(19)中华人民共和国国家知识产权局

(12)发明专利

(10)授权公告号 CN 1856951 B
(45)授权公告日 2011.03.23

(21)申请号 200480026328.5

(22)申请日 2004.09.30

(30)优先权数据
10/676,390 2003.09.30 US

(85)PCT申请进入国家阶段日
2006.03.13

(86)PCT申请的申请数据
PCT/US2004/032555 2004.09.30

(87)PCT申请的公布数据
WO2005/034410 EN 2005.04.14

(73)专利权人 思科技术公司
地址 美国加利福尼亚州

(72)发明人 丹尼尔·C·比德尔曼 杨李卓

(74)专利代理机构 北京东方亿思知识产权代理有限责任公司 11258
代理人 王怡

(51)Int.Cl.
H04J 3/12(2006.01)
H04J 3/16(2006.01)
H04L 12/56(2006.01)
H04L 12/66(2006.01)
H03M 7/00(2006.01)
H03M 7/34(2006.01)
G06F 15/16(2006.01)

(56)对比文件
CN 1361958 A,2002.07.31, 全文.
WO 03010940 A2,2003.02.06, 说明书第5页第6段,第6页第2段, 第9页第3段,第12页第3段, 第13页第2－3段,附图1,4,8.

审查员 刘佳

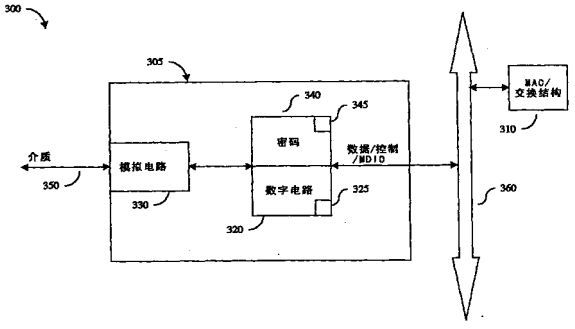
权利要求书 1 页 说明书 5 页 附图 5 页

(54)发明名称

将链路层安全性集成到物理层收发器中的方法和装置

(57)摘要

公开了用于在物理层收发器(PHY)中提供链路层安全性的装置。在一个实施例中,该装置可以包括被配置为与数据传输介质接口的模拟电路、被配置为与介质访问控制器(MAC)接口的数字电路以及耦合到该数字电路的密码引擎。



CN 1856951 B

1. 一种用于在物理层收发器 PHY 中提供链路层安全性的装置，包括：

被配置为向数据传输介质发送数据和从数据传输介质接收数据的模拟电路；

耦合到所述模拟电路的数字电路，所述数字电路被配置为向介质访问控制器 MAC 发送数据和从介质访问控制器 MAC 接收数据；以及

耦合到所述数字电路的密码引擎，该密码引擎被配置为对接收自所述 MAC 的数据进行加密以产生加密数据，其中

所述密码引擎还被配置为存储所述加密数据，并且

在发生分组冲突的情况下，所存储的加密数据被重新发送。

2. 如权利要求 1 所述的装置，其中所述密码引擎和所述 PHY 被放置在同一物理芯片上。

3. 如权利要求 2 所述的装置，其中所述密码引擎使用所述芯片上预先存在的硬件，所述硬件是为了实现所述 PHY 的功能而预先存在的。

4. 如权利要求 2 所述的装置，其中所述装置是多 PHY 设备的组件。

5. 如权利要求 2 所述的装置，其中所述 PHY 利用串行 PHY 介质接口通信。

6. 如权利要求 3 所述的装置，其中所述密码引擎重新使用 PHY 的引脚或接口布局、存储器映射、状态机的各个元素、逻辑门或上述的一个或多个。

7. 如权利要求 2 所述的装置，其中所述密码引擎还被配置为标记不合需要的数据以待丢弃。

8. 如权利要求 2 所述的装置，其中所述 MAC 包括 ASIC，该 ASIC 还被配置为交换结构。

9. 如权利要求 8 所述的装置，其中所述装置被放置在路由器内。

10. 如权利要求 1 所述的装置，其中所述密码引擎还被配置为执行数据压缩。

11. 一种在发送方 PHY 和接收方 PHY 之间提供链路层安全性的方法，所述方法包括：

通过所述发送方 PHY 接收来自第一 MAC 的数据；

通过所述发送方 PHY 对所述数据加密，以产生加密数据；

通过所述发送方 PHY 存储所述加密数据；

通过所述发送方 PHY 将所述加密数据发送到所述接收方 PHY；

确定是否发生分组冲突；

如果发生分组冲突，则重新发送已存储的加密数据；

通过所述接收方 PHY 接收所述加密数据；

通过所述接收方 PHY 对所述加密数据解密；以及

将解密后的数据提供到第二 MAC。

将链路层安全性集成到物理层收发器中的方法和装置

技术领域

[0001] 本发明一般地涉及链路层数据通信。

背景技术

[0002] 在本领域中已经知道，物理层收发器（“PHY”）用于通过各种介质（例如铜线缆和光缆）发送和接收数据。

[0003] 在接收模式中，PHY 充当从介质接收数据并将数据解码成适合于接收设备的形式的设备。在发送模式中，PHY 从设备（通常从介质访问控制器（“MAC”））获取数据，并将数据转换成适合于正在使用的介质的形式。

[0004] 图 1 是典型现有技术的 PHY 100 的功能框图。PHY 100 通常被配置为在主机设备的 MAC 110 和介质 120 之间充当接口。

[0005] PHY 100 通常包括模拟电路 130，该模拟电路 130 被配置成用于接收来自介质 120 的数据，并使用本领域已知技术将该数据解码成适合于主机设备的形式。PHY 100 还包括数字电路 140，该数字电路 140 被配置成用于接收来自 MAC 110 的数据，并将该数据转换成适合于介质 120 的形式。

[0006] PHY 100 还包括被配置为控制 PHY 的操作，尤其是数字电路 140 的操作的存储器和控制电路 150。存储器和控制电路 150 通常将包括用于通过总线接口 160 与 MAC110 接口的电路。非限制性示例包括介质独立接口（“MII”）、千兆位介质独立接口（“GMII”）、十千兆位介质独立接口（“XGMII”或“XAUI”）、缩减千兆位介质独立接口（RGMII）和串行千兆位介质独立接口（SGMII）。

发明内容

[0007] 根据本发明的一个方面，提供了一种用于在物理层收发器（PHY）中提供链路层安全性的装置，包括：被配置为向数据传输介质发送数据和从数据传输介质接收数据的模拟电路；耦合到所述模拟电路的数字电路，所述数字电路被配置为向介质访问控制器（MAC）发送数据和从介质访问控制器接收数据；以及耦合到所述数字电路的密码引擎，该密码引擎被配置为对接收自所述 MAC 的数据进行加密以产生加密数据，其中所述密码引擎还被配置为存储所述加密数据，并且在发生分组冲突的情况下，所存储的加密数据被重新发送。

[0008] 根据本发明的另一个方面，提供了一种在发送方 PHY 和接收方 PHY 之间提供链路层安全性的方法，所述方法包括：通过所述发送方 PHY 接收来自第一 MAC 的数据；通过所述发送方 PHY 对所述数据加密，以产生加密数据；通过所述发送方 PHY 存储所述加密数据；通过所述发送方 PHY 将所述加密数据发送到所述接收方 PHY；确定是否发生分组冲突；如果发生分组冲突，则重新发送已存储的加密数据；通过所述接收方 PHY 接收所述加密数据；通过所述接收方 PHY 对所述加密数据解密；以及将解密后的数据提供到第二 MAC。

[0009] 根据本发明的另一个方面，提供了一种用于在物理层收发器（PHY）中提供链路层安全性的装置，包括：用于接收来自第一 MAC 的数据的装置；用于对所述数据加密以产生

加密数据的装置；用于存储所述加密数据的装置；用于将所述加密数据发送到所述接收方PHY的装置；用于确定是否发生分组冲突的装置；用于如果发生分组冲突则重新发送已存储的加密数据的装置；用于接收所述加密数据的装置；用于对所述加密数据解密的装置；以及用于将解密后的数据提供到第二MAC的装置。

附图说明

[0010] 图1是现有技术PHY的概念性框图。

[0011] 图2是数据传输系统的概念性框图。

[0012] 图3是PHY的概念性框图。

[0013] 图4是用于提供链路层安全性的方法的流程图。

[0014] 图5是用于利用密码引擎(crypto engine)管理分组冲突的方法的流程图。

具体实施方式

[0015] 本领域普通技术人员将意识到，以下描述仅仅是示例性的，决不是限制性的。能受益于本公开的那些技术人员将容易想到其他修改和改良。在以下描述中，类似的标号始终指代类似的元件。

[0016] 本公开可能涉及数据通信。各种被公开的方面可被体现在各种计算机和机器可读数据结构中。此外，可以设想，可通过计算机和机器可读介质来传输体现本公开的教导的数据结构，并且可以通过利用诸如用于实现因特网的协议之类的标准协议以及其他计算机联网标准来经由通信系统传输体现本公开的教导的数据结构。

[0017] 本公开可能涉及存储有本公开的各个方面的机器可读介质。可以设想，适合于检索指令的任何介质都在本公开的范围内。例如，这样的介质可采取磁介质、光介质或半导体介质的形式，并可被配置为可由本领域已知的机器所访问。

[0018] 本公开的各个方面可通过使用流程图来描述。通常，本公开的一个方面的单个实例可被示出。但是，本领域普通技术人员将意识到，这里描述的协议、过程和程序可被连续重复或按所需频率重复，以满足这里所述的需求。因此，通过使用流程图对本公开各个方面的表示不应被用于限制本公开的范围。

[0019] 本公开在系统的链路层上提供了安全性。在这点上，链路层可根据OSI参考标准来定义。具体而言，I.E.E.E 802.3标准将链路层定义为存在于MAC和介质之间的设备，并且这里也如此定义。

[0020] 在本公开中，在发送模式中，通过在从MAC接收到数据时并且在将数据从PHY发送出去之前，针对保密性加密数据、针对完整性认证数据或既针对保密性加密数据又针对完整性认证数据，从而来提供链路层安全性。相反，在接收模式中，在由PHY接收到数据时并在将数据提供给MAC之前，数据被解密、认证或者既被解密又被认证。

[0021] 图2是根据本公开的教导配置的链路层数据传输系统205的图。系统205包括发送设备200，该发送设备通过介质240被耦合到接收设备260。

[0022] 发送设备200包括被用本领域已知技术配置成充当MAC的ASIC以及诸如图1所述那样的PHY 230。

[0023] 密码设备220被耦合在MAC 210和PHY 230之间。密码设备220优选地被配置为

利用 DES、3DES、MD5、SHA1、RC4 或 AES 或其它类似协议来加密 / 认证数据分组 250。

[0024] 在本示例中，数据分组由密码设备 220 从 MAC 210 接收到，并且在被提供到 PHY 230 并被发送到介质 240 上之前被加密 / 认证。

[0025] 系统 205 还包括与发送设备 200 类似配置的接收设备 260，其包括 MAC 270、密码设备 280 和 PHY 290。

[0026] 在接收设备中，加密后的数据分组 250 被 PHY 290 所接收并被提供到密码引擎 280，在密码引擎 280 中，数据被解密 / 认证并提供到 MAC270。

[0027] 当然，图 2 所公开的操作可以工作在相反路径中。

[0028] 图 3 是根据本公开的教导配置的 PHY 的另一实施例的概念性框图。

[0029] 图 3 的实施例规定，密码设备被部署在与 PHY 相同的芯片上，从而提供了单芯片链路层安全性解决方案。

[0030] 设备 300 包括 MAC 310 和 PHY 305。PHY 305 包括模拟电路 330，该模拟电路 330 在接收模式中被配置成用于接收来自介质 350 的数据，并利用本领域已知技术将该数据解码成适合于主机设备的形式。在发送模式中，该模拟电路被配置为接收来自 MAC 310 的数据，并将其转换成适合于介质 350 的形式。

[0031] PHY 305 还包括数字电路 320，该数字电路 320 在发送模式中被配置成用于接收来自 MAC 310 的数据，并将该数据转换成适合于介质 350 的形式，而在接收模式中被配置成用于接收来自模拟电路 330 的数据，并将其转换成适合于 MAC 310 的格式。

[0032] PHY 305 还包括被配置为控制 PHY 的操作，尤其是数字电路 320 的操作的存储器和控制电路 325。存储器和控制电路 325 通常将包括用于通过总线接口（例如 MII 或 GMII 或 XGMII 或 XAUI 或 SGMII 或 RGMII）与 MAC 310 接口的电路。

[0033] PHY 305 还包括耦合到数字电路 320 的密码模块 340。该密码模块可以包括用于密码功能操作的控制和存储器电路 345。密码模块 340 优选地被配置为在将接收自 MAC 310 的数据提供到模拟电路 330 之前对该数据进行加密 / 认证，并在将接收自模拟电路 330 的数据提供到 MAC 310 之前对该数据进行解密 / 认证。密码模块可以采用以上公开的密码技术。

[0034] 在另一实施例中，密码模块 340 可以利用已存在于 PHY 中的现有硬件来部署。将会意识到，通过重新使用已存在于 PHY 上的现有硬件来实现密码特征，可以大大节省设备中的不动产。

[0035] 可以设想，在实现所公开的密码特征时，可以重新使用大批 PHY 组件。例如，密码设备可以重新使用 PHY 的引脚或接口布局、存储器映射、状态机的各个元素、逻辑门或甚至上述的一个或多个。类似地，存在包含多个 PHY 的设备，例如包含 8 个 PHY 接口的 Octal PHY。在这些设备中，对已存在于 PHY 中的引脚和其他元件的重新使用可以减小裸芯和封装尺寸，从而使设备制造起来便宜得多。

[0036] 类似地，某些芯片并入了 MAC 作为 PHY 芯片的一部分。在此情况下，或许可以利用来自 MAC 和 PHY 两者的元件。

[0037] 还可以设想，由密码设备提供的附加功能可被用于其他功能或特征。例如，密码设备可被配置为执行数据压缩。

[0038] 例如，在一个实施例中，图 3 的设备 300 可以包括这样的路由器，在该路由器中，

MAC 310 包括被配置为还充当交换结构的 ASIC。在此情况下，在该设备中可以存在很多 PHY，并且通过交叉利用 PHY 的已有结构，可以在无需附加芯片的情况下添加附加的安全性特征。

[0039] 在被公开的另一实施例中，密码设备可被用于提高数据传输系统的整体性能和可靠性。

[0040] 如本领域普通技术人员将意识到的，很多这样的设备利用半双工模式工作，其中常见的性能问题是数据分组的冲突。

[0041] 可以设想，由密码设备提供的附加功能可以改善冲突管理。

[0042] 在本实施例中，密码存储器 345 可被用于在分组被发送时暂时存储数据和相关的安全性信息。如果检测到冲突，则可以立即重新使用并重新发送已存储的信息，而处理器或 MAC 无需重新发送数据，或发送诸如安全性关联之类的新安全性信息。

[0043] 正如能够受益于本公开的技术人员将意识到的，本公开的益处在于可以节省处理器周期的时间，并且还可以通过将某些处理时间从 ASIC 转移到 PHY 来改善性能。

[0044] 可以设想，密码设备可以利用 PHY 上的存储器的某些区域。如果 PHY 符合某些工业标准（例如 I.E.E.E 802.3），则 PHY 被提供有存储器中的某些为特定目的预留的寄存器，这种寄存器被称为 MII 管理接口。例如，寄存器 11-14 是预留的，而寄存器 16-30 是供应商专用区域。

[0045] 可以设想，可指导本公开中使用的安全性关联数据库 (SAD) 按预定顺序被写入某些区域。例如，寄存器 11 中的一个比特可被用于打开或关闭密码功能。类似地，密码技术可能需要诸如密钥或安全性关联之类的数据来执行密码功能。该数据可通过寄存器 12 来访问。这利用了已有的存储器管理技术和结构。当然，也可使用其他寄存器。

[0046] 本公开的另一益处是减少流量，这是因为 PHY 可被编程为丢弃或“废弃”接收到的没有通过解密模块的流量。在本示例中，没有被正确解密的数据被加注标记，以在被交换结构交换之前被随后的模块所丢弃，从而节省交换结构中的带宽以用于其他重要功能。这可以降低未经授权的用户由于拒绝服务攻击而使网络或联网设备崩溃的危险，从而增强了网络的可靠性。或者，安全性逻辑可以中断处理器以进行其他动作。

[0047] 图 4 是在数据传输系统的链路层处加密 / 认证数据的方法的流程图。在动作 400 中，想要进行通信的 PHY 可以利用本领域已知的技术来自动协商链路。应该理解，也可以在链路的自动协商之前应用这里公开的加密 / 认证技术。

[0048] 在动作 410 中，发送方 PHY(transmitting PHY，即“TX PHY”) 的 MAC 将想要发送的数据提供到密码引擎。在动作 420 中，数据被密码引擎加密，并被 TX PHY 放置在链接 PHY 的介质上。

[0049] 在动作 430 中，接收方 PHY(receiving PHY，即“RCV PHY”) 接收来自链路的密码数据，并将数据提供给 RCV PHY 的密码引擎，在密码引擎中，数据被解密、认证或者既被解密又被认证。

[0050] 在动作 440 中，无格式数据随后被传递到 RCV PHY 的 MAC。

[0051] 图 5 是利用密码引擎管理分组冲突的方法的流程图。

[0052] 在动作 500 中，TX PHY 的 MAC 将想要发送的数据提供到密码引擎。在动作 520 中，数据被密码引擎加密、认证或既被加密又被认证，并被 TX PHY 放置在链接 PHY 的介质上。如

上所述，想要通信的 PHY 可以利用本领域已知的技术来自动协商链路，但是数据也可以在链路的自动协商前被加密。此时，经加密 / 认证的数据由密码引擎所存储。

[0053] 在查询 530 中，PHY 确定是否已经发生分组冲突。如果已经发生冲突，则由 TX PHY 重新发送已存储的分组。如果没有发生冲突，通信过程则如正常情况那样前进，并且已存储的任何数据可被冲掉，或者已使用的空间被回收。

[0054] 虽然已示出和描述了本公开的实施例和应用，但是对于本领域技术人员显而易见的是，在不脱离本发明的概念的情况下，可以对上述内容执行更多修改和改良。因此，除了在所附权利要求书的精神内之外，本公开将不受限制。

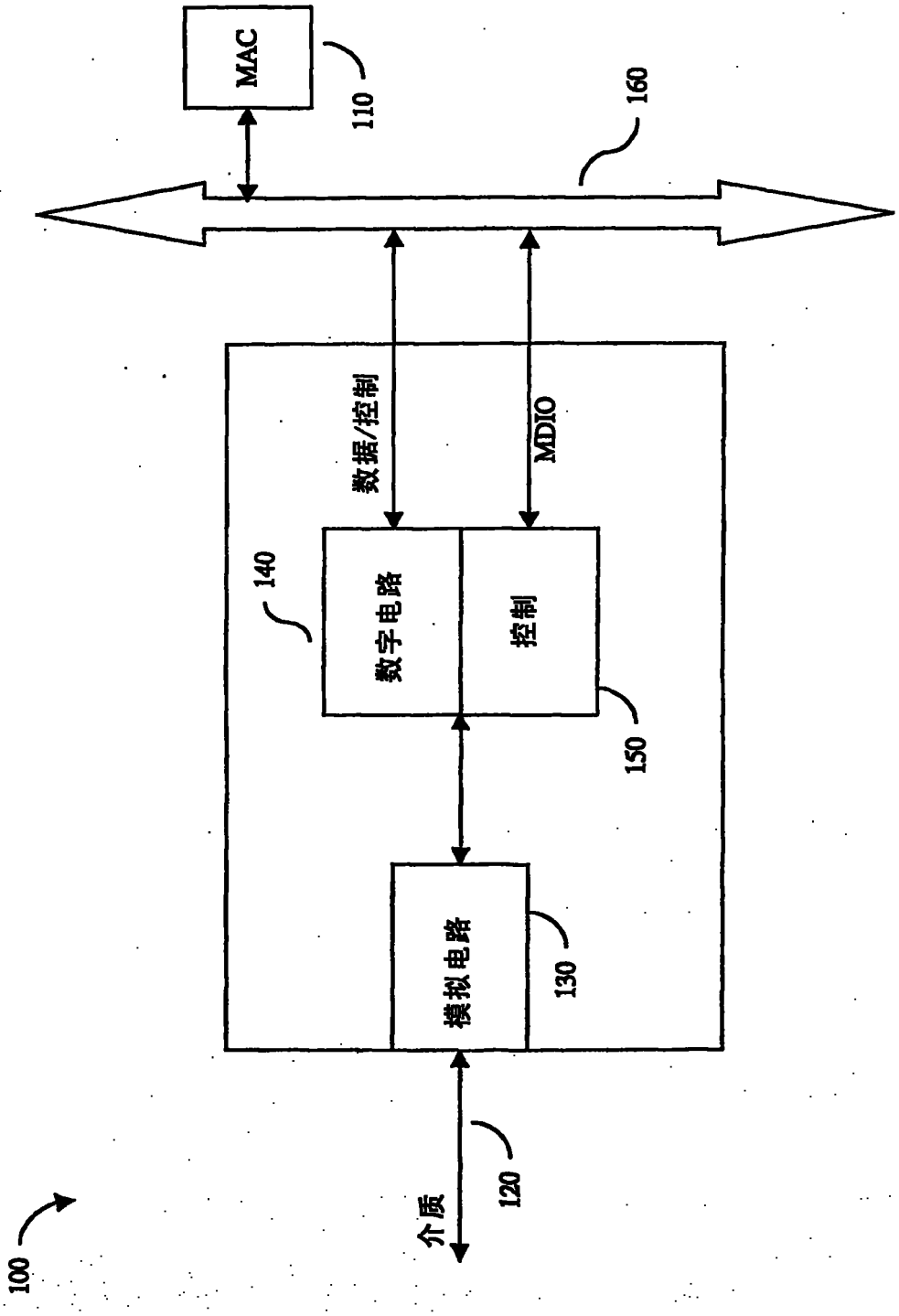


图1

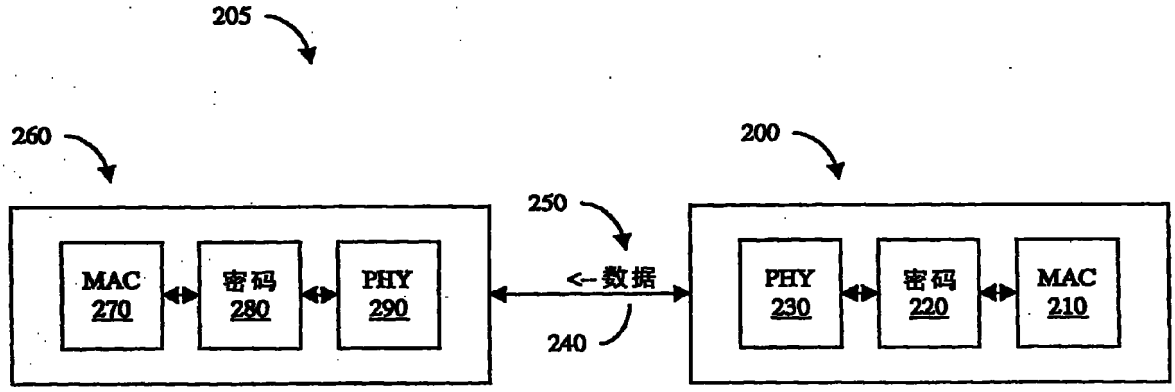

205
260
200
250
MAC
270
密码
280
PHY
290
<--数据
240
PHY
230
密码
220
MAC
210

图2

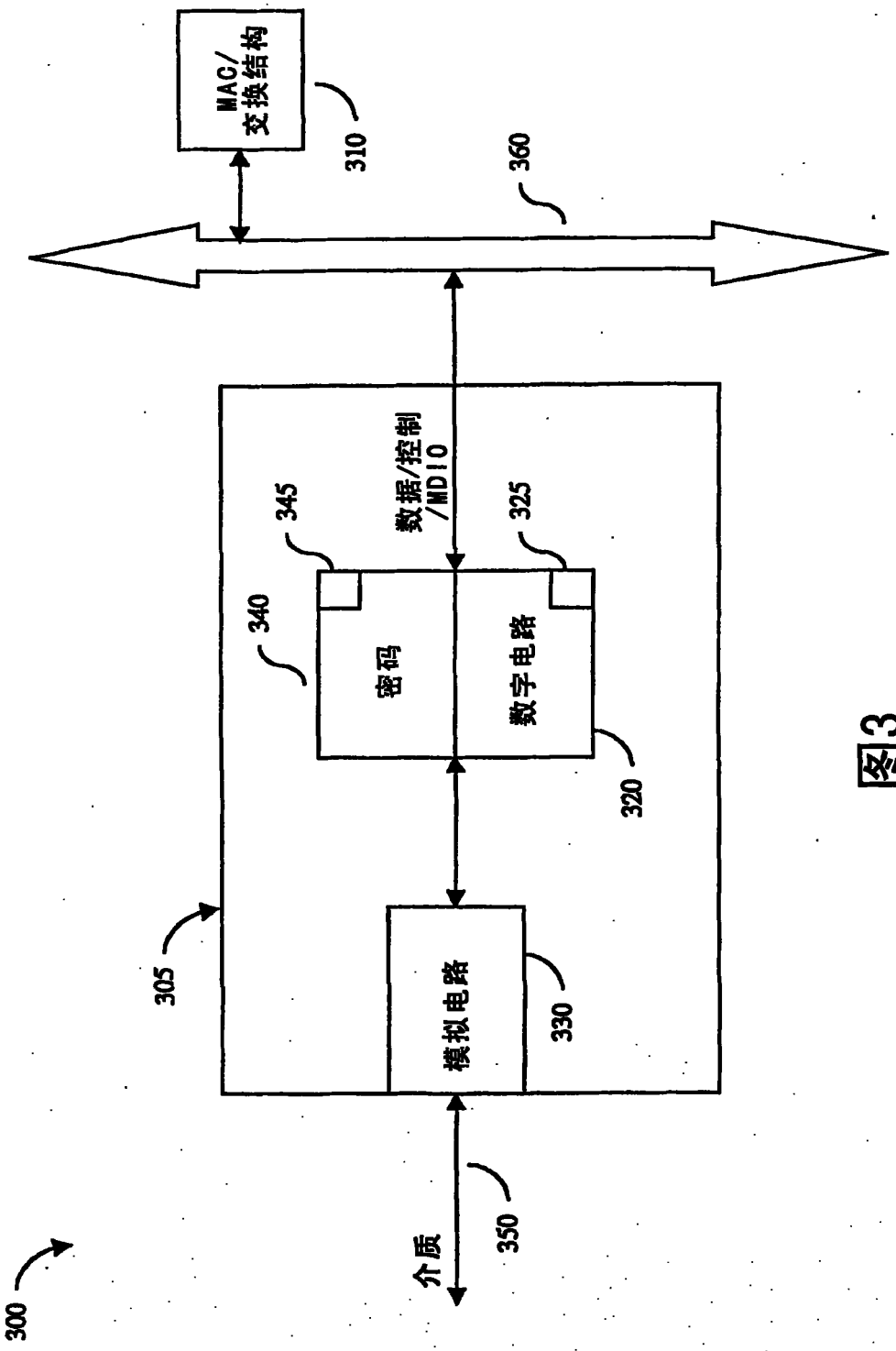

MAC/
交换结构
310
360
数据/控制
/MDIO
345
325
340
密码
数字电路
320
305
模拟电路
330
介质
350
300

图3

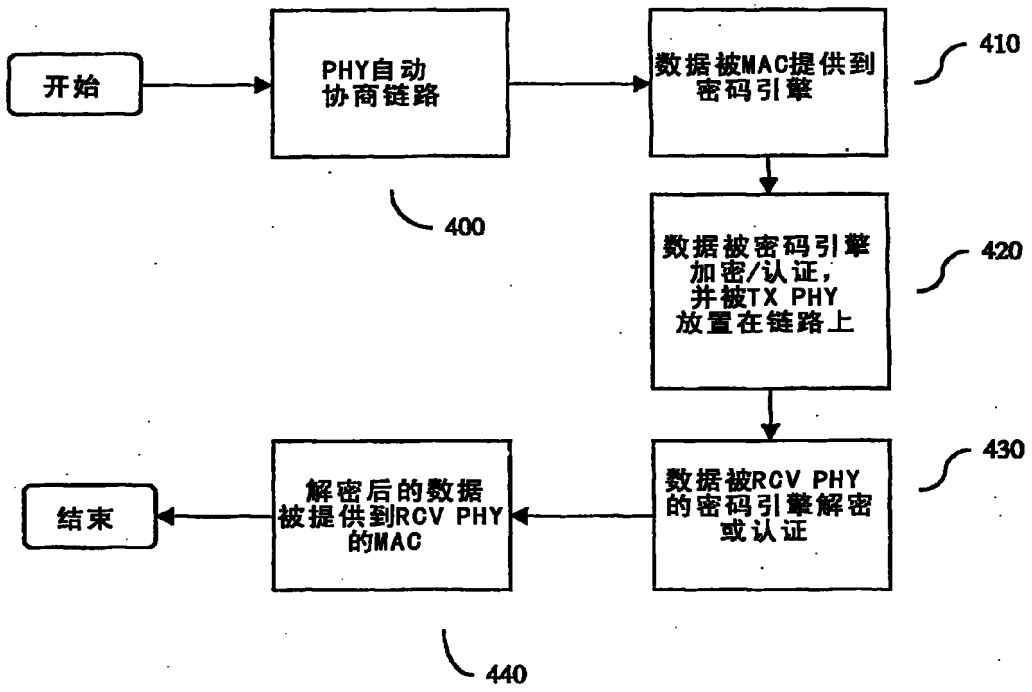

开始
PHY自动
协商链路
400
数据被MAC提供到
密码引擎
410
数据被密码引擎
加密/认证，
并被TX PHY
放置在链路上
420
数据被RCV PHY
的密码引擎解密
或认证
430
解密后的数据
被提供到RCV PHY
的MAC
440
结束

图4

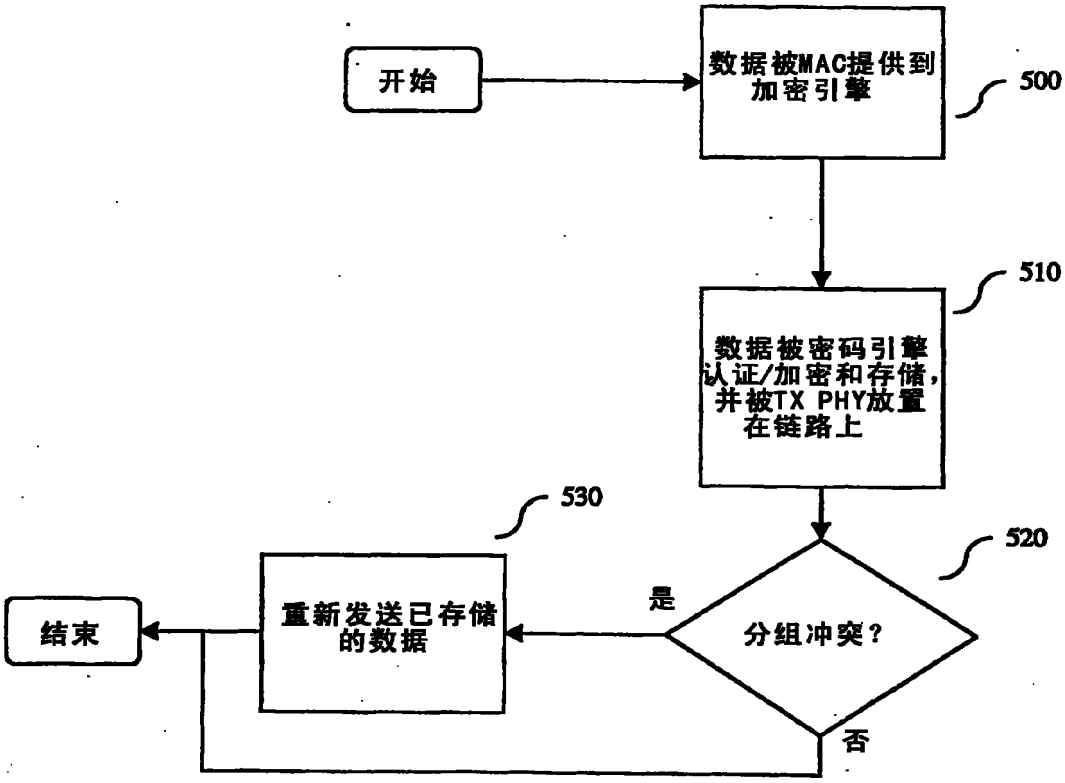

开始
数据被MAC提供到加密引擎
500
数据被密码引擎认证/加密和存储，并被TX PHY放置在链路上
510
分组冲突？
520
是
否
重新发送已存储的数据
530
结束

图5