。

当从输入端口模块20 (n) 接收到一元数据分组排队请求时，处理器模块40 (n) 能够接收并在输入队列41 (n) 中对元数据分组进行排队的话，它便这么做。但当从输入端口模块20 (n) 接收到一元数据分组排队请求时处理器模块40 (n) 不能够接收并在输入队列41 (n) 中对元数据分组进行排队的话，它便就此通过保持INP (n) _ENQ_DRP输入 (n) 排队丢掉信号来通知输入端口模块20 (n)。这例如输入队列41 (n) 排满或输入队列41 (n) 中元数据分组数量超过一规定阈值的话会发生。如上所述，若处理器模块40 (n) 通过保持INP (n) _ENQ_DRP输入 (n) 排队丢掉信号来通知输入端口模块20 (n) 不能够对其输出的元数据分组进行接收和排队的话，输入端口模块20 (n) 便会忽略与该元数据分组相关联的分组。

输入队列41 (n) 将其中经过排队的元数据分组按序耦合至分组传递/丢掉电路42 (n)。分组传递/丢掉电路42 (n) 进而对输入队列41 (n) 耦合至其的每一元数据分组，根据存储在输出端口模块21 (n) 状态信息存储43 (n) 中与处理器模块40



(n) 相关联的输出端口模块21 (n) 的状态信息作出一传递/丢掉判定。所设置的存储43 (n) 中的该状态信息由相关联的输出端口模块21 (n) 提供, 如OP_PORT (n) _STATUS输出端口 (n) 状态信号所示, 形成图2所示在分组元数据处理器23和输出端口模块21 (n) 之间所传送的那样的OUT (n) _CTRL/STATUS输出 (n) 控制/状态信号其中之一。存在存储43 (n) 中的输出端口状态信息反映该输出端口模块的工作状态, 具体来说反映其在任何时刻接收输入端口模块20 (n) 输出的额外分组用于传输的能力, 可以是输出端口模块21 (n) 必须可供缓存从输入端口模块20 (n) 当中检索用于传输的分组的缓存量的函数, 并反映该可提供的缓存量是增加、减少还是基本不变。当分组传递/丢掉电路42 (n) 从输入队列41当中接收到元数据分组并与其相联系作出传递/丢掉判定时, 若存储43 (n) 中的状态信息表明输出端口模块21 (n) 具有能力从输入端口模块20 (n) 接收额外分组用于传输的话, 它(即分组传递/丢掉电路42 (n))便会在输出FIFO 44 (n) 中加载元数据分组。该分组传递/丢掉电路42 (n) 还可调整如该状态信息存储43 (n) 中存储的状态信息, 来反映将要由输出端口模块21 (n) 检索和发送的额外分组其元数据分组已经加载至FIFO 44 (n) 中这一事实。例如, 该元数据分组提供标识与元数据分组相关联分组其长度的长度信息的话, 而且存储43 (n) 中的状态信息提供的是关于输出端口模块21 (n) 所能提供的缓存量的信息的话, 该分组传递/丢掉电路42 (n) 便可以例如减少保存在存储43中以反映与元数据分组相关联的分组其长度的状态信息。

相反, 当分组传递/丢掉电路42 (n) 从输入队列41当中接收到元数据分组并与其相联系作出传递/丢掉判定时, 若存储43 (n) 中的状态信息表明输出端口模块21 (n) 没有能力从输入端口模块20 (n) 接收额外分组用于传输的话, 它便向从其接收到元数据分组的输入端口模块20 (n) 提供一丢掉通知, 如所发出的INP (n) _PKT_DRP输入 (n) 分组丢掉信号所示, 并向其提供一指向与该元数据分组相关联分组的指针。如上所述, 当输入端口模块20 (n) 收到来自处理器模块40 (n) 的通知时, 它(即输入端口模块20 (n))便可以忽略该分组。

输出FIFO 44 (n) 经连接向与处理器模块40 (n) 关联的输出端口模块21 (n) 提供元数据分组。当输出端口模块21 (n) 判定其处于从输入端口模块20 (n) 接收该分组状态时, 它便提供一由所发出的OUT (n) _FIFO_RETR_EN输出 (n) FIFO检索使能信号所示的输出FIFO检索请求, 它包括图2所示在分组元数据处理器23和输出端口模块21 (n) 之间所传送的那样的OUT (n) _CTRL/STATUS输出 (n) 控制/状态信号其中之一。响应来自输出端口模块21 (n) 的输出FIFO检索请求, 处理器模块40



(n) 将向输出端口模块21 (n) 传送如PKT_META-DATA分组元数据信号所示的元数据分组，其中包括图2所示在分组元数据处理器23和输出端口模块21 (n) 之间所传送的那样的OUT (n) _CTRL/STATUS输出 (n) 控制状态信号，其处于该输出FIFO 44 (n) 的首部。

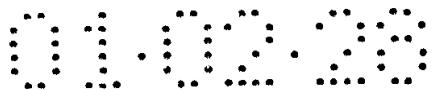
输出端口模块21 (n) 从分组元数据处理器23 (图2) 中与其相关联的处理器模块40 (n) (图5) 当中检索出元数据分组后，它(即输出端口模块21 (n))将使其中缓存分组的输入端口模块20 (n) 能够经过端口间分组交换机22将分组传送给它。图6示出交换机11 (n) 中所用的输出端口模块21 (n) 的功能框图。参照图6，输出端口模块21 (n) 包括一输出端口模块21 (n) 控制电路50、分组存储器51和网络输出接口52。输出端口模块21 (n) 控制电路50与交换节点11 (n) 的其他部件连接，并

(i) 提供给分组元数据处理器23中与输出端口模块21 (n) 相关联的处理器模块40 (n) 用于其传递/丢掉判定中，并更新反映输出端口模块21 (n) 从输入端口模块20 (n) 接收分组用于通过与其连接的通信链路13 (p) 传输的当前能力的工作状态信息；

(ii) 从分组传递/丢掉电路42 (n) 已传递并加载至输出FIFO 44 (n) 的处理器模块40 (n) 当中检索元数据分组，以及

(iii) 启动由相应的输入端口模块20 (n) 将与所检索的元数据分组(上面参照项(ii))相关联的分组经过端口间分组交换机22传送给输出端口模块21 (n)。通常，输出端口模块21 (n) 控制电路50生成由OUT (N) _FIFO_RETR_EN输出端口模块21 (n) FIFO检索使能信号所示的元数据分组检索请求，并将它们提供给分组元数据处理器23，尤其提供给与输出端口模块21 (n) 相关联的处理器模块40 (n) 的输出FIFO 44 (n) 来启动对元数据分组的检索(上面参照项(ii))。响应每一元数据分组检索请求，输出FIFO 44 (n) 若其具有该输出端口模块21 (n) 的元数据分组，将如PKT_META-DATA信号所示将一元数据分组提供给输出端口模块21 (n) 控制电路50。

检索到元数据分组后，输出端口模块21 (n) 控制电路50将使得元数据分组中所标识的输入端口模块20 (n) 能够将分组传送给输出端口模块21 (n) (上面项(iii))。在那操作中，输出端口模块21 (n) 控制电路50将生成一由OUT (n) /INP (n') _PKT_REQ输出 (n) /输入 (n') 分组请求信号所示的分组传送请求，用于传送给正对分组进行缓存的输入端口模块20 (n')。每一分组传送请求包括指向



所要传送的分组的指针，输出端口模块21（n）控制电路50从所检索的元数据分组当中获得该指针。与图3相联系如上所述，输入端口模块20（n'）接收到分组传送请求后，它从其内部缓存器当中检索该分组，并将该分组提供给端口间分组交换机22用于传送给输出端口模块21（n）。

端口间分组交换机22便进而如OUT（n）_PKT_DATA输出（n）分组数据信号所示将分组传送给输出端口模块21（n）控制电路50。更为具体地来说，包括端口间分组交换机22的不同交换平面22（p）将表示相应的分组分段的相应的OP（n）_PKT_DATA_（PLN_1）输出（n）分组数据（平面1）至OP（n）_PKT_DATA_（PLN_P）输出（n）分组数据（平面P）信号提供给分段/分组生成器54。分段/分组生成器54起到使将要从端口间分组交换机22接收到的不同分组重新生成的作用，这些在传送前将缓存在分组存储器51中。

输出端口模块21（n）控制电路50从端口间分组交换机22接收到分组后，便将它提供给网络接口52，用于通过合适的通信链路13（p）传输。通常，在传输前，输出端口模块21（n）控制电路50会将分组缓存在分组存储器51中。网络输出接口52包括输出端口26（n）（m），它们与通信链路13（p）连接以便有利于PKT_OUT（n）（1）至PKT_OUT（n）（M）（通常为PKT_OUT（n）（m））分组输入信号所示的分组的传输。尽管会理解可以采用其他类型的输入端口，但在一个实施例中，每一输出端口26（n）（m）形成为按照公知的SONET规范构成的加入/丢掉多路复用器其中一部分。

如上所述，输出端口模块21（n）控制电路50还向与分组元数据处理器23中输出端口模块21（n）相关联的处理器模块40（n），提供并更新反映输出端口模块21（n）从输入端口模块20（n）接收分组用于通过与其连接的通信链路13（p）传输的当前能力的工作状态信息，来用于其传递/丢掉判定（上面参照项（i））。因此，该工作状态信息将最好反映分组存储器51可用的缓存量，也可以表示该量是否增加、下降或保持不变，所有这些都提供关于输出端口模块当前可从输入端口模块20（n）接收分组的能力的信息。将会理解，若工作状态信息反映分组存储器可用的缓存量，输出端口模块21（n）便可以随着其发送分组，通过使得状态信息存储43（n）中的信息被增加一反映相应分组长度的数值来更新工作状态信息。

如上所述，参照图3，输入端口模块20（n）控制电路33向元数据生成器32提供关于从网络输入接口30接收到和缓存在分组存储器31中的每一分组的信息，其中包括宿地址和存储器31中指向分组的指针，元数据生成器32生成一用于其的元数据分

组，用于传送至分组元数据处理器23。在生成元数据分组过程中，元数据生成器32从该宿地址识别合适的路由（若它有某一路由的话），它识别将要接收该分组的特定输出端口模块21（n），以及经过其将要发送该分组的其中一个输出端口26（n）（m），以便将该分组发送至宿设备12（m₀）或网路中沿到达该宿设备12（m₀）路径的下一交换节点。

顺便说一下背景技术，在一个实施例中采用了IP分组路由协议，分组的合适路由被识别时无需基于分组中宿地址与交换节点路由信息的完全符合。而是，基于与宿地址最长“前缀”的符合来选择合适路由，这种前缀包括宿地址的高有效位。如上所述，每一路由包括地址信息和对输出端口模块21（n）和输出端口26（n）（m）的标识，通过比较该前缀和该路由的地址信息部分来判定符合。当前，IP地址（如上面所用的源地址或宿地址）是具有A₃₁…A₀形式的32位整数，其中每一“A_i”均为一二进制数字。这样，合适的路由就是那个其最长位序列A₃₁…A_j（其索引j为最小值）符合如交换节点所保存那种路由中的地址信息。若发现对于宿地址不符合，该交换节点便不具有该特定宿地址的路由。

可以采用若干种机制来判定这种符合。例如，可以将路由信息按多个数据表形式组织，每一数据表对一特定前缀长度存储路由选择信息。在此情况下，这些数据表可从与最长前缀长度至较短的前缀长度相关联数据表依次顺序处理。在表中发现宿地址前缀和该数据表中所存储的地址信息之间符合的第一路由便为合适路由；若任何数据表中均未发现符合，则该交换节点不具有该宿地址的路由。但与这种安排相联系会发生若干问题，最为关注的问题是大量的数据表可能需要由交换节点生成并保存，这可能使该交换节点的复杂性和需要存储相应数据表的存储器数量增加。另外，在通常情况下，在取得地址的路由选择信息时会存取大量数据表，尤其要识别具有该地址的合适路由选择信息的数据表。若按当前所提议的那样，互联网地址所用的地址位个数增加到128位的话，这些问题就会加剧。

针对这些问题，元数据生成器32在一个实施例中利用一经过压缩的检索树（“trie”）方案，在一替代实施例中利用一二叉搜索树方案，以便就输入端口模块控制电路33（图2）与其耦合的互联网地址生成路由选择信息。图7中示出由标号70标识的经过压缩的检索树方案的功能框图，图8中则示出由标号90标识的二叉搜索树方案的功能框图。这两个实施例中，元数据生成器32均在多个流水线处理级中按流水线方式生成路由选择信息，在每一处理级中执行一项如下面结合图7和图8所作的说明那样生成合适路由选择信息的操作。因为该信息是按流水线方式生成的，因

而元数据生成器32可同时结合多个分组的互联网地址来执行各种处理操作，而且减少本该可能需要访问的数据表数量。

先参照图7，经过压缩的检索树方案70包括多个地址寄存器71 (1) 至71 (N-1) (通常由标号71 (n) 标识)，每一个与多个处理级72 (1) 至72 (N-1) 中的一个相关联。对于下面所述的用途还提供一“第N”处理级72 (N)。这里，将通常用标号72 (n) 来标识处理级72 (1) 至72 (N)。在一系列流水线处理级中进行路由选择信息的生成，其中，

(i) 在第一流水线处理级中，地址寄存器71 (1) 一开始接收将要生成路由选择信息的分组其互联网地址，然后该相关联处理级72 (1) 在图7中作为“ADRS_FIELD_0”和“ADRS_FIELD_1”标识的两个地址字段中利用规定个数的高有效地址位，来生成耦合至处理级72 (2) 的信息；

(ii) 在第二流水线处理级中，处理级71 (1) 中的互联网地址传送至地址寄存器71 (2)。处理级72 (1) 耦合至处理级72 (2) 的信息，与当前在地址寄存器71 (2) 中互联网地址在图7中作为“ADRS_FIELD_2”标识、接在高有效地址位之后的规定个数的地址位一起，由处理级72 (2) 用来生成耦合至处理级72 (3) 的信息，依此类推，直到

(N-1) 在最后的“第N-1”流水线处理级中，处理级“N-1”中的互联网地址传送至地址寄存器71 (N-1)。处理级72 (N-2) 耦合至处理器72 (N-1) 的信息，与当前在地址寄存器71 (N-1) 中互联网地址在图7中作为“ADRS_FIELD_N-1”标识的规定个数的低地址位一起，由处理级72 (N-1) 用来生成耦合至处理级72 (N) 的信息。

将会理解，对于如上所述的每一流水线处理级来说，当互联网地址从地址寄存器71 (1) 传送至地址寄存器71 (2) 时，可使另一互联网地址移入地址寄存器71 (1)，对后续的地址寄存器71 (n) 也同样，这样有利于对互联网地址的流水线处理。

每一处理级72 (n) 包括一数据表73 (n) 和相关联的存储和/或处理电路。处理级72 (1) 至72 (N-1) 中每一个均包括一数据表73 (n)，每一个均包括多个记录项73 (n) (y_n)。每一记录项73 (n) (y_n) 进而包括多个字段，其中包括数据/指针标志75，一映射字段76和一数据/指针字段77 (直接在数据表73 (2) 中示出)。该映射字段76包括一系列位 $M_{B-1} \dots M_0$ (通常用“ M_b ”标识)，这其中每一个与地址寄存器72 (n) 中ADRS_FIELD_n中的各地址位所示的按二进制编码的数值中可能的某一数

值相关联。通常，若数据/指针标志75置1的话，该数据/指针字段77便对与地址寄存器71（n）中ADRS_FIELD_n中的各地址位相关联的全部地址包含路由选择信息。反之，若数据/指针标志75清零，但与地址寄存器71（n）中ADRS_FIELD_n中的各地址位的二进制编码数值相关联的映射字段76中的位M_b置1的话，便将该数据/指针字段77与映射字段76其中一部分一起，用于生成一在下一处理级72（n+1）中数据表73（n+1）当中指向记录项73（n+1）（y_{n+1}）的指针。

该映射字段76还包括一默认标志M_B，若数据/指针标志75清零，且与地址寄存器71（n）中ADRS_FIELD_n的二进制编码数值相关联的位M_b清零的话，便采用该默认标志。若位M_b清零但默认标志M_B置1的话，该数据/指针字段77便与整个映射字段76（而非默认标志M_B本身）一起，用于生成一在下一处理级72（n+1）中数据表73（n+1）当中指向记录项73（n+1）（y_{n+1}）的指针。最后，若该数据/指针标志75、与地址寄存器71（n）中ADRS_FIELD_n的二进制编码数值相关联的位M_b以及默认标志M_B都清零的话，该地址寄存器71（n）中便没有该互联网地址的路由选择信息。

处理级72（N）还包括一其中包含多个记录项73（N）（Y_N）的数据表73（N）。但在数据表73（N）中，每一记录项仅包括一个字段，它进而包括路由选择信息。

如下处理数据表中的信息73（n）（y_n）来生成路由选择信息。处理级72（1）中，ADRS_FIELD_0的二进制编码数值指向将要采用的数据表73（1）中的特定记录项。每一处理级72（1）至72（N-1）还包括如下操作的处理级（n）处理器80（n）。处理级（1）处理器80（1）从地址寄存器71（n）中接收ADRS_FIELD_1，并从数据表73（1）中由ADRS_FIELD_0所指向的记录项73（1）（y₁）当中接收字段75、76和77的内容，并响应其执行若干操作。具体来说，处理级（1）处理器80（1）执行如下操作：

(i) 确定数据/指针标志75是否置1，若是，便将数据/指针标志和数据/指针字段77的内容提供给下一处理级72（1），该置1的数据/指针标志表明数据/指针字段77所提供的信息包括路由选择信息（“RT_INFO”），

(ii) 若数据/指针标志75清零，但与地址寄存器71（1）中ADRS_FIELD_1中的二进制编码数值相关联的选定记录项73（1）（y₁）中映射字段76中的位M_b置1的话，便对M₀、M₁、…直至但不包括置1的位M_b进行计数，并将该计数与数据/指针字段77中所含内容相加，将该求和数据/指针标志提供给下一处理级72（2），该清零的数据/指针标志表明该求和将用于该处理级72（2）在数据表73（2）中识别记录项73（2）（y₂），

(iii) 若数据/指针标志75和与地址寄存器71 (1) 中ADRS_FIELD_1中的二进制编码数值相关联的选定记录项73 (1) (y_1) 中映射字段76中的位 M_0 两者均清零的话, 便确定位 M_0 是否置1, 若是这样, 便对 M_0 、 M_1 、…直至但不包括置1的位 M_b 进行计数, 并将该计数与数据/指针字段77中所含内容相加, 将该求和数据/指针标志提供给下一处理级72 (2), 该清零的数据/指针标志表明该求和将用于该处理级72 (2) 在数据表73 (2) 中识别记录项73 (2) (y_2); 以及

(iv) 若数据/指针标志75、与地址寄存器71 (1) 中ADRS_FIELD_1中的二进制编码数值相关联的选定记录项73 (1) (y_1) 中映射字段76中的位 M_0 以及位 M_b 均清零的话, 它(即处理级(1)处理器80 (1))向下一处理级72 (2) 提供一表明该地址寄存器71 (1) 中没有该地址的路由选择信息这一通知。

因而与上述(ii)项和(iii)项相联系, 将会理解, 记录项73 (1) (y_1) (直至最大值“B”的记录项73 (2) (y_2))中的映射字段76中, 数据表73 (2) 仅需具有至多与位 M_0 、 M_1 、 M_b 的数量相对应的记录项73 (2) (y_2) 数量, 这些映射字段对于数据/指针标志75清零(如上所述表明数据/指针字段77包含一指针)的那些记录项73 (1) (y_1) 才被设置(直至最大值“B”的记录项73 (2) (y_2)), 上述设置的73 (2) (y_2) 的最多数量可显著小于假使对每一记录项73 (1) (y_1) 和对地址寄存器71 (2) 中ADRS_FIELD_2中各地址位的每一可能的二进制编码数值提供一个记录项73 (2) (y_2) 要用的数量。

处理级(1)处理器80 (1)生成了如上所述的输出信息后, 地址寄存器71 (1) 中的互联网地址将传送给地址寄存器71 (2) 用于处理级72 (2) 的处理。处理级72 (2) 结合处理级72 (1) 的输出根据地址寄存器71 (2) 中地址的下一地址字段ADRS_FIELD_2(未分开示出)处理各地址位。如上所述, 处理级72 (2) 包括一数据表73 (2), 其与处理级72 (1) 中的数据表73 (1) 同样构成。另外, 处理级72 (2) 包括一按与上面结合处理级(1)处理器80 (1)所说明的同样方式工作的处理级(2)处理器80 (2), 只是存在下面不同, 即如果处理级72 (2) 从数据/指针标志置1的处理级72 (1) 中接收信息, 并表明从该处理级72 (1) 中接收到的信息要么是路由选择信息, 要么是给处理级72 (2) 的说明该元数据生成器32对于当前位于地址寄存器71 (2) 中的互联网地址没有路由选择信息的通知, 这样的话, 处理级(2)处理器80 (2)便将该信息耦合至下一处理级72 (3), 而不会结合该处理级72 (2) 的数据表73 (2) 中的记录项73 (2) (y_2) 执行处理操作。

随着通过后续地址寄存器71 (3) 至71 (N-1) 对地址的传送, 将对后续处理级

72 (3) 至72 (N-1) 重复上述操作。处理级72 (N-1) 处理了互联网地址中地址字段ADRS_FIELD_N-1后，该处理级72 (N-1) 还将向处理级72 (N) 提供路由选择信息或指针。处理级72 (N) 还包括一处理级 (N) 处理器80 (N) 。若该处理级 (N) 处理器80 (N) 从处理级72 (N-1) 接收路由选择信息（其中包括默认路由选择信息或元数据生成器32对该地址不具有路由选择信息这一通知），便会将该路由选择信息作为元数据生成器32的输出路由选择信息。反之，若该处理级72 (N) 从该处理级72 (N-1) 接收指针，在其数据表73 (N) 中指向一记录项73 (N) (Y_N) 的话，该处理级 (N) 处理器80 (N) 便会提供该记录项内容作为互联网的路由选择信息。

图8示出包括一用于对输入端口模块控制电路33 (图2) 与其耦合的互联网地址生成路由选择信息的二叉搜索树方案的替代实施例的功能框图。图8中示出由标号90标识的二叉搜索树方案的功能框图。首先，该二叉搜索树方案实际定义多个二叉搜索树。每一二叉搜索树所具有的节点组织成为从最高级的单个“根”节点至最低级的多个叶节点的多个等级。根节点和叶节点间的二叉搜索树中每一节点都在稍高一级有一个父节点，在稍低一级有两个子节点，根节点也具有两个子节点（但没有父节点），每一叶节点具有一个父节点（但没有子节点）。对于每一节点，其中一个子节点在此称为“右手”子节点，另一个则称为“左手”子节点。将会理解，对于每一二叉树，级数“NUM_LEVELS”是叶节点数“NUM_LEAVES”以2为底的对数函数，具体来说 $NUM_LEVELS = \log_2 (NUM_LEAVES) + 1$ 。

通常采用一二叉搜索树如下按与非叶节点数相对应的迭代数，使一输入值与叶节点中某一个相关联。叶节点以上二叉树的每一节点与一比较值相关联，叶节点级的各叶节点与将要同输入值相关的各种数值相关联。在第一迭代中，输入值与同根节点相关联的比较值相比较，以及

- (i) 如果该输入值大于或等于该比较值，将选择右手子节点，但
- (ii) 如果该输入值小于该比较值，将选择左手子节点。

然后，若在第二次迭代中该第二级节点不是叶节点，将结合与该第二级中选定节点相关联的比较值执行比较，至于根节点（上面参照项 (i) 和 (ii) ）将按同样方式选定其子节点其中之一。叶节点级以上相应级各节点的相应迭代将重复上述操作。当结合该叶节点级以上正好一级的节点，按与上面所述相同的方式（上面参照项 (i) 和 (ii) ）来执行该操作时，便选定该叶节点级中、进而与同输入值相关联的数值相关联的子节点。

由对结合二叉搜索树执行的操作所进行的说明，将会理解，与该叶节点级以上

各级节点相关联的比较值，需要起到将可行输入值范围划分为多个区间的作用。这样，若可行输入值范围为 [0, Max]，其中“Max”表示最大的可行输入值，“[x, y]”表示包括端点在内的范围，该根节点将整个范围分成2个区间，即 [0, A] 和 [A, Max]，其中“A”是与根节点相关联的比较值，而 “[x, y]”则表示一包括左端点“x”但不包括右端点“y”的区间。因而，输入值如果落在左区间 [0, A)，便选定左手子节点，但输入值如果落在右区间 [A, Max]，便选定右手子节点。该根节点的左手子节点进一步将区间 [0, A] 分成2个区间 [0, B) 和 [B, A]，该根节点的右手子节点进一步将区间 [A, Max] 分成2个区间 [A, C) 和 [C, Max]。通常，叶节点级以上的相对较低级的节点，起到将高一级限定的区间集合进一步分成连续的更小区间这一作用。因而将会理解，二叉搜索树需要起到使输入值与一区间相关联，而每一区间与一叶节点相关联的作用。

本发明的来龙去脉中，该输入值是互联网地址，而与叶节点相关联的数值则包括路由选择信息。

按照此技术背景，将结合图8说明该二叉搜索树方案90。参照图8，该二叉搜索树方案包括一输入寄存器91、一输入数据表92、一系列树分级95 (1) 至95 (N) (通常由标号95 (n) 标识) 以及一路由选择信息表112。通常，树分级95 (1) 至95 (N) 存储包括叶节点级以上二叉搜索树方案90的二叉搜索树各节点的节点信息，而路由选择信息表则存储包括二叉搜索树方案的二叉搜索树各叶节点的路由选择信息。如上所述，该二叉搜索树方案90定义多个二叉搜索树，但该二叉搜索树可具有不同的级数，因而对于全部的二叉搜索树，树分级95 (n) 的数量“N”是叶节点级以上最大级数的函数。但由于全部叶节点都需要与路由选择信息表112相关联，因而不同二叉搜索树的根节点可与不同的某些树分级95 (n) 相关联。也就是说，若二叉搜索树具有相对较少的等级，其根节点便与索引“n”相对靠近“1”的树分级95 (n) 相关联，索引“1”与最后一树分级95 (1) 相关联。相反，若二叉搜索树具有相对较多的分层，其根节点便与索引“n”相对靠近“N”的树分级95 (n) 相关联，索引“N”与第一树分级95 (N) 相关联。由互联网地址的高有效位部分确定一互联网地址将与其相关联的特定二叉搜索树。对互联网地址中高有效位部分的每一二进制编码数值，输入数据表92包含一对将要用于该二进制编码数值的二叉搜索树的根节点相关联的树分级95 (n) 进行识别的指针。

具体来说，输入数据表92包括多个记录项92 (1) 至92 (N) (通常由标号92 (n) 标识)，其中每一个与互联网地址高有效位部分的一个二进制编码数值相关

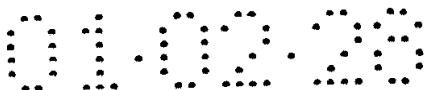
联。每一记录项92 (n) 包括2个字段，即数据表延迟字段93和下一数据表索引字段94。数据表延迟字段93包含一用于识别与搜索树根节点相关联的树分级95 (n) 的数值，该下一数据表索引字段94包含某一树分级95 (n) 所保存的数据表（如下面所述）中指向一记录项的指针，而该树分级95 (n) 包含一将要用于与互联网地址低有效位部分相比较的比较值。

树分级95 (N) 至95 (1) 都同样，故图8中仅详细示出树分级95 (N)。如图8所示，树分级95 (N) 包括一低有效位地址存储97，一数据表96 (N) 以及如下面所述的处理部件。该低有效位地址存储97在访问了输入数据表92中的记录项92 (n) 之后，接收并存储地址寄存器91输出的互联网地址低有效位部分。同时，将在存储体100中存储输入数据表92中与地址寄存器91中互联网地址的高有效位部分相关联的记录项92 (n) 的数据表延迟字段93和下一数据表索引94的内容。该数据表96 (N) 包括多个记录项96 (N) (0) 至96 (N) (Y_N) (通常由标号96 (N) (y_N) 标识)，这其中每一个与二叉比较树中一节点相关联，并包括一比较值字段101。每一记录项96 (N) (y_N) 中该比较值字段101与最大的二叉比较树（即叶节点级以上“N”级的二叉比较树）的根节点其节点信息相对应。比较值字段101存储的是一将与存储97中互联网地址的低有效位部分相比较的比较值。

将会理解，每一树分级95 (N-1), …, 95 (1) (通常由标号95 (n) 标识) 也会具有数据表96 (N-1), …, 96 (1) (通常由标号96 (n) 标识)，这其中每一个将具有 Y_{N-1} , …, Y_1 (通常为 Y_n) 个记录项，这其中每一个也存储比较值。每一数据表96 (N-1), …, 96 (1) 中至少某些记录项将与相应的前一级数据表96 (N), …, 96 (2) 中各记录项的相应二叉比较树的子节点相对应，这样每一数据表96 (N-1), …, 96 (1) 将具有2倍的前一级数据表96 (N), …, 96 (2) 中的记录项数来适应该子节点。另外，对于就此利用其树分级开始进行二叉比较树的树分级95 (N-1), …, 95 (1) 中一树分级来说，与该树分级相关联的数据表96 (N-1), …, 96 (1) 也将具有各自与由该树分级开始的各个二叉比较树的根节点相关联的另外数目的记录项，在那些情况下，数据表96 (N-1), …, 96 (1) 可能具有超过2倍的与前一级相关联数据表96 (N), …, 96 (2) 中的记录项数。

该树分级95 (N) 的处理部件执行若干操作，其中包括：

(i) 选择数据表96 (N) 中由存储体100中表索引“TBL_IDX”部分指向的一记录项96 (N) (y)，进而与输入数据表中所选定记录项92 (n) 的下一数据表索引字段94的内容相对应，



(ii) 将存储97中的互联网地址低有效位部分的二进制编码数值与所选定记录项96 (N) (y_1) 中字段101的比较值相比较,

(iii) 根据上面 (ii) 项的比较结果生成一在下一级数据表96 (N-1) 中指向一记录项的指针, 以及

(iv) 根据存储体100的数据表延迟数值和树分级索引 “N” 之间的比较, 选择上面 (iii) 项生成的指针或与下一树分级95 (N-1) 耦合的存储体100的表索引。若存储体100中的数据表延迟数值与树分级95 (N) 的索引 “N” 相对应的话, 存储体100中表索引所指向的记录项便指向作为将要用于其低有效位部分位于存储97中的互联网地址的二叉搜索树的根节点的树分级95 (N) 中的记录项96 (N) (y_N)。同样, 若存储体100中数据表延迟数值小于树分级95 (N) 的索引 “N” 的话, 存储体100中表索引所指向的记录项便指向将要用于其低有效位部分位于存储97中的互联网地址的二叉搜索树中但在根节点以下的树分级95 (N) 中的记录项96 (N) (y_N)。不论哪一种情况, 将根据存储体97中互联网地址低有效位部分二进制编码数值与所选定记录项96 (N) (y_N) 的字段101中比较值的比较结果, 将其中一个所生成的指针与树分级95 (N) 中存储体100的数据表延迟数值一起, 耦合到下一树分级95 (N-1) 的存储体100以用作一表索引。

相反, 若存储体100中的数据表延迟数值大于该树分级索引的话, 存储体100中表索引所指向的记录项便不指向作为将要用于其低有效位部分位于存储97中的互联网地址的二叉搜索树的一节点的树分级数据表中一记录项。在此情况下, 处理部件便将存储体100的索引值与存储体100的数据表延迟数值一起, 耦合到下一树分级95 (N-1) 以用作表索引。

树分级95 (N) 的处理部件包括若干个单元, 其中包括比较器98和99、多路复用器104和105、乘法器106以及加法器107。乘法器106和加法器107起到生成指向下一树分级95 (N-1) 中记录项指针的作用。如上所述, 在二叉比较树中, 每一节点具有2个子节点。因而在一实施例中, 生成下一树分级中的记录项指针, 以便96 (N) (0) 的子记录项将位于树分级95 (N-1) 数据表中记录项96 (N-1) (0) 和96 (N-1) (1) 中, 而子记录项96 (N) (1) 将位于记录项96 (N-1) (2) 和96 (N-1) (3), 依此类推。因而在该实施例中, 乘法器106和加法器107可通过乘法和加法生成指针, 可通过使存储体100中的表索引值乘以数值 “2” (将由乘法器106执行) 来生成与记录项96 (N) (y_N) 相关联的二叉树子节点其中之一的指针, 通过使数值 “1” 与乘法器106生成的数值相加来生成另一子节点的指针。在该情况下, 下

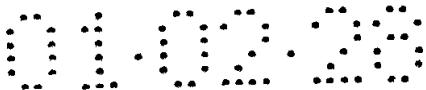
一树分级95 (N-1) 的数据表96 (N-1) 中的记录项96 (N-1) (0) 至96 (N-1) ($2Y_N - 1$) 将与其根节点与同树分级95 (N) 相关联的数据表96 (N) 中的记录项96 (N) (y_N) 相关联的二叉比较树中的子节点相关联。若任何二叉比较树具有与树分级95 (N-1) 相关联的根节点的话，那些根节点将与数据表96 (N-1) 中相应的另外记录项96 (N-1) ($2Y_N$)，…等相关联。数据表96 (N-2)，…，96 (1) 具有同样组成。

比较器98则起到将存储97的互联网地址的低有效位部分与所选定记录项96 (N) (Y) 的字段101当中的比较值相比较并响应该比较结果来控制多路复用器104的作用。若比较器98判定存储97的互联网地址的低有效位部分大于或等于所选定记录项96 (N) (Y) 的字段101当中的比较值，多路复用器104便将加法器107生成的指针耦合至多路复用器105其中一个输入端。相反，若比较器98判定存储97的互联网地址的低有效位部分小于所选定记录项96 (N) (Y) 的字段101当中的比较值，多路复用器104便将乘法器106生成的指针耦合至多路复用器105的相同输入端。存储体100的表索引则耦合至多路复用器105的第二输入端。

该多路复用器105进而又被比较器99输出的信号所控制。比较器99从存储体100当中接收数据表延迟数值，还接收一与树分级索引“N”相对应数值。若该数据表延迟数值小于或等于该树分级索引，多路复用器105便将多路复用器104所提供的数值即乘法器106或加法器107所生成的指针耦合至下一树分级95 (N-1)。相反，若存储体100中数据表延迟其数值大于该树分级索引，多路复用器105便将存储体100的表索引耦合至下一树分级95 (N-1)。

将会理解，如乘法器106、加法器107和多路复用器104所提供的那样说明的功能，可在不提供用于此用途的专门部件的情况下实现。因为表索引乘以“2”的乘法与包括存储体100中表索引的二进制数字“左移”1位相对应，而加“1”则与在低有效位位置中设定二进制数字相对应，可通过将包括存储体100中表索引的各个位 $TI_H \dots TI_0$ （未分开图示）耦合到多路复用器105相关联输入端（即图示为接收多路复用器104所输出信号的输入端）的信号通路 $S_{H+1} \dots S_1$ （也未分开图示）。低有效位信号通路 S_0 从比较器98接收信号，因而，数据表96 (N) 中所选定记录项中的比较值与存储中的地址的低有效位部分的比较结果，控制所选定的是与下一树分级95 (N-1) 相关联的数据表96 (N-1) 中记录项96 (N-1) (y_{N-1}) 当中的那些。

将会理解，若相应二叉比较树的记录项和节点间的关联没有如上所述限制的话，相应树分级95 (N)，…，95 (2) 的数据表96 (N)，…，96 (2) 中的记录项



96 (N) (y_N) , …, 96 (2) (y_2) 便会包括指向相应下一树分级95 (N-1) , …, 95 (1) 中记录项96 (N-1) (y_{N-1}) , …, 96 (1) (y_1) 的直接指针。

本发明具有若干优点。具体来说，其提供的交换节点提供了一输入排队的交换节点的规模可伸缩性，同时基本上保持了输出排队的交换节点其交换网路的使用效率。因为分组在输入端口模块20 (n) 一直缓存到（由分组元数据处理器23）作出将通过相应的输出端口模块21 (n) 发送这些分组这种判定时为止，甚至到由相应的输出端口模块21 (n) 请求分组应传送给其以便传输时为止，所以仅需要“N”个缓存器或队列，每一输入端口模块20 (n) 需一个缓存器或队列，而一输出排队的交换节点却需要 N^2 个缓存器。另一方面，由于结合从全部输入端口模块20 (n) 当中为此排队的元数据分组对每一输出端口模块21 (n) 作出有关是传递还是丢掉一分组这种判定，因而对全部输入端口模块20 (n) 所作的传递/丢掉判定是按与输出排队的交换节点中的方式基本相同的方式进行的。但由于对元数据分组作出传递/丢掉的判定，元数据分组通常比经过网络传送的典型分组的规模小许多，通过使丢掉的分组不经过分组交换机22传递，所以可减小交换带宽，否则如同输出排队的交换节点那样，这些交换带宽将被用来传送后来由输出端口模块21 (n) 丢掉的分组。

而且，由于在分组从相应的输入端口模块20 (n) 传送至相应的输出端口模块21 (n) 之前作出传递/丢掉决定，若输出端口模块21 (n) 拥塞的话，分组将很可能被输出端口模块21 (n) 丢掉，所以这些丢掉分组将不会占用经过端口间分组交换机22的输入和输出端口模块间的带宽。这通过确保将由输入端口模块仅向其传送当其到达输出端口模块时便很可能将要发送的那些分组，来起到使输入和输出端口模块间的分组通信量优化的作用。这提供给交换节点11 (n) 处理相对高通信容量的能力，因为没有资源浪费于经过该交换节点11 (n) 传送而在它们到达相应输出端口模块21 (n) 时却被丢掉的分组，否则输出端口模块21 (n) 是会发送它们的，但由于拥塞而无法发送。

总体而言，由于传递/丢掉决定是分组元数据处理器就每一输出端口模块21 (n) 根据全部输入端口模块20 (n) 提供给其的“全局”信息作出的，所以交换节点11 (n) 能够作为单个实体工作，而不是作为更小实体的集合那样工作。该全局信息提供给相应的输出端口模块21 (n) 一表示全部输入端口模块20 (n) 状态的全景图。在不提供这种全景图的设备中，可根据输入端口模块20 (n) 的团组所提供的信息作出决定，这种团组信息是汇总成为一相应输出端口模块21 (n) 所采用的公共视图。根据输入端口模块20 (n) 如何形成团组以及各团组所提供的信息如何汇总，可

能忽略关于各团组中相应输出端口模块21 (n) 状态的重要信息，进而可能造成该输出端口模块21 (n) 工作时就象该交换节点是一基于输入端口模块20 (n) 与相应团组的关联的较小实体的集合。这会降低该交换节点作为一整体的通信量处理能力。

将会理解，可对上面结合图2至图6说明的交换节点11 (n) 进行种种修改。例如，虽然是将交换节点11 (n) 说明为利用传送消息用的IP路由选择协议，但会理解，可以采用包括但不限于IP分组、ATM单元、帧延迟分组等利用可变或固定长度分组的若干方便的路由选择协议。

另外，尽管将元数据处理器40 (n) 说明为对相应的输出端口模块21 (n) 提供一分开的状态信息存储43 (n)，但会理解，元数据处理器23也可以具有一高速缓存存储器（未图示）来存储对处于拥塞状态的输出端口模块21 (n) 的标识，它可以被全部元数据处理器40 (n) 中的分组传递/丢掉电路42 (n) 所使用。如所希望的、在某一时刻只有很少数量的输出端口模块21 (n) 拥塞的话，通过采用高速缓存存储器所需要的存储器数量，便会小于对每一元数据处理器40 (n) 提供一存储43 (n) 所需的数量。

而且，交换节点11 (n) 尽管说明为利用交叉点接线器作为端口间分组交换机，但会理解，可以采用任意的若干其他形式的交换机。

另外，分组元数据处理器23尽管说明为提供一个与每一输出端口模块21 (n) 相关联的处理器模块40 (n)，但会理解，它也可以提供与每一输出端口26 (n) (m) 相关联的处理器模块。在此情况下，相应的输出端口模块21 (n) 将向处理器模块提供与相应的输出端口26 (n) (m) 相关联的工作状态信息，处理器模块将它们用于传递/丢掉的判定。此外，每一输出端口模块21 (n) 可调度对与其输出端口26 (n) (m) 相关联的相应处理器模块所传递的元数据分组的检索和对与其相关联分组的后续检索，以便能够使经过全部输出端口26 (n) (m) 的分组传输最大化。

另外，元数据生成器32尽管说明为利用单个比较树来识别宿地址的合适路径，但会理解元数据生成器32可构成为利用若干个这种比较树，所要利用的对特定比较树的选择将由例如宿地址高有效位选定位所表示的二进制编码数值来决定。在此情况下，将根据剩余的低有效位组成比较表，并结合宿地址的低有效位作出种种比较。

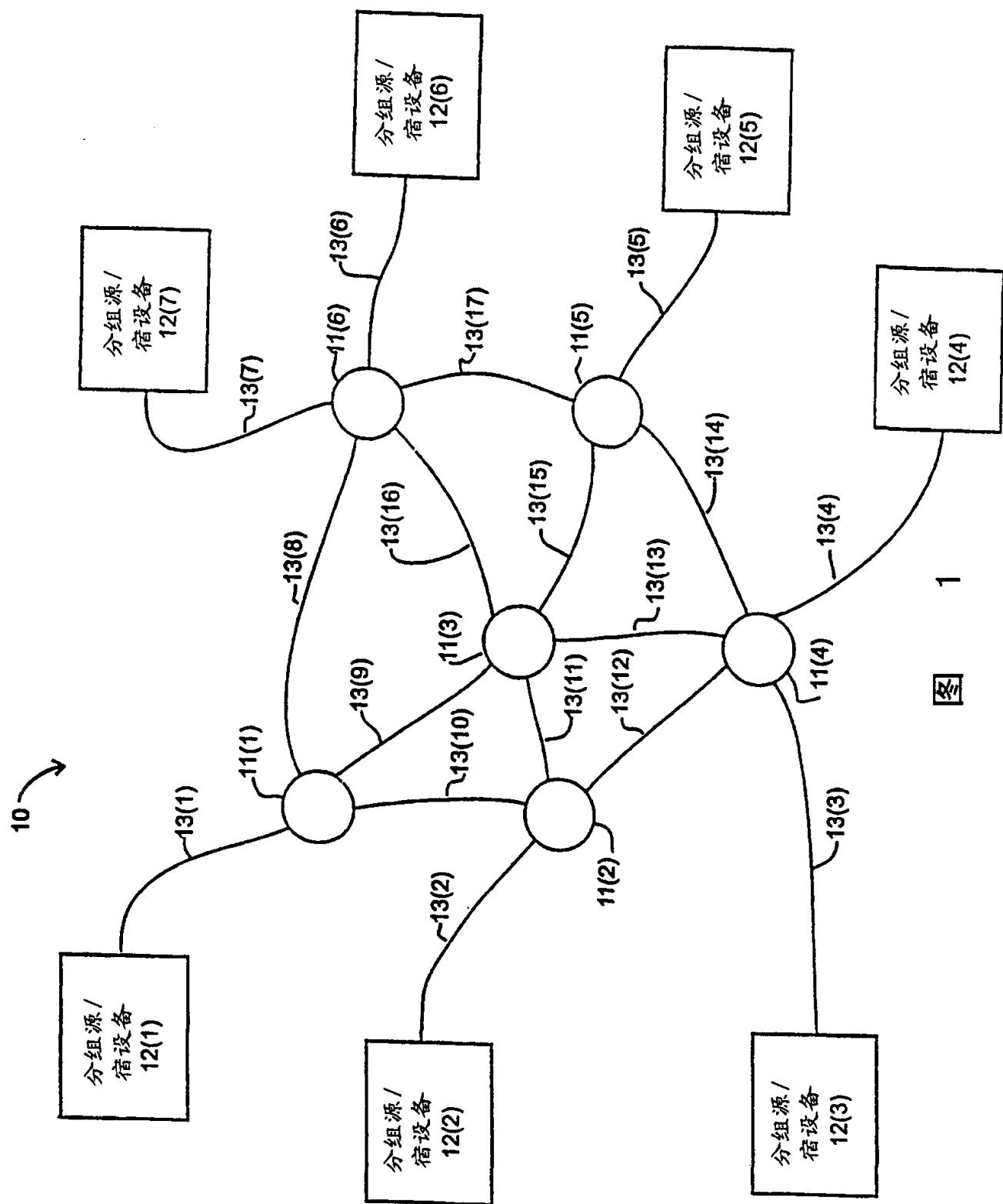
另外，交换节点11 (n) 尽管说明为利用SONET加入/丢掉多路复用器作为输入端口25 (n) (m) 和输出端口26 (n) (m)，但会理解，也可很方便地使用其他类型用于形成通信链路接口的器件。

将会理解，按照本发明的系统可全部或部分由专用硬件或通用计算机系统或其任意的组合来构成，其中的任何部分可以由合适的程序来控制。任何程序其全部或部分可按传统方式包括系统中一部分或存储在该系统上，或者，其部分或全部可通过网络或其他按常规方式传送信息的机制来提供给该系统。此外，将会理解，该系统可由操作员操作，和/或另外由操作员利用操作员输入单元（未图示）所提供的信息来控制，该操作员输入单元可以直接与系统连接，或可通过网络或其他按常规方式传送信息的机制将信息传送给该系统。

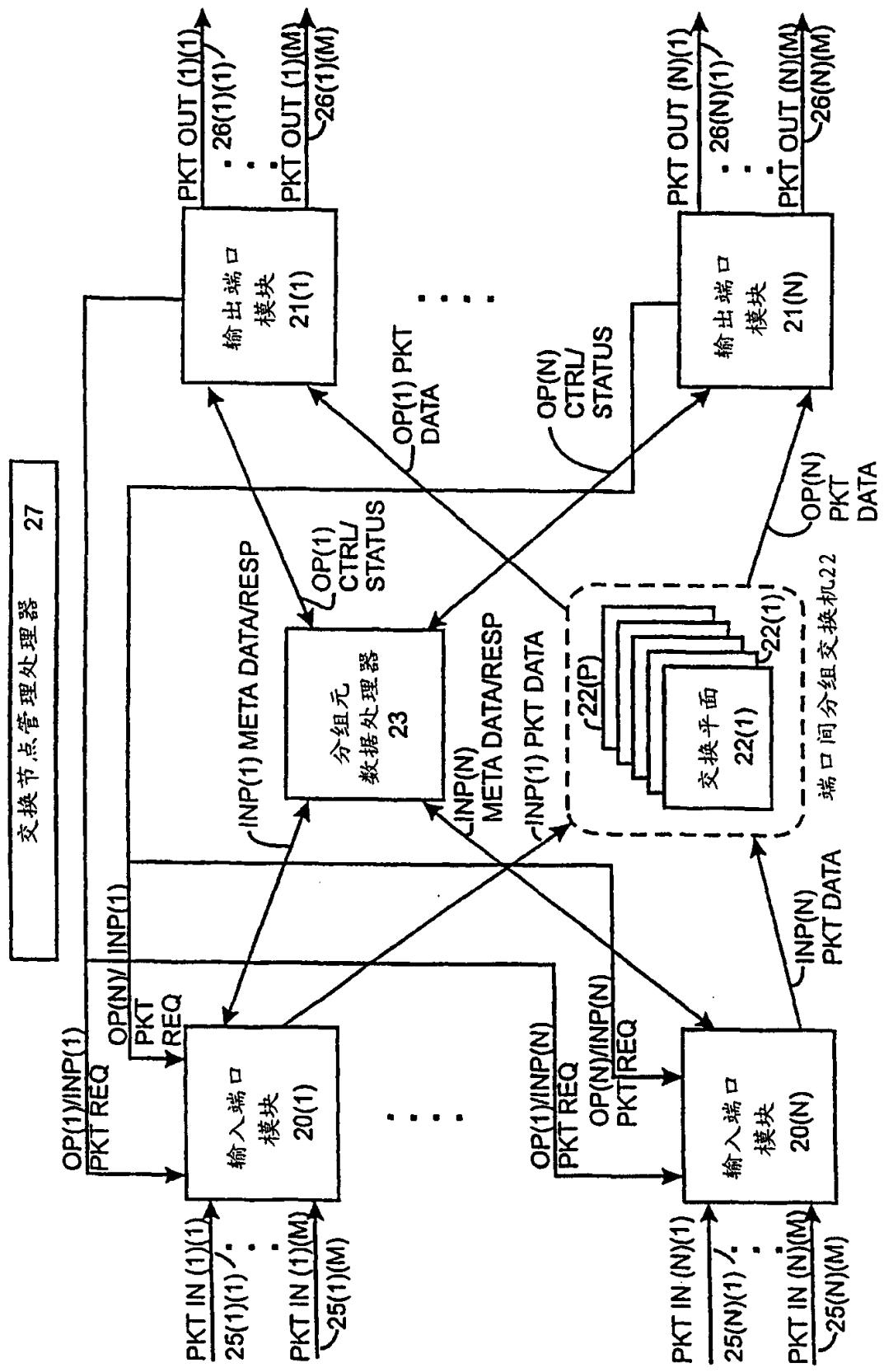
上述说明书所限定的是本发明一特定实施例。显然，可对本发明作种种变动和修改以实现本发明的某些或全部优点。所附权利要求，其目的在于涵盖在本发明实质和范围内的上述以及其他这类变动和修改。

要求作为新发明并期望得到美国专利证书保护的权利要求是：

说 明 书 附 图



交换节点管理处理器 27



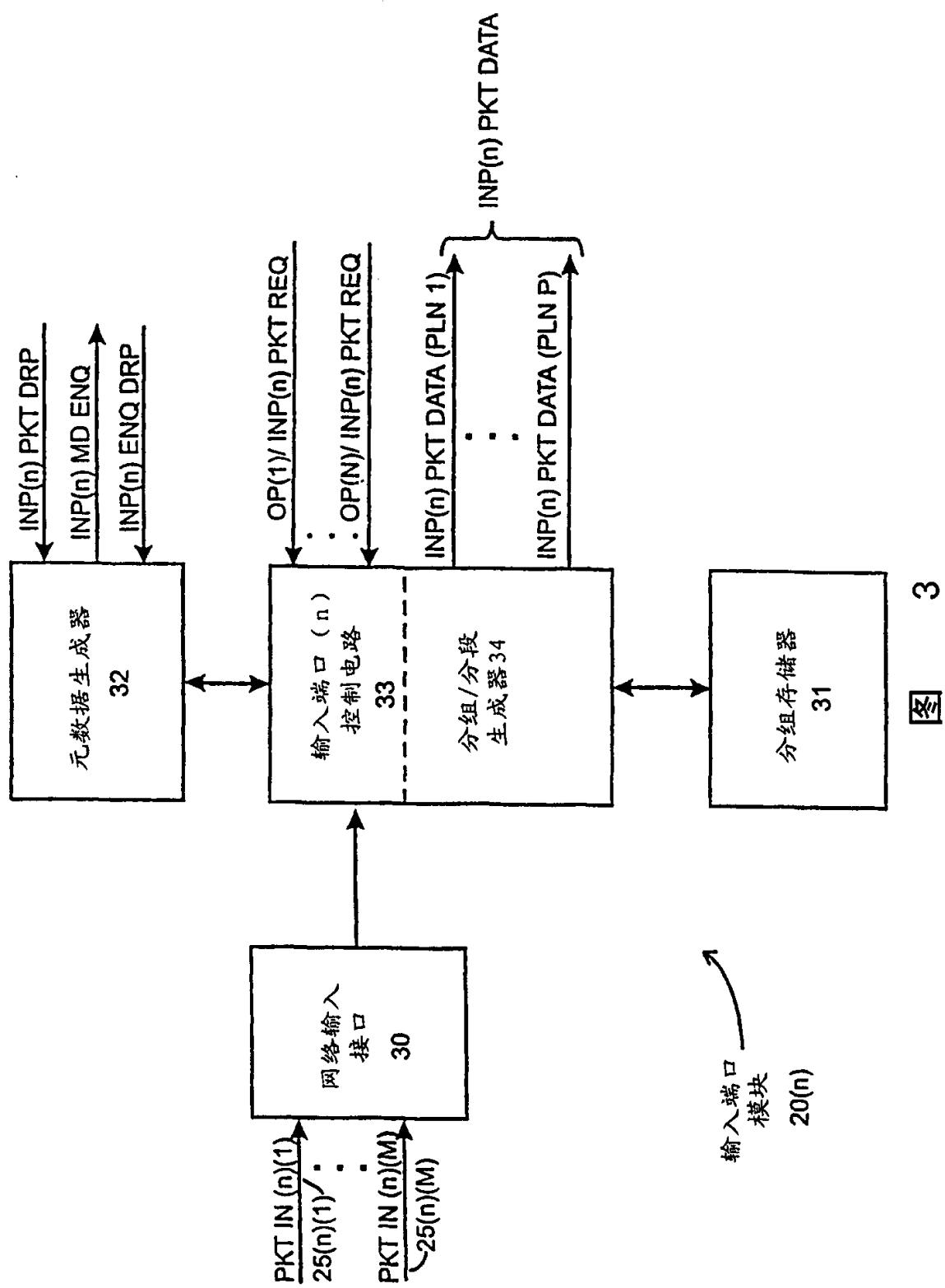
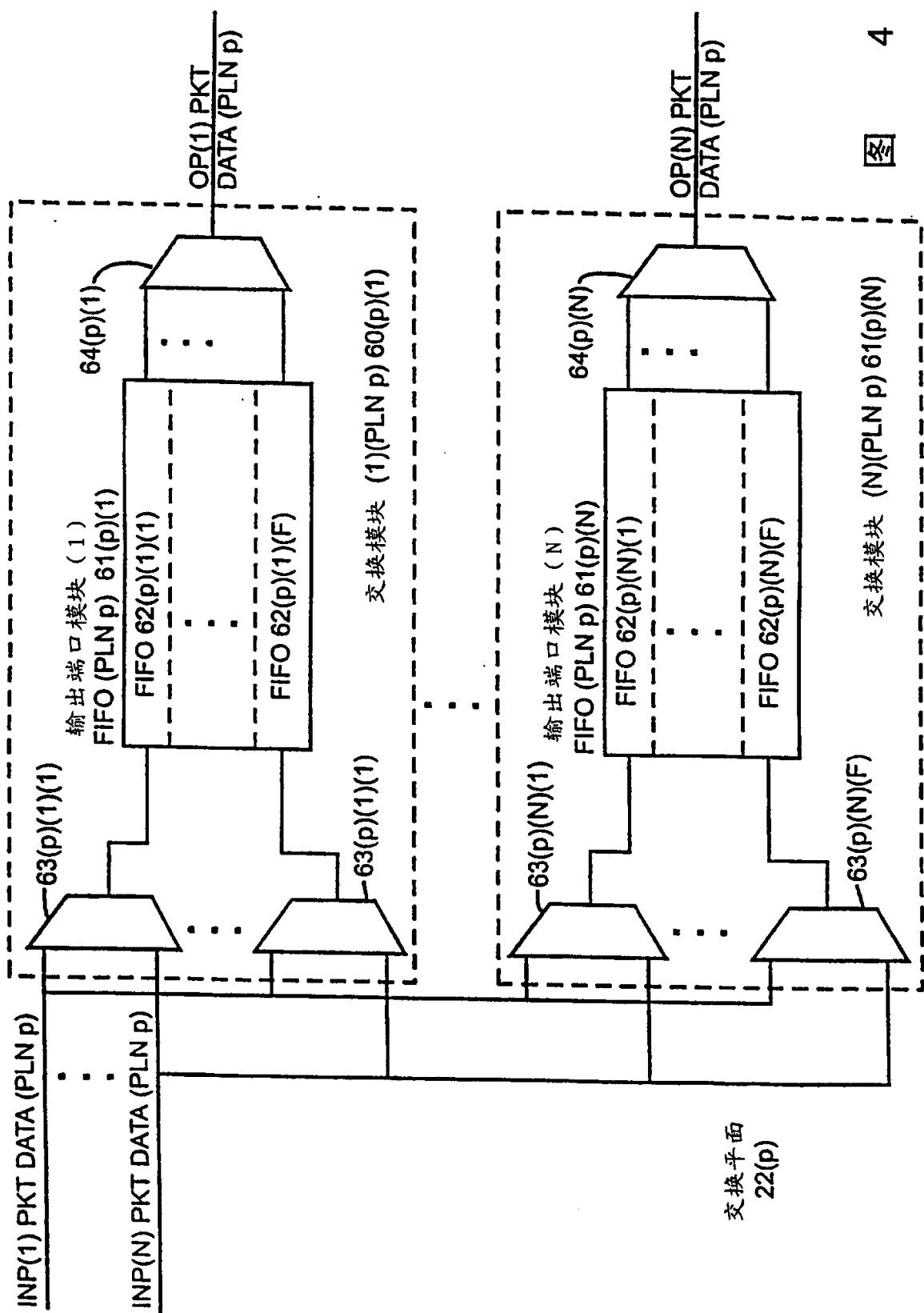
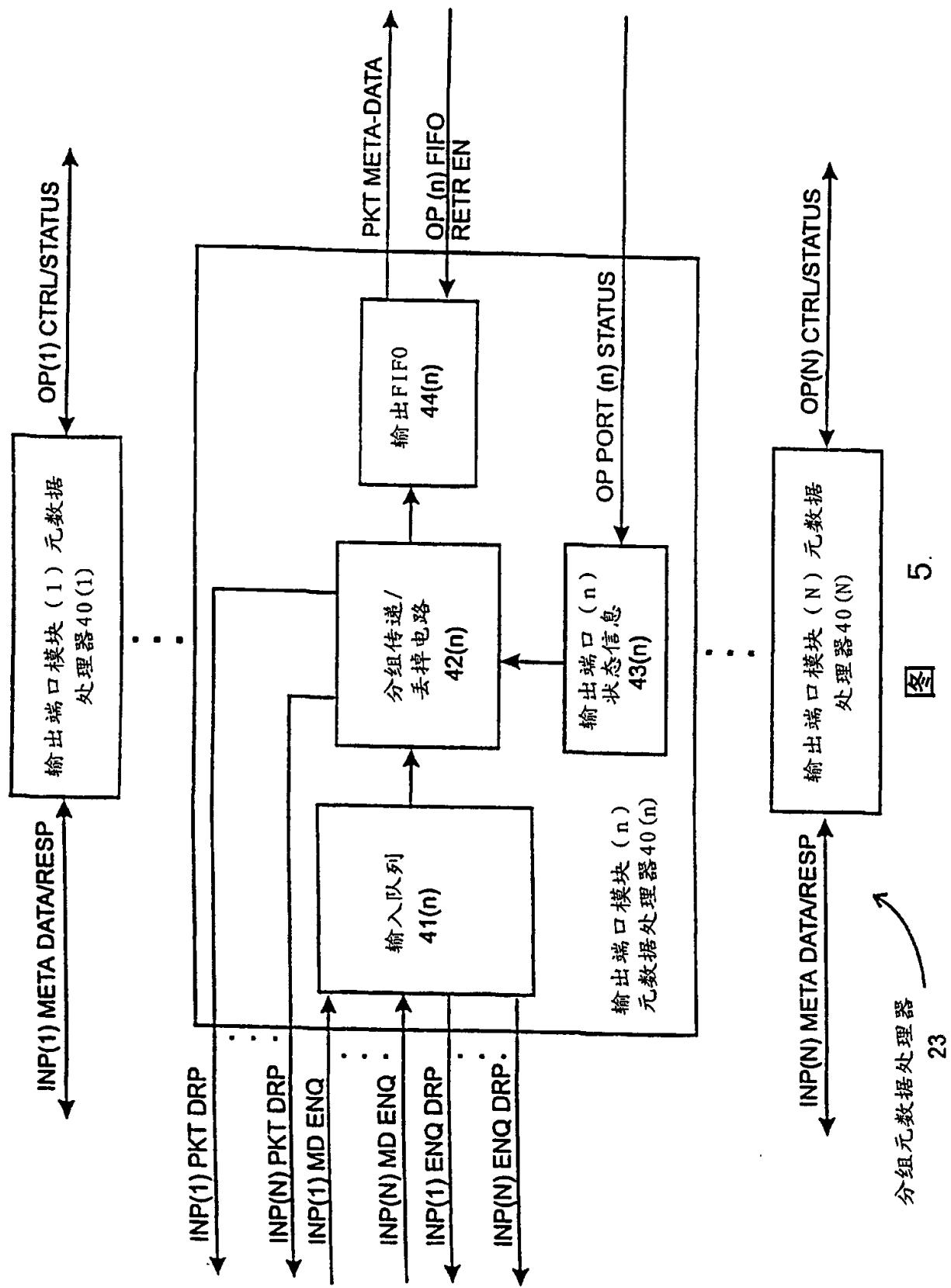


图 3





分组元数据处理器
23

图 5.

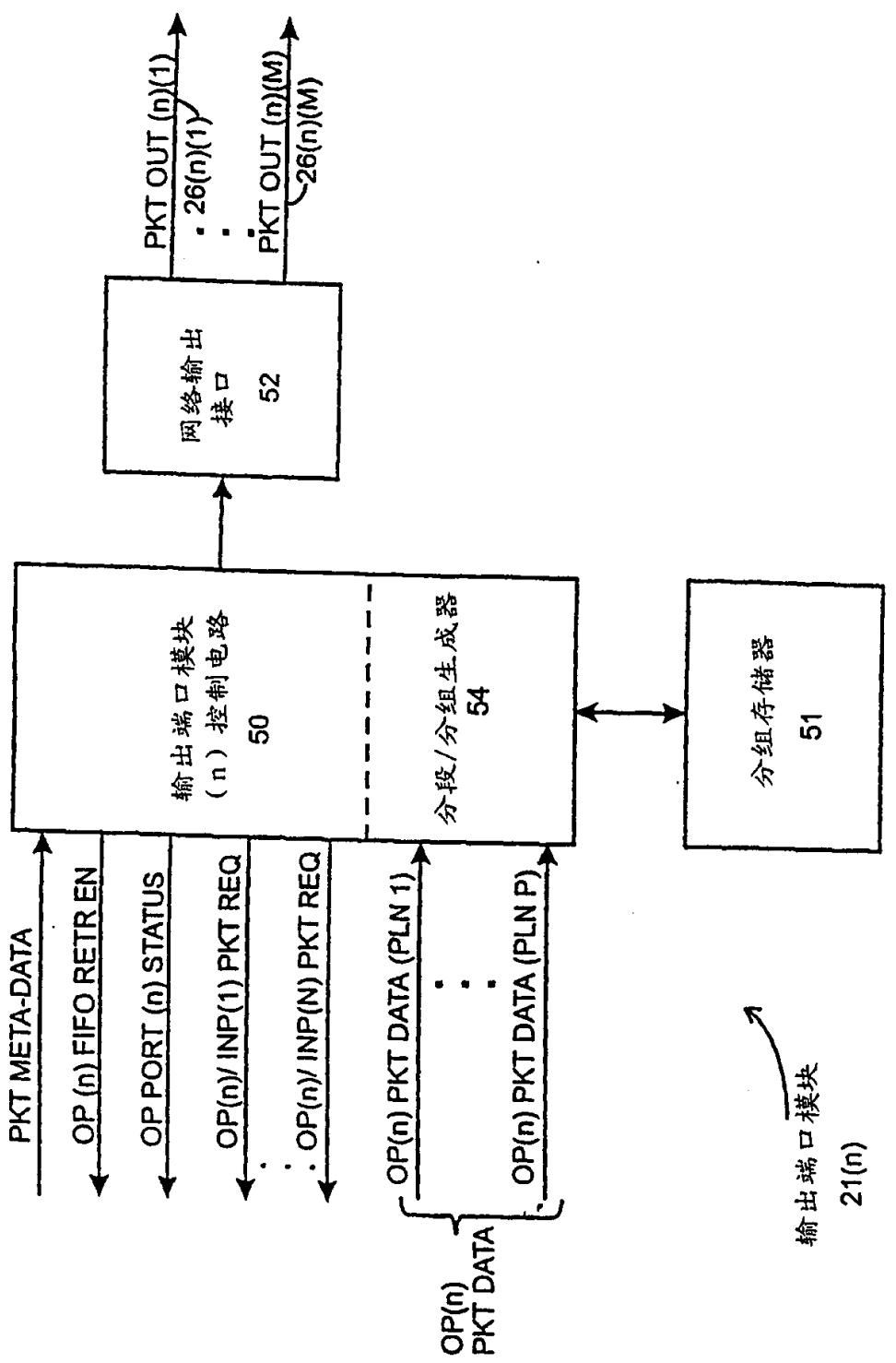
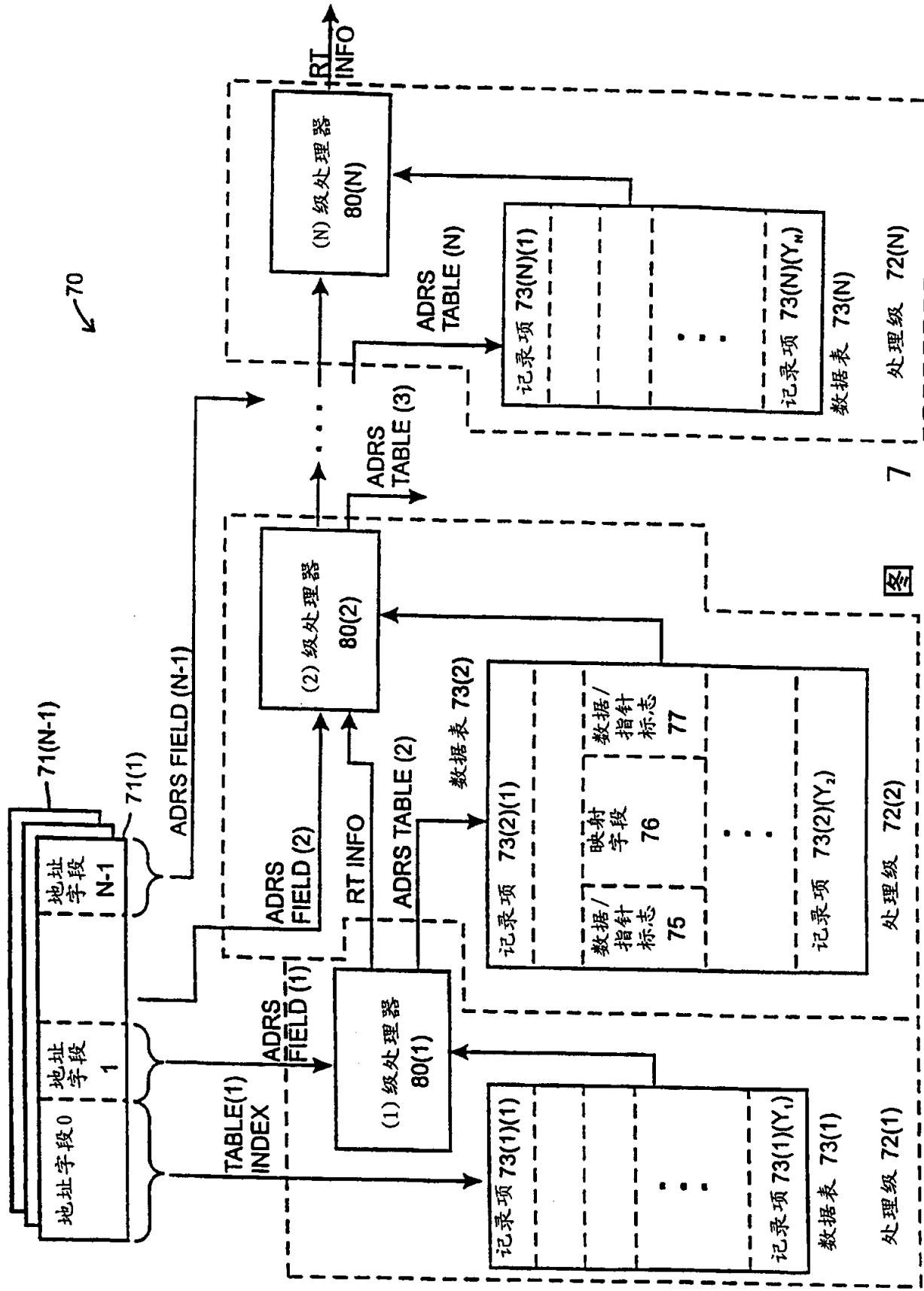
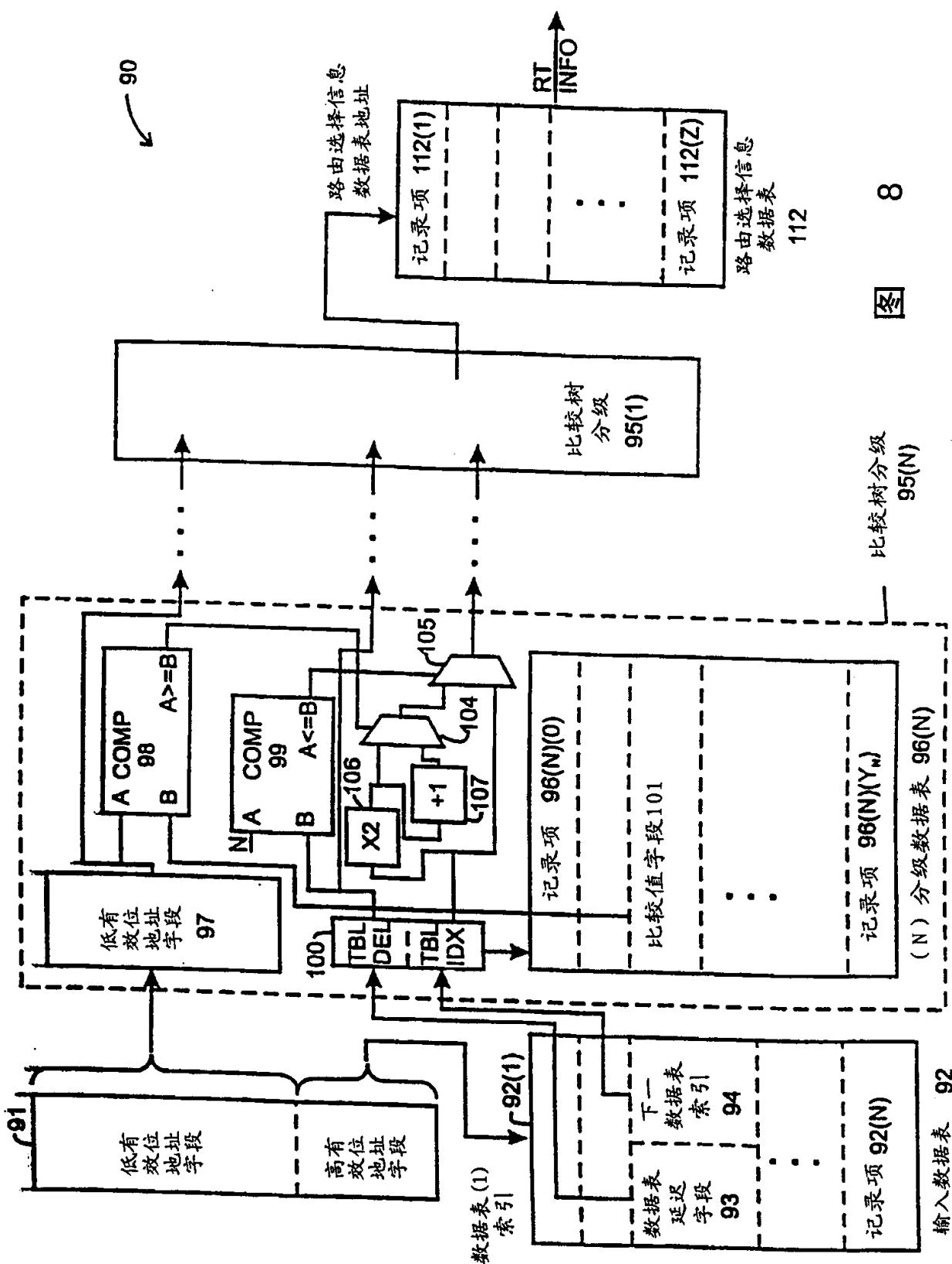


图 6





8

图

95(N)

(N) 分级数据表 95(N)

输入数据表 92