

码字同步方法、接收器、网络设备及网络系统

本申请要求于 2020 年 4 月 10 日提交的申请号为 202010280832.9、发明名称为“一种码字同步方法、接收器及网络设备”的中国专利申请以及于 2020 年 5 月 19 日提交的申请号为 202010424884.9、发明名称为“码字同步方法、接收器、网络设备及网络系统”的中国专利申请的优先权，其全部内容通过引用结合在本申请中。

技术领域

本申请涉及通信领域，尤其涉及一种码字同步方法、接收器、网络设备及网络系统。

背景技术

随着生产工艺水平的提高，信道的损耗和噪声已成为限制数据传输速率和距离的关键因素。前向纠错（Forward Error Correction）的出现，为传输中的数据提供了纠错保护，从而提高了信道的数据传输速率以及传输距离。FEC 按照对信息序列的处理方式不同可分为分组码（block code）和卷积码（convolutional code）。对于分组码，可再细分为线性分组码（linear block code）与非线性分组码（non-linear block code）。线性分组码由于其编译码实现较为简单而在以太网的开放式系统互联模型（Open System Interconnection Model, OSI）中的物理层（physical layer）以及数据链路层（data link layer）的介质访问控制（media access control）子层中得到广泛应用。

线性分组码的检错及纠错功能实现必须基于完整的码字（codeword）进行，因而需要在数据流中确定码字边界，即找到一个完整码字的开始和结尾，这一过程称为码字同步（codeword synchronization）或者帧同步（frame synchronization）。

目前业界已经有适用于线性分组码的同步方案。以 802.3 标准内 200/400 GE 中使用的对齐标志（alignment marker, AM）同步方案为例，该方案中每间隔一定长度的码字均须插入一段固定的 AM 序列，接收端识别该 AM 序列即可进行码字同步。然而，AM 序列的存在相当于在发送端数据流中插入了额外数据，增加了冗余信息。

发明内容

本申请提供了一种自同步的码字同步方法、接收器及网络设备，用于解决 AM 同步方案中增加额外数据的技术问题。

第一方面，本申请提供了一种码字同步方法。所述方法包括：步骤一，接收数据序列，所述数据序列包括多个比特；步骤二，确定所述数据序列中的备选比特，所述备选比特包括在所述多个比特中；步骤三，根据所述备选比特确定同步位置，所述同步位置用于指示所述数据序列中码字的起始位置。

该方法由网络中的接收设备执行。通过该方法无须在发送端数据流中插入额外数据，即可实现接收端数据流高精度码字同步的技术效果，其同步性能达到较高的可靠性。

在一种可能的实现方式中，所述步骤三包括：对所述备选比特进行验证，当验证成功

时, 确定所述备选比特的位置为所述同步位置。

在一种可能的实现方式中, 根据所述备选比特在所述数据序列中划分出至少一个第二测试数据块, 所述备选比特的位置为所述至少一个第二测试数据块的起始位置; 对所述至少一个第二测试数据块的特性值进行验证, 当验证成功时, 确定所述备选比特的位置为所述同步位置。

在一种可能的实现方式中, 对所述至少一个第二测试数据块的特性值进行验证, 包括: 依次累计所述至少一个第二测试数据块中各个第二测试数据块的特性值以获得累计值, 直至所述累计值满足同步条件时, 验证成功。

在一种可能的实现方式中, 所述特性值的累计值为所述至少一个第二测试数据块中被判定为正确码字的第二测试数据块的数量, 所述同步条件为所述累计值大于或等于同步阈值; 所述特性值为校验序列中全零序列的数量, 所述同步条件为所述累计值大于或等于同步阈值; 或者, 所述特性值为校验序列中零元素的数量, 所述同步条件为所述累计值大于或等于同步阈值; 或者, 所述特性值为可纠错测试数据块的数量, 所述同步条件为所述累计值大于或等于同步阈值; 或者, 所述特性值为重检验位与原始校验位相同的测试数据块的数量, 所述同步条件为所述累计值大于或等于同步阈值; 其中, 所述测试数据块的长度为 n 比特, 所述测试数据块的前 k 比特为信息位, 所述测试数据块的后 $n-k$ 比特为所述原始校验位, 所述重检验位是根据所述信息位获得的, 所述重校验位的长度为 $n-k$ 比特, 所述 n 和所述 k 为整数。

在一种可能的实现方式中, 对所述至少一个第二测试数据块的特性值进行验证, 包括: 将所述至少一个第二测试数据块中所有第二测试数据块的特性值相加以获得总值, 当所述总值满足同步条件时, 验证成功。

在一种可能的实现方式中, 所述特性值的总值为所述至少一个第二测试数据块中被判定为正确码字的第二测试数据块的数量, 所述特性值的总值为所述至少一个第二测试数据块中被判定为错误码字的第二测试数据块的数量, 所述同步条件为所述总值小于或等于同步阈值; 或者, 所述特性值为校验序列中全零序列的数量, 所述同步条件为所述总值大于或等于同步阈值; 或者, 所述特性值为校验序列中非全零序列的数量, 所述同步条件为所述总值小于或等于同步阈值; 或者, 所述特性值为校验序列中零元素的数量, 所述同步条件为所述总值大于或等于同步阈值; 或者, 所述特性值为校验序列中非零元素的数量, 所述同步条件为所述总值小于或等于同步阈值; 或者, 所述特性值为可纠错测试数据块的数量, 所述同步条件为所述总值大于或等于同步阈值; 或者, 所述特性值为不可纠错测试数据块的数量, 所述同步条件为所述总值小于或等于同步阈值; 或者, 所述特性值为重检验位与原始校验位相同的测试数据块的数量, 所述同步条件为所述总值大于或等于同步阈值; 或者, 所述特性值为重检验位与原始校验位不相同的测试数据块的数量, 所述同步条件为所述总值小于或等于同步阈值; 其中, 所述测试数据块的长度为 n 比特, 所述测试数据块的前 k 比特为信息位, 所述测试数据块的后 $n-k$ 比特为所述原始校验位, 所述重检验位是根据所述信息位获得的, 所述重校验位的长度为 $n-k$ 比特, 所述 n 和所述 k 为整数。

在一种可能的实现方式中, 将所述备选比特的位置作为所述同步位置。

在一种可能的实现方式中, 从所述数据序列中选取多个观察比特; 从所述多个观察比特中选取所述备选比特。

在一种可能的实现方式中，所述从所述多个观察比特中选取所述备选比特，包括：根据所述多个观察比特从所述数据序列中确定多组第一测试数据块，其中，所述多组第一测试数据块中的每组第一测试数据块包括至少一个第一测试数据块，所述多个观察比特中的每一个观察比特的位置为所述多组第一测试数据块中的每组第一测试数据块的起始位置；

5 根据所述多组第一测试数据块的特性值，从所述多个观察比特中选取一个观察比特作为所述备选比特。

在一种可能的实现方式中，所述根据所述多组第一测试数据块的特性值，从所述多个观察比特中选取一个观察比特作为所述备选比特，包括：依次判断所述多组第一测试数据块中各组第一测试数据块的特性值是否满足备选条件，直至确定出一组第一测试数据块的特性值满足所述备选条件；将满足所述备选条件的所述一组第一测试数据块对应的观察比特作为所述备选比特。

10

在一种可能的实现方式中，所述一组第一测试数据块的特性值为所述一组第一测试数据块中所有第一测试数据块的校验序列中全零序列的数量总值，所述备选条件为所述总值大于或等于同步阈值；或者，所述一组第一测试数据块的特性值为所述一组第一测试数据块中所有第一测试数据块的校验序列中非全零序列的数量总值，所述备选条件为所述总值小于或等于同步阈值；或者，所述一组第一测试数据块的特性值为所述一组第一测试数据块中所有第一测试数据块的校验序列中零元素的数量总值，所述备选条件为所述总值大于或等于同步阈值；或者，所述一组第一测试数据块的特性值为所述一组第一测试数据块中所有第一测试数据块的校验序列中非零元素的数量总值，所述备选条件为所述总值小于或等于同步阈值；或者，所述一组第一测试数据块的特性值为所述一组第一测试数据块中所有第一测试数据块中可纠错测试数据块的数量总值，所述备选条件为所述总值大于或等于同步阈值；或者，所述一组第一测试数据块的特性值为所述一组第一测试数据块中所有第一测试数据块中不可纠错测试数据块的数量总值，所述备选条件为所述总值小于或等于同步阈值；或者，所述一组第一测试数据块的特性值为所述一组第一测试数据块中所有第一测试数据块中重检验位与原始校验位相同的测试数据块的数量总值，所述备选条件为所述总值大于或等于同步阈值；或者，所述一组第一测试数据块的特性值为所述一组第一测试数据块中所有第一测试数据块中重检验位与原始校验位不相同的测试数据块的数量总值，所述备选条件为所述总值小于或等于同步阈值；或者，所述一组第一测试数据块的特性值为所述一组第一测试数据块中前 X 个第一测试数据块的校验序列中全零序列的数量累计值，所述备选条件为所述累计值大于或等于同步阈值；或者，所述一组第一测试数据块的特性值为所述一组第一测试数据块中前 X 个第一测试数据块的校验序列中零元素的数量累计值，所述备选条件为所述累计值大于或等于同步阈值；或者，所述一组第一测试数据块的特性值为所述一组第一测试数据块中前 X 个第一测试数据块中可纠错测试数据块的数量累计值，所述备选条件为所述累计值大于或等于同步阈值；或者，所述一组第一测试数据块的特性值为所述一组第一测试数据块中前 X 个第一测试数据块中重检验位与原始校验位相同的测试数据块的数量累计值，所述备选条件为所述累计值大于或等于同步阈值；其中，所述测试数据块的长度为 n 比特，所述测试数据块的前 k 比特为信息位，所述测试数据块的后 n-k 比特为所述原始校验位，所述重检验位是根据所述信息位获得的，所述重校验位的长度为 n-k 比特，所述 n，所述 k 和所述 X 为整数。

15

20

25

30

35

在一种可能的实现方式中,所述根据所述多组第一测试数据块的特性值,从所述多个观察比特中选取一个观察比特作为所述备选比特,包括:比较所述多组第一测试数据块中每组第一测试数据块的特性值,将所述特性值为极值的一组第一测试数据块对应的观察比特作为所述备选比特。

- 5 在一种可能的实现方式中,所述特性值为校验序列中全零序列的数量,所述极值为最大值;或者,所述特性值为校验序列中非全零序列的数量,所述极值为最小值;或者,所述特性值为校验序列中零元素的数量,所述极值为最大值;或者,所述特性值为校验序列中非零元素的数量,所述极值为最小值;或者,所述特性值为可纠错测试数据块的数量,所述极值为最大值;或者,所述特性值为不可纠错测试数据块的数量,所述极值为最小值;
- 10 或者,所述特性值为重检验位与原始校验位相同的测试数据块的数量,所述极值为最大值;或者,所述特性值为重检验位与原始校验位不相同的测试数据块的数量,所述极值为最小值;其中,所述测试数据块的长度为 n 比特,所述测试数据块的前 k 比特为信息位,所述测试数据块的后 $n-k$ 比特为所述原始校验位,所述重检验位是根据所述信息位获得的,所述重校验位的长度为 $n-k$ 比特,所述 n 和所述 k 为整数。

- 15 在一种可能的实现方式中,所述从所述数据序列中选取多个观察比特,包括:从所述数据序列中每间隔 T 个比特选取一个比特,作为所述观察比特,所述 T 为大于零的整数;或者,从所述数据序列中每间隔 $L*n+T$ 个比特选取一个比特,作为所述观察比特,所述 L 为间隔的测试数据块的数量,所述测试数据块的长度为所述 n 个比特,所述 L 和所述 T 为大于零的整数。

- 20 在一种可能的实现方式中,所述数据序列为调制信号,所述数据序列包括多个调制码元,其中,所述从所述第一数据中选取多个观察比特,包括:从所述数据序列中每间隔 T 个调制码元,选取调制码元的起始比特,作为所述观察比特,所述 T 为大于零的整数;或者,从所述数据序列中每间隔 $L*m+T$ 个调制码元,选取调制码元的起始比特,作为所述观察比特,所述 L 为间隔的测试数据块的数量,所述测试数据块的长度为所述 m 个调制
- 25 码元,所述 L 和所述 T 为大于零的整数。

在一种可能的实现方式中,所述观察比特的数量为 P ,所述 P 为正整数,所述码字的长度为所述 P 个比特。

- 在一种可能的实现方式中,所述步骤二中,所述确定所述数据序列中的备选比特包括,确定第一子序列中的备选比特,所述步骤三中,所述根据所述备选比特在所述数据序列中
- 30 划分出至少一个第二测试数据块包括,根据所述备选比特在第二子序列中划分出至少一个第二测试数据块,所述第一子序列和所述第二子序列包括在所述数据序列中,所述第二子序列与所述第一子序列相同、部分相同或不同。

在一种可能的实现方式中,在所述步骤三之后,所述方法还包括:步骤四,响应于所述数据序列为失锁状态,确定所述数据序列的更新同步位置。

- 35 在一种可能的实现方式中,所述确定所述数据序列的更新同步位置,包括:重新执行所述步骤二和所述步骤三,将重新执行的步骤三所确定的同步位置作为所述更新同步位置。

在一种可能的实现方式中,在所述步骤三之后、所述步骤四之前,所述方法还包括:根据所述同步位置从所述数据序列中划分出多个同步码字,所述同步位置为所述多个同步码字的起始位置;对所述多个同步码字的特性值进行验证,当验证失败时,确定所述数据

序列为所述失锁状态。

在一种可能的实现方式中，对所述多个同步码字的特性值进行验证，包括：依次累计所述多个同步码字中各个同步码字的特性值以获得累计值，直至所述累计值满足失锁条件时，验证失败。

- 5 在一种可能的实现方式中，所述特性值为校验序列中非全零序列的数量，所述失锁条件为所述累计值大于或等于同步阈值；或者，所述特性值为校验序列中非零元素的数量，所述失锁条件为所述累计值大于或等于同步阈值；或者，所述特性值为不可纠错码字的数量，所述失锁条件为所述累计值大于或等于同步阈值；或者，所述特性值为重检验位与原始校验位不相同的码字的数量，所述失锁条件为所述累计值大于或等于同步阈值；其中，
10 所述码字的长度为 n 比特，所述码字的前 k 比特为信息位，所述码字的后 $n-k$ 比特为所述原始校验位，所述重检验位是根据所述信息位获得的，所述重校验位的长度为 $n-k$ 比特，所述 n 和所述 k 为整数。

- 在一种可能的实现方式中，对所述多个同步码字的特性值进行验证，包括：将所述多个同步码字中所有同步码字的特性值相加以获得总值，当所述总值满足失锁条件时，验证
15 失败。

- 在一种可能的实现方式中，所述特性值为校验序列中全零序列的数量，所述失锁条件为所述总值小于或等于同步阈值；或者，所述特性值为校验序列中非全零序列的数量，所述失锁条件为所述总值大于或等于同步阈值；或者，所述特性值为校验序列中零元素的数量，所述失锁条件为所述总值小于或等于同步阈值；或者，所述特性值为校验序列中非零
20 元素的数量，所述失锁条件为所述总值大于或等于同步阈值；或者，所述特性值为可纠错码字的数量，所述失锁条件为所述总值小于或等于同步阈值；或者，所述特性值为不可纠错码字的数量，所述失锁条件为所述总值大于或等于同步阈值；或者，所述特性值为重检验位与原始校验位相同的码字的数量，所述失锁条件为所述总值小于或等于同步阈值；或者，所述特性值为重检验位与原始校验位不相同的码字的数量，所述失锁条件为所述总值
25 大于或等于同步阈值；其中，所述码字的长度为 n 比特，所述码字的前 k 比特为信息位，所述码字的后 $n-k$ 比特为所述原始校验位，所述重检验位是根据所述信息位获得的，所述重校验位的长度为 $n-k$ 比特，所述 n 和所述 k 为整数。

在一种可能的实现方式中，所述数据序列为线性分组码。

- 第二方面，本申请提供了一种通信设备，执行第一方面或第一方面的任意一种可能的
30 实现方式中的方法。具体地，该网络设备包括用于执行第一方面或第一方面的任意一种可能的实现方式中的方法的单元。

- 第三方面，本申请提供了一种通信设备，该通信设备包括：处理器、通信接口和存储器。通信接口可以是收发器。存储器可以用于存储程序代码，处理器用于调用存储器中的程序代码执行前述第一方面或第一方面的任意一种可能的实现方式中的方法，此处不再赘
35 述。

第四方面，本申请提供了一种网络系统，该网络系统包括发送设备和接收设备，所述接收设备为前述第二或第三方面提供的通信设备，所述接收设备用于接收所述发送设备发送的数据序列。

第五方面，本申请提供了一种计算机可读存储介质，所述计算机可读存储介质中存储

有指令，当其在计算机上运行时，使得计算机执行上述各方面所述的方法。

第六方面，本申请提供了一种包括计算机程序指令的计算机程序产品，当该计算机程序产品在网络设备上运行时，使得网络设备执行第一方面或第一方面的任意一种可能的实现方式中提供的方法。

- 5 第七方面，本申请提供了一种芯片，包括存储器和处理器，存储器用于存储计算机程序，处理器用于从存储器中调用并运行该计算机程序，以执行上述第一方面及其第一方面任意可能的实现方式中的方法。

可选地，上述芯片仅包括处理器，处理器用于读取并执行存储器中存储的计算机程序，当计算机程序被执行时，处理器执行第一方面或第一方面任意可能的实现方式中的方法。

- 10 第八方面，本申请提供了一种网络节点，该网络节点包括：主控板和接口板。主控板包括：第一处理器和第一存储器。接口板包括：第二处理器、第二存储器和接口卡。主控板和接口板耦合。

- 第一存储器可以用于存储程序代码，第一处理器用于调用第一存储器中的程序代码执行如下操作：确定所述数据序列中的备选比特，所述备选比特包括在所述多个比特中；根据所述备选比特确定同步位置，所述同步位置用于指示所述数据序列中码字的起始位置。

- 15 第二存储器可以用于存储程序代码，第二处理器用于调用第二存储器中的程序代码，触发接口卡执行如下操作：接收数据序列，所述数据序列包括多个比特。

- 在一种可能的实现方式中，主控板和接口板之间建立进程间通信协议（inter-process communication, IPC）通道，主控板和接口板之间通过 IPC 通道进行通信。

20

附图说明

- 为了更清楚地说明本申请的技术方案，下面将对实施例中所使用的附图作简单地介绍。显而易见地，下面附图只是本申请的一些实施例的附图，对于本领域普通技术人员来说，在不付出创造性劳动性的前提下，还可以根据这些附图获得同样能实现本申请的其他技术方案和附图。

- 25 图 1 为本申请实施例提供的一种码字的同步位置；
图 2 为本申请实施例提供的一种同步流程图；
图 3 为本申请实施例提供的一种同步流程图；
图 4 为本申请实施例提供的一种观察比特的选择方法；
30 图 5 为本申请实施例提供的一种观察比特的选择方法；
图 6 为本申请实施例提供的一种观察比特的选择方法；
图 7 为本申请实施例提供的一种观察比特的选择方法；
图 8 为本申请实施例提供的一种同步锁定判断第一阶段；
图 9 为本申请实施例提供的一种同步锁定判断第二阶段；
35 图 10 为本申请实施例提供的一种同步失锁判断；
图 11 为本申请实施例提供的一种一种同步流程图；
图 12 为本申请实施例提供的一种通信设备的结构示意图；
图 13 为本申请实施例提供的一种通信设备的结构示意图；
图 14 为本申请实施例提供的一种通信设备的结构示意图。

具体实施方式

下面结合附图，对本申请的实施例进行描述。

由于信道损耗和噪声的影响，当信号在信道中传输时会出现错误，影响通信系统的可靠性。当信号表示为多个比特组成的数据序列的形式时，传输错误具体表现为数据序列中部分比特的值发生变化，即出现错误比特。FEC 是一种在通信系统中控制传输错误的技术，在 FEC 技术中，冗余信息与原始数据序列一起发送，以用于传输过程中的错误恢复，降低比特误码率。FEC 按照对信息序列的处理方式不同可分为分组码和卷积码。对于分组码，可再细分为线性分组码与非线性分组码。以线性分组码中的系统码（systematic code）为例，发送端将原始数据序列进行分组，每组的长度为 k 比特（bit）。进而在每个分组中，按照特定编码规则，增加 $n-k$ 比特的冗余信息，也称为校验位（parity），最终得到长度为 n 比特的码字。从而，长度为 n 比特的码字中，前 k 比特为原始数据，也称为信息位，后 $n-k$ 比特为校验位，整个码字由信息位和校验位组成。码字经过信道到达接收端之后，如果码字内错误比特的个数在可纠错范围之内，接收端通过解码过程即可检查并纠正错误，将接收到的码字恢复为发送端发送的原始数据，从而抵抗信道带来的干扰，提高通信系统的可靠性。线性分组码的检错及纠错功能的实现必须基于完整的码字进行，因而，在对接接收端所收到的数据序列进行解码之前，需要在数据序列中确定码字边界，即找到一个完整码字的起始位置和结束位置。这一过程称为码字同步或者帧同步。如果码字同步不正确，即未能确定真正的码字边界，则后续解码过程中无法达到检错或纠错的应有效果，甚至可能增加误码，造成通信系统性能的恶化。

每个码字的起始位置也可以称为同步位置（synchronization position）。同样以系统码为例，图 1 示出了码字中的同步位置。如图 1 所示，整个码字的长度为 n 比特，其中前 k 比特为原始数据（information bits），后 $n-k$ 比特为通过编码规则加入的额外校验位（parity bits），同步位置为码字的起始位置，即码字中的首个比特所在的位置。其中， n 和 k 均为整数。由此可见，在数据序列中，同步位置的数量为多个，且多个同步位置相互关联，其中每个同步位置之间的间隔是固定的，该间隔即为码字的长度。

在本申请实施例中，虽然采用系统码进行举例说明，但本申请并不限于系统码，也适用于非系统码（non-systematic code），本申请能够适用于所有线性分组码的通信系统。所述线性分组码包括但不限于里德-所罗门码（Reed-Solomon code, RS code），博斯-查德胡里-霍克金亨码（Bose-Chaudhuri-Hocquenghem code, BCH code），低密度奇偶校验码（Low-Density Parity-Check code, LDPC code），汉明码（Hamming code），格莱码（Golay code），以及里德-穆勒（Reed-Muller code），等等。

本申请实施例提供了码字同步方法以及基于该方法的设备和系统。这些方法、设备和系统基于同一发明构思。该方法可以包括两个阶段，在第一阶段中，根据多个观察比特划分多组第一测试数据块，根据多组第一测试数据块从多个观察比特中选取一个最可能位于同步位置的观察比特，在第二阶段中，根据所述最可能位于同步位置的观察比特划分一组第二测试数据块，根据该组第二测试数据块判断所述最可能位于同步位置的观察比特是否为同步位置。该方法还可以只包括所述第一阶段，在确定选取所述最可能位于同步位置的观察比特后，直接将该最可能位于同步位置的观察比特作为同步位置。在该方法中，无需

在发送端数据流中插入额外数据,即可实现接收端数据流高精度码字同步的技术效果,其同步性能达到较高可靠性。

图2示出了本申请实施例的方法流程图。该方法应用于通信网络中的接收端设备。所述接收端设备可以是执行FEC的各种设备,包括但不限于路由器、交换机和服务端。该方法的步骤包括:

S210,接收数据序列。

具体的,接收端接收来自发送端的数据序列,所述数据序列包括多个比特,每个比特为二进制数据。即所述数据序列是多个比特组成的序列,也可以称为比特序列。在一些实施例中,所述数据序列为线性分组码。所述数据序列经过信道传输,可能存在误码,需要进行检错及纠错等操作。

S220,第一阶段,确定所述数据序列中的备选比特,所述备选比特包括在所述多个比特中。所述备选比特是多个观察比特中最可能位于同步位置的观察比特,或者说,所述备选比特的位置是最可能的同步位置。

如图3所示,在第一阶段中,可以包括S221和S222两个步骤:

S221,接收端从接收到的数据序列中选取多个观察比特。这些观察比特的位置包括所述同步位置,即这些观察比特中某个观察比特的位置可能为所述同步位置,或者说,这些观察比特的位置能够覆盖所述同步位置。

可选的,在第一阶段中,该方法在确定所述数据序列中的备选比特时,仅利用所述数据序列中的一部分,例如所述数据序列中的第一子序列,即确定第一子序列中的备选比特。所述第一子序列包括多个比特,所述第一子序列可以是所述数据序列中的任何一个部分。接收端从第一子序列中选取所述多个观察比特,这些观察比特的位置可能为同步位置。例如,所述观察比特的数量为 P , P 为大于1的整数。

可选的,从接收到的数据序列中选取多个观察比特时,所述多个观察比特中每两个相邻的观察比特之间的间隔是相同的,所述多个观察比特中每个观察比特的位置依次向后移动。

在码字同步过程中,引入测试数据块的概念。所述测试数据块为数据序列中的一部分,所述测试数据块由连续的若干比特组成,所述测试数据块的长度与所述码字的长度相同。可以说,所述测试数据块是用于模拟一个码字的。

可选的,所述多个观察比特可以位于同一个测试数据块内,所述多个观察比特中每个观察比特在同一个测试数据块内的位置依次向后移动。具体的,每间隔 T 个比特选取一个比特作为所述观察比特,所述 T 为大于零的整数。图4示出了观察比特的选取方式,其中 b_1, b_2, b_3 为相邻的三个观察比特,分别对应数据序列中的 $x(0), x(1), x(2)$ 三个比特。每个观察比特之间的间隔为1比特,即 T 为1。三个观察比特位于同一个测试数据块内,例如图4中以 $x(0)$ 为起始位置所划分出的测试数据块内。在该测试数据块内, b_1 所在的位置为该测试数据块内的第一个位置, b_2, b_3 所在的位置依次为该测试数据块内的第二个和第三个位置,即三个观察比特中每个观察比特在同一个测试数据块内的位置依次向后移动。虽然图4仅示出了三个观察比特,但更多的观察比特的选取方式可以由此类推,在此不再赘述。

可选的,所述多个观察比特可以分别位于多个测试数据块内,所述多个观察比特中每个观察比特在每个测试数据块内的位置依次向后移动。具体的,每间隔 $L \cdot n + T$ 个比特选取

一个比特作为所述观察比特,所述 L 为间隔测试数据块的数量,所述测试数据块的长度为所述 n 个比特,所述 L 和所述 T 为大于零的整数。图 5 示出了观察比特的选取方式,其中 b_1, b_2 为相邻的两个观察比特,分别对应数据序列中的 $x(0), x(L \cdot n + 1)$ 两个比特。两个观察比特分别位于两个不同的测试数据块内,例如图 5 中以 $x(0)$ 为起始位置所划分出的测试数据块内和以 $x(L \cdot n)$ 为起始位置所划分出的测试数据块内。两个测试数据块之间的间隔为 L 个测试数据块,两个观察比特之间的间隔为 $L \cdot n + 1$ 个比特,即 T 为 1。在以 $x(0)$ 为起始位置所划分出的测试数据块内, b_1 所在的位置为该测试数据块内的第一个位置,在以 $x(L \cdot n)$ 为起始位置所划分出的测试数据块内, b_2 所在的位置为该测试数据块内的第二个位置,即两个观察比特中每个观察比特在每个测试数据块内的位置依次向后移动。虽然图 5 仅示出了两个观察比特,但更多的观察比特的选取方式可以由此类推,在此不再赘述。此时,相邻的观察比特之间的间隔更远,观察比特之间的相关性更低,从而能够减少突发误码带来的影响,提高码字同步的准确性。

可选的,在选取所述多个观察比特时,可以使得所述多个观察比特的位置遍历测试数据块内的所有位置。

例如,当所述多个观察比特位于同一个测试数据块内时,可以使得所述多个观察比特的位置遍历该同一个测试数据块内的所有位置,即使得所述多个观察比特的位置遍历该同一个码字内的所有位置。在图 4 所示的观察比特的选取方式中,码字的长度为 n 比特,观察比特的数量为 P ,当 P 等于 n 时,所述多个观察比特的位置即可遍历同一个测试数据块内的所有位置。

例如,当所述多个观察比特分别位于多个测试数据块内时,所述多个测试数据块中每个测试数据块内的相同位置视为一个等同位置,例如,多个测试数据块中每个测试数据块内的第一个位置视为一个等同位置,可以使得所述多个观察比特的位置遍历所有等同位置,即所述多个观察比特的位置遍历多个测试数据块内的所有位置。在图 5 所示的观察比特的选取方式中,码字的长度为 n 比特,则等同位置的数量为 n ,观察比特的数量为 P ,当 P 等于 n 时,所述多个观察比特的位置即可遍历所有等同位置,即所述多个观察比特的位置即可遍历多个测试数据块内的所有位置。

可选的,在选取所述多个观察比特时,也可以使得所述多个观察比特的位置不遍历码字内的所有位置。即所述 P 的值可以小于所述 n 。

如果所述多个观察比特的位置遍历测试数据块内的所有位置,将使得码字同步结果的准确度更高。然而,随着观察比特的数量增加,码字同步过程所消耗的系统资源也将增加。

在发送端可以对信号进行调制,以获得调制信号。常用的调制方式包括脉冲幅度调制 (Pulse-Amplitude Modulation, PAM)、正交幅度调制 (Quadrature Amplitude Modulation, QAM)、相移键控 (Phase-Shift Keying, PSK) 等。调制信号的脉冲幅度可以包括多个阶次,该多个阶次的数量可以称为调制的阶数。以 PAM 为例,当调制信号的脉冲幅度有 2 个阶次时,调制阶数为 2,调制方式可以称为 PAM2;当调制信号的脉冲幅度有 4 个阶次时,调制阶数为 4,调制方式可以称为 PAM4,以此类推。调制信号可以表示为调制的数据序列。与未调制的数据序列相比,调制的数据序列的基本单元由比特变更为码元,即调制的数据序列中,码字由若干个码元组成,码元由若干个比特组成。码元中包括的比特数量与调制的阶数相关。当调制阶数为 M 时,码元包括的比特数量为 $\log_2 M$,其中 M 为 2 的整

数倍。例如，当调制阶数为 2 时，码元中包括 1 个比特，当调制阶数为 4 时，码元中包括 2 个比特。由此可见，码字的起始位置必然也是某个码元的起始位置，而不会是码元的其他位置。因此，对于调制的信号来说，在选取所述多个观察比特时，可以只考虑码元的起始比特，而不用考虑码元的其他比特。

- 5 可选的，所述多个观察比特分别为多个码元的起始比特，所述多个码元在同一个测试数据块内，且所述多个码元在同一个测试数据块内的位置依次向后移动。具体的，每间隔 T 个调制码元，选取调制码元的起始比特，作为所述观察比特，所述 T 为大于零的整数。也就是说，每间隔 $T \cdot \log_2 M$ 个比特，选取调制码元的起始比特，作为所述观察比特。图 6 示出了观察比特的选取方式，其中数据序列为调制信号，调制阶数为 4，则该数据序列包
10 括多个调制码元，每个码元中包括 2 个比特，例如 $x(0)$ 和 $x(1)$ 组成的码元， $x(2)$ 和 $x(3)$ 组成的码元，以及 $x(4)$ 和 $x(5)$ 组成的码元，以此类推。其中 b_1, b_2, b_3 为相邻的三个观察比特，分别对应数据序列中的 $x(0), x(2), x(4)$ 三个比特。所述三个观察比特分别为三个码元的起始比特，三个码元为 $x(0)$ 和 $x(1)$ 组成的码元， $x(2)$ 和 $x(3)$ 组成的码元，以及 $x(4)$ 和 $x(5)$ 组成的码元。所述三个码元中每两个相邻码元之间间隔 1 个码元，即 T 为 1。并且，所述三个
15 码元在同一个测试数据块内的位置依次向后移动。虽然图 6 仅示出了三个观察比特，但更多的观察比特的选取方式可以由此类推，在此不再赘述。

- 可选的，所述多个观察比特分别为多个码元的起始比特，所述多个码元分别位于多个测试数据块内，且所述多个码元中每个码元在每个测试数据块内的位置依次向后移动。具体的，每间隔 $L \cdot m + T$ 个调制码元，选取调制码元的起始比特，作为所述观察比特，所述
20 L 为间隔测试数据块的数量，所述测试数据块的长度为所述 m 个调制码元，所述 L 和所述 T 为大于零的整数。也就是说，每间隔 $(L \cdot m + T) \cdot \log_2 M$ 个比特，选取调制码元的起始比特，作为所述观察比特。图 7 示出了观察比特的选取方式，其中数据序列为调制信号，调制阶数为 4，则该数据序列包括多个调制码元，每个码元中包括 2 个比特，例如 $x(0)$ 和 $x(1)$ 组成的码元， $x(L \cdot n)$ 和 $x(L \cdot n + 1)$ 组成的码元，以及 $x(L \cdot n + 2)$ 和 $x(L \cdot n + 3)$ 组成的码元，
25 以此类推。其中 b_1, b_2 为相邻的两个观察比特，分别对应数据序列中的 $x(0), x(L \cdot n + 2)$ 两个比特。所述两个观察比特分别为两个码元的起始比特，两个码元为 $x(0)$ 和 $x(1)$ 组成的码元， $x(L \cdot n + 2)$ 和 $x(L \cdot n + 3)$ 组成的码元。并且，所述两个观察比特分别位于两个不同的测试数据块内，例如图 7 中以 $x(0)$ 为起始位置所划分出的测试数据块内和以 $x(L \cdot n)$ 为起始位置所划分出的测试数据块内。所述两个测试数据块之间的间隔为 L 个测试数据块，两个观察
30 比特之间的间隔为 $L \cdot m + 1$ 个码元，即 T 为 1。在以 $x(0)$ 为起始位置所划分出的测试数据块内， b_1 所在的码元为该测试数据块内的第一个码元，在以 $x(L \cdot n)$ 为起始位置所划分出的测试数据块内， b_2 所在的码元为该测试数据块内的第二个码元，即两个码元中每个码元在每个测试数据块内的位置依次向后移动。虽然图 7 仅示出了两个观察比特，但更多的观察比特的选取方式可以由此类推，在此不再赘述。此时，相邻的观察比特之间的间隔更远，
35 观察比特之间的相关性更低，从而能够减少突发误码带来的影响，提高码字同步的准确性。

S222, 从所述多个观察比特中选取备选比特。所述备选比特是所述多个观察比特中的一个观察比特，且所述备选比特是最可能位于同步位置的观察比特。具体的，S222 可以包括 S2221 和 S2222 两个步骤。

S2221, 根据所述多个观察比特从数据序列中划分出多组第一测试数据块，所述多组

第一测试数据块与所述多个观察比特一一对应。具体的，所述多组第一测试数据块中的每组第一测试数据块包括至少一个第一测试数据块，所述多个观察比特中的每一个观察比特的位置为所述多组第一测试数据块中的每组第一测试数据块的起始位置。

- 可选的，分别以所述多个观察比特中的每个观察比特为起始位置，选取其后的 N 个测试数据块，其中 N 为大于等于 1 的整数。所述多个观察比特中的每一个观察比特的位置为所述多组第一测试数据块中的每组第一测试数据块的起始位置。以每个观察比特为起始位置选取的 N 个测试数据块为一组第一测试数据块，则分别以多个观察比特中的每个观察比特为起始位置，选取的多组 N 个测试数据块为所述多组第一测试数据块。所述多组第一测试数据块中的每组第一测试数据块对应作为这组第一测试数据块的起始位置的一个观察比特，即所述多组第一测试数据块与所述多个观察比特一一对应。

可选的，所述 N 个测试数据块可以是连续的，即每组第一测试数据块包括连续的 N 个数据块。当每个测试数据块的长度为 n 比特时，即选取其后的连续 $N*n$ 个比特，即获得所述 N 个测试数据块。

- 可选的，所述 N 个测试数据块也可以是不连续的，所述 N 个测试数据块中每个测试数据块的起始位置均与该组测试数据块对应的观察比特相关联。具体的，每个测试数据块的起始位置与该组测试数据块对应的观察比特的间隔均为码字的长度的整数倍，即所述 N 个测试数据块中每两个测试数据块的间隔均为码字的长度的整数倍。并且，所述 N 个测试数据块中每两个测试数据块的间隔可以相同，也可以不同。

- 如图 4 所示，以观察比特 b_1 为起始位置，选取其后的一组第一测试数据块，该组第一测试数据块包括 N 个测试数据块，即 $B_1, B_2, \dots, B_N$ ，每个测试数据块的长度为 n 比特。类似的，以观察比特 b_2 或 b_3 为起始位置，也能够分别选取出其后的一组第一测试数据块。分别以观察比特 b_1, b_2 或 b_3 为起始位置，则选取出所述多组第一测试数据块。图 5、图 6 和图 7 中，所述多组第一测试数据块的方式与图 4 类似，在此不再赘述。

- S2222, 根据所述多组第一测试数据块的特性值，从所述多个观察比特中选取一个观察比特作为所述备选比特，即所述最可能位于同步位置的观察比特。

- 可选的，当根据所述多组第一测试数据块的特性值，从所述多个观察比特中选取一个观察比特作为所述最可能位于同步位置的观察比特时，选取方式可以是提前终止的选取方式。具体的，依次判断所述多组第一测试数据块中各组第一测试数据块的特性值是否满足备选条件，直至确定出一组第一测试数据块的特性值满足所述备选条件，将满足所述备选条件的所述一组第一测试数据块对应的观察比特作为所述备选比特。

- 可选的，当根据所述多组第一测试数据块的特性值，从所述多个观察比特中选取一个观察比特作为所述最可能位于同步位置的观察比特时，选取方式可以是遍历的选取方式。具体的，比较所述多组第一测试数据块中每组第一测试数据块的特性值，将所述特性值为极值的一组第一测试数据块对应的观察比特作为所述备选比特。

- 在所述提前终止的选取方式和所述遍历的选取方式下，针对特性值的计算都可以并行进行，即采用多个计算单元同时计算多个测试数据块的特性值。例如，采用 10 个计算单元同时计算 10 个测试数据块的特性值。并行的计算方式能够节约计算时间，降低时延。

可选的，所述特性值可以是校验序列的特性值。

针对每个线性分组码可以生成一个校验矩阵 (parity-check matrix)，该校验矩阵描述

了线性分组码的码字内的数据之间的线性关系，并且该校验矩阵可以应用于解码过程。当所述数据序列为线性分组码时，针对所述数据序列也存在所述校验矩阵，可以称为所述数据序列的校验矩阵。具体的，数据序列与该数据序列的校验矩阵之间，满足下列关系式：

$$S = C \cdot H^T = \vec{0} \quad (1)$$

- 5 其中，C 为数据序列中的一个码字，S 为所述码字的校验序列，H 为该数据序列的校验矩阵。由上述关系式 (1) 可知，数据序列中码字的校验序列由码字与该数据序列的校验矩阵的转置矩阵相乘而得，且所述校验序列为一个零向量，也即一个全零矩阵。

- 如上所述，当该数据序列为接收端所接收到的数据序列时，需要在数据序列中确定码字的起始位置。如果没有以正确的起始位置从数据序列中划分码字，则得到的码字将不再满足上述关系式 (1)。此时，该数据序列中码字的校验序列可以用下列关系式表示：

$$S_R = R \cdot H^T \quad (2)$$

- 其中，R 为接收端所接收的数据序列中的一个码字， S_R 为该码字的校验序列，H 为发送端所发送的原始数据序列的校验矩阵。在上述关系式 (2) 中，接收端所接收的数据序列中的码字的校验序列，为该码字与发送端所发送的原始数据序列的校验矩阵的转置矩阵的乘积。由上述分析可知，对于一个接收码字，码字内没有出现错误比特时， S_R 是全零矩阵；当码字内存在错误比特时， S_R 极大概率不再为全零矩阵。因此，可以利用接收端所接收的数据序列中测试数据块的校验序列来实现码字同步。具体的，可以利用接收端所接收的数据序列中测试数据块的校验序列，从所述多个观察比特中选取一个观察比特作为所述最可能位于同步位置的观察比特。

- 20 可选的，分别计算所述多组第一测试数据块中每组第一测试数据块的校验序列。

- 如图 4 所示，计算以观察比特 b_1 为起始比特的一组第一测试数据块的校验序列，即图 4 所示的 $S^{(1)}$ ，包括 $S_1^{(1)}, S_2^{(1)}, \dots, S_N^{(1)}$ ，分别对应测试数据块 $B_1, B_2, \dots, B_N$ 。以此类推，分别计算以观察比特 b_2, b_3 为起始比特的一组第一测试数据块的校验序列，即图 4 所示的 $S^{(2)}, S^{(3)}$ 。虽然图 4 仅示出了以三个观察比特为起始比特的一组第一测试数据块的校验序列的计算方式，但更多的校验序列的计算方式可以由此类推，在此不再赘述。图 5、图 6 和图 7 中，所述多组第一测试数据块的校验序列的计算方式与图 4 类似，在此不再赘述。

- 例如，所述特性值可以是校验序列中全零序列的数量。当采取上述提前终止的选取方式时，所述备选条件与所述特性值相关联。所述备选条件可以是一组第一测试数据块的特性值大于或等于同步阈值。具体的，依次判断所述各组第一测试数据块的校验序列中全零序列的数量与同步阈值的关系，直至确定出一组第一测试数据块的校验序列中全零序列的数量大于或等于同步阈值时，就将满足所述备选条件的所述一组第一测试数据块对应的观察比特作为所述备选比特。一组第一测试数据块对应的观察比特，也就是作为该组第一测试数据块的起始比特的观察比特。

- 所述同步阈值的取值可以通过仿真分析获得。例如，将一段数据序列作为接收端所接收到的数据序列，设计该数据序列的误码率在可接受的范围内，分别统计基于同步位置所划分的一组测试数据块的校验序列中全零序列的数量，以及基于非同步位置所划分的一组测试数据块的校验序列中全零序列的数量，根据两者的区别确定同步阈值，以使得所述同步阈值能够用于区别同步位置和非同步位置。

当特性值为校验序列中全零序列的数量，采取上述提前终止的选取方式时，所述最可

能位于同步位置的观察比特的选取过程可以参见图 4。其中，同步阈值可以为 2。如图 4 所示，依次判断校验序列 $S^{(1)}, S^{(2)}, S^{(3)}$ 中全零序列的数量与同步阈值的关系，例如，第一组第一测试数据块的校验序列中全零序列的数量 $Q_1=0$ ，小于所述同步阈值，第二组第一测试数据块的校验序列中全零序列的数量 $Q_2=2$ ，等于所述同步阈值，则选取 b_2 为所述最可能位于同步位置的观察比特。并且，当判断至 b_2 时，确定 b_2 满足所述备选条件，则不再判断后续的观察比特对应的校验序列中全零序列的数量与同步阈值的关系，即不再判断 $S^{(3)}$ 中全零序列的数量与同步阈值的关系。

可选的，在上述提前终止的选取方式中，对于每组第一测试数据块的特性值，也可以采取提前终止的统计方式。也就是说，所述一组第一测试数据块的特性值可以是所述一组第一测试数据块中所有第一测试数据块的校验序列中全零序列的数量总值，相应的，所述备选条件可以是所述总值大于或等于同步阈值。如图 4 所示的选取过程中，依次判断各组校验序列 $S^{(1)}, S^{(2)}, S^{(3)}$ 中全零序列的数量与同步阈值的关系时，对于每组第一测试数据块，依次计算其中每个测试数据块的校验序列，当校验序列为全零序列时，则将全零序列的数量增加 1，当所述全零序列的数量累积值满足备选条件，即所述全零序列的数量累积值大于或等于所述同步阈值时，则认为该组第一测试数据块已经满足备选条件，对于该组中后续未计算校验序列的测试数据块将不再进行计算。例如，对于第二组第一测试数据块来说，如果 $S_1^{(2)}, S_2^{(2)}$ 均为全零序列，则认为第二组第一测试数据块的校验序列中全零序列的数量 Q_2 已累积至 2，达到所述同步阈值，该组第一测试数据块已经满足备选条件，则选取 b_2 为所述最可能位于同步位置的观察比特，不再计算后续 $S_3^{(2)}$ 至 $S_N^{(2)}$ 的校验序列。可选的，在上述提前终止的选取方式中，对于每组第一测试数据块的特性值，也可以采取遍历的统计方式。也就是说，所述一组第一测试数据块的特性值可以是所述一组第一测试数据块中前 X 个第一测试数据块的校验序列中全零序列的数量累计值，相应的，所述备选条件可以是所述累计值大于或等于同步阈值。在具体选取过程中，计算该组中每个第一测试数据块的校验序列，以确定该组校验序列中全零序列的数量总值是否满足备选条件，即该组校验序列中全零序列的数量总值是否大于或等于所述同步阈值，具体操作过程在此不再赘述。

例如，所述特性值可以是校验序列中全零序列的数量。当采取上述遍历的选取方式时，所述极值与所述特性值相关联。所述极值可以是最大值。具体的，可以统计所述每组第一测试数据块的校验序列中全零序列的数量，确定全零序列的数量最多的一组第一测试数据块，将该组第一测试数据块对应的观察比特作为所述最可能位于同步位置的观察比特。

当特性值为校验序列中全零序列的数量，采取上述遍历的选取方式时，所述最可能位于同步位置的观察比特的选取过程也可以参见图 4。如图 4 所示，分别统计校验序列 $S^{(1)}, S^{(2)}, S^{(3)}$ 中全零序列的数量，分别记为 Q_1, Q_2, Q_3 ，确定其中的最大数值，例如， $Q_1=0, Q_2=0, Q_3=2$ ，即最大数值为 $Q_3=2$ ，则选取 b_3 为所述最可能位于同步位置的观察比特， b_3 所在的位置为最可能的同步位置。

例如，所述特性值可以是校验序列中非全零序列的数量。当采取上述提前终止的选取方式时，所述备选条件与所述特性值相关联。所述备选条件可以是一组第一测试数据块的特性值小于或等于同步阈值。具体的，依次判断所述各组第一测试数据块的校验序列中非全零序列的数量与同步阈值的关系，直至确定出一组第一测试数据块的校验序列中非全零序列的数量小于或等于同步阈值时，就将满足所述备选条件的所述一组第一测试数据块对

应的观察比特作为所述备选比特。所述同步阈值的取值也可以通过仿真分析获得，具体过程与全零序列对应的同步阈值类似，在此不再赘述。

例如，所述特性值可以是校验序列中非全零序列的数量。当采取上述遍历的选取方式时，所述极值与所述特性值相关联。所述极值可以是最小值。具体的，可以统计所述每组
5 第一测试数据块的校验序列中非全零序列的数量，确定非全零序列的数量最少的一组第一测试数据块，将该组第一测试数据块对应的观察比特作为所述最可能位于同步位置的观察比特。

例如，所述特性值可以是校验序列中零元素的数量。当采取上述提前终止的选取方式时，所述备选条件与所述特性值相关联。所述备选条件可以是一组第一测试数据块的特性
10 值大于或等于同步阈值。具体的，依次判断所述各组第一测试数据块的校验序列中零元素的数量与同步阈值的关系，直至确定出一组第一测试数据块的校验序列中零元素的数量大于或等于同步阈值时，就将满足所述备选条件的所述一组第一测试数据块对应的观察比特作为所述备选比特。所述同步阈值的取值也可以通过仿真分析获得。

例如，所述特性值可以是校验序列中零元素的数量。当采取上述遍历的选取方式时，
15 所述极值与所述特性值相关联。所述极值可以是最大值。具体的，可以统计所述每组第一测试数据块的校验序列中零元素的数量，确定零元素的数量最多的一组第一测试数据块，将该组第一测试数据块对应的观察比特作为所述最可能位于同步位置的观察比特。

例如，所述特性值可以是校验序列中非零元素的数量。当采取上述提前终止的选取方式时，所述备选条件与所述特性值相关联。所述备选条件可以是一组第一测试数据块的特
20 性值小于或等于同步阈值。具体的，依次判断所述各组第一测试数据块的校验序列中非零元素的数量与同步阈值的关系，直至确定出一组第一测试数据块的校验序列中非零元素的数量小于或等于同步阈值时，就将满足所述备选条件的所述一组第一测试数据块对应的观察比特作为所述备选比特。所述同步阈值的取值也可以通过仿真分析获得。

例如，所述特性值可以是校验序列中非零元素的数量。当采取上述遍历的选取方式时，
25 所述极值与所述特性值相关联。所述极值可以是最小值。具体的，可以统计所述每组第一测试数据块的校验序列中非零元素的数量，确定非零元素的数量最少的一组第一测试数据块，将该组第一测试数据块对应的观察比特作为所述最可能位于同步位置的观察比特。

当所述特性值为校验序列中非全零序列、零元素或非零元素时，结合图 4 所示的数据序列的具体实现过程，可以参考当所述特性值为校验序列中全零序列时的介绍，在此不再
30 赘述。

可选的，所述特性值可以是可纠错特性值。接收端在数据序列中划分出码字后，能够确定码字的状态，所述状态包括可纠错状态和不可纠错状态。当码字处于可纠错状态时，该码字也称为可纠错码字；当码字处于不可纠错状态时，该码字也称为不可纠错码字。一组码字的可纠错特性值可以与该组码字中码字的状态相关联。类似的，对于接收端接收到的数据序列中划出的测试数据块来说，也能够确定测试数据块的状态，所述状态包括可
35 纠错状态和不可纠错状态。当测试数据块处于可纠错状态时，该测试数据块也称为可纠错测试数据块；当测试数据块处于不可纠错状态时，该测试数据块也称为不可纠错测试数据块。一组测试数据块的可纠错特性值可以与该组测试数据块中测试数据块的状态相关联。

例如，所述可纠错特性值可以是可纠错测试数据块的数量。当采取上述提前终止的选

取方式时,所述备选条件与所述特性值相关联。所述备选条件可以是一组第一测试数据块的特性值大于或等于同步阈值。具体的,依次判断所述各组第一测试数据块中可纠错测试数据块的数量与同步阈值的关系,直至确定出一组第一测试数据块中可纠错测试数据块的数量大于或等于同步阈值时,就将满足所述备选条件的所述一组第一测试数据块对应的观察比特作为所述备选比特。所述同步阈值的取值也可以通过仿真分析获得。

5 当所述可纠错特性值是可纠错测试数据块的数量,采取上述提前终止的选取方式时,所述最可能位于同步位置的观察比特的选取过程可以参见图4。其中,同步阈值可以为2。如图4所示,依次判断各组第一测试数据块中可纠错测试数据块的数量与同步阈值的关系,例如,第一组第一测试数据块中可纠错测试数据块的数量 $Q_1=0$,小于所述同步阈值,第二组第一测试数据块中可纠错测试数据块的数量 $Q_2=2$,等于所述同步阈值,则选取 b_2 为所述最可能位于同步位置的观察比特。并且,当判断至 b_2 时,确定 b_2 满足所述备选条件,则不再判断后续的观察比特对应的可纠错测试数据块的数量与同步阈值的关系,即不再判断 Q_3 与同步阈值的关系。

15 例如,所述可纠错特性值可以是可纠错测试数据块的数量。当采取上述遍历的选取方式时,所述极值与所述特性值相关联。所述极值可以是最大值。具体的,可以统计所述每组第一测试数据块中可纠错测试数据块的数量,确定可纠错测试数据块的数量最多的一组第一测试数据块,将该组第一测试数据块对应的观察比特作为所述最可能位于同步位置的观察比特。

20 当所述可纠错特性值是可纠错测试数据块的数量,采取上述遍历的选取方式时,所述最可能位于同步位置的观察比特的选取过程也可以参见图4。如图4所示,分别统计以观察比特 b_1, b_2, b_3 为起始比特的各组第一测试数据块中可纠错测试数据块的数量,分别记为 Q_1, Q_2, Q_3 ,确定其中的最大数值,例如, $Q_1=0, Q_2=0, Q_3=2$,即最大数值为 $Q_3=2$,则选取 b_3 为所述最可能位于同步位置的观察比特, b_3 所在的位置为最可能的同步位置。

25 例如,所述可纠错特性值可以是不可纠错测试数据块的数量。当采取上述提前终止的选取方式时,所述备选条件与所述特性值相关联。所述备选条件可以是一组第一测试数据块的特性值小于或等于同步阈值。具体的,依次判断所述各组第一测试数据块中不可纠错测试数据块的数量与同步阈值的关系,直至确定出一组第一测试数据块中不可纠错测试数据块的数量小于或等于同步阈值时,就将满足所述备选条件的所述一组第一测试数据块对应的观察比特作为所述备选比特。所述同步阈值的取值也可以通过仿真分析获得。

30 例如,所述可纠错特性值可以是不可纠错测试数据块的数量。当采取上述遍历的选取方式时,所述极值与所述特性值相关联。所述极值可以是最小值。具体的,可以统计所述每组第一测试数据块中不可纠错测试数据块的数量,确定不可纠错测试数据块的数量最少的一组第一测试数据块,将该组第一测试数据块对应的观察比特作为所述最可能位于同步位置的观察比特。

35 可选的,所述特性值可以是重检验特性值。所述重校验特性值与重校验位与原始校验位之间的关系相关联。结合以上描述,长度为 n 比特的码字中,前 k 比特为信息位,后 $n-k$ 比特为校验位。后 $n-k$ 比特的校验位是以前 k 比特的信息位为依据、按照特定编码规则而计算得出的。对于接收端收到的数据序列来说,如果没有找到数据序列中码字的起始位置,则划分出的码字中,不再满足上述信息位与校验位之间的关系。也就是说,即便按照

相同的特征编码规则,以该前 k 比特为依据,也将无法得出与每个测试数据块中后 $n-k$ 比特相同的内容。接收端收到的数据序列中的每个测试数据块中,后 $n-k$ 比特与前 k 比特之前可能不再满足上述特定编码规则。所以,当以接收端收到的数据序列中的每个测试数据块中的前 k 比特为依据、按照所述特定编码规则重新计算得出的后 $n-k$ 比特,与接收端收到的数据序列中的每个测试数据块中的后 $n-k$ 比特相比,可能存在差异。接收端收到的数据序列中的每个测试数据块中的后 $n-k$ 比特可以称为原始校验位,以接收端收到的数据序列中的每个测试数据块中的前 k 比特为依据、按照所述特定编码规则重新计算得出的后 $n-k$ 比特可以称为重校验位。

由上述分析可知,当测试数据块内存在错误比特时,所述重校验位与所述原始校验位不相同。因此,可以对接收端所接收的数据序列中每个测试数据块中的所述重校验位与所述原始校验位进行比较,来实现码字同步。具体的,可以对接收端所接收的数据序列中每个测试数据块中的所述重校验位与所述原始校验位进行比较,从所述多个观察比特中选取一个观察比特作为所述最可能位于同步位置的观察比特。

例如,所述重校验特性值可以是重校验位与原始校验位相同的测试数据块的数量。当采取上述提前终止的选取方式时,所述备选条件与所述特性值相关联。所述备选条件可以是大于或等于同步阈值。具体的,依次判断所述各组第一测试数据块中重校验位与原始校验位相同的测试数据块的数量与同步阈值的关系,直至确定出一组第一测试数据块中重校验位与原始校验位相同的测试数据块的数量大于或等于同步阈值时,就将满足所述备选条件的所述一组第一测试数据块对应的观察比特作为所述备选比特。所述同步阈值的取值也可以通过仿真分析获得。

当所述重校验特性值是重校验位与原始校验位相同的测试数据块的数量,采取上述提前终止的选取方式时,所述最可能位于同步位置的观察比特的选取过程可以参见图 4。其中,同步阈值可以为 2。如图 4 所示,依次判断各组第一测试数据块中重校验位与原始校验位相同的测试数据块的数量与同步阈值的关系,例如,第一组第一测试数据块中重校验位与原始校验位相同的测试数据块的数量 $Q_1=0$,小于所述同步阈值,第二组第一测试数据块中重校验位与原始校验位相同的测试数据块的数量 $Q_2=2$,等于所述同步阈值,则选取 b_2 为所述最可能位于同步位置的观察比特。并且,当判断至 b_2 时,确定 b_2 满足所述备选条件,则不再判断后续的观察比特对应的重校验位与原始校验位相同的测试数据块的数量与同步阈值的关系,即不再判断 Q_3 与同步阈值的关系。

例如,所述重校验特性值可以是重校验位与原始校验位相同的测试数据块的数量。当采取上述遍历的选取方式时,所述极值与所述特性值相关联。所述极值可以是最大值。具体的,可以统计所述每组第一测试数据块中重校验位与原始校验位相同的测试数据块的数量,确定重校验位与原始校验位相同的测试数据块的数量最多的一组第一测试数据块,将该组第一测试数据块对应的观察比特作为所述最可能位于同步位置的观察比特。

当所述重校验特性值是重校验位与原始校验位相同的测试数据块的数量,采取上述遍历的选取方式时,所述最可能位于同步位置的观察比特的选取过程也可以参见图 4。如图 4 所示,分别统计以观察比特 b_1, b_2, b_3 为起始比特的各组第一测试数据块中重校验位与原始校验位相同的测试数据块的数量,分别记为 Q_1, Q_2, Q_3 ,确定其中的最大数值,例如, $Q_1=0, Q_2=0, Q_3=2$,即最大数值为 $Q_3=2$,则选取 b_3 为所述最可能位于同步位置的观察比特,

b₃所在的位置为最可能的同步位置。

例如，所述重校验特性值可以是重检验位与原始校验位不相同的测试数据块的数量。当采取上述提前终止的选取方式时，所述备选条件与所述特性值相关联。所述备选条件可以是小于或等于同步阈值。具体的，依次判断所述各组第一测试数据块中重检验位与原始校验位不相同的测试数据块的数量与同步阈值的关系，直至确定出一组第一测试数据块中重检验位与原始校验位不相同的测试数据块的数量小于或等于同步阈值时，就将满足所述备选条件的所述一组第一测试数据块对应的观察比特作为所述备选比特。所述同步阈值的取值也可以通过仿真分析获得。

例如，所述重校验特性值可以是重检验位与原始校验位不相同的测试数据块的数量。当采取上述遍历的选取方式时，所述极值与所述特性值相关联。所述极值可以是最小值。具体的，可以统计所述每组第一测试数据块中重检验位与原始校验位不相同的测试数据块的数量，确定重检验位与原始校验位不相同的测试数据块的数量最少的一组第一测试数据块，将该组第一测试数据块对应的观察比特作为所述最可能位于同步位置的观察比特。

可选的，根据所述多组第一测试数据块的特性值，分别确定所述多个观察比特的同步可能性指标（synchronization possibility index），该同步可能性指标的数值越大，表明该观察比特的位置为同步位置的可能性越大。例如，当所述多组第一测试数据块的特性值为所述多组第一测试数据块中每组第一测试数据块的校验序列中全零序列的数量时，所述同步可能性指标可以与所述全零序列的数量正相关，即全零序列的数量越多，同步可能性指标的数值越大，表明该观察比特的位置为同步位置的可能性越大。例如，当所述多组第一测试数据块的特性值为所述多组第一测试数据块中每组第一测试数据块的校验序列中非全零序列的数量时，所述同步可能性指标可以与所述非全零序列的数量负相关，即非全零序列的数量越少，同步可能性指标的数值越大，表明该观察比特的位置为同步位置的可能性越大。同理，当所述多组第一测试数据块的特性值为所述多组第一测试数据块中每组第一测试数据块的校验序列中零元素的数量，或者所述多组第一测试数据块的特性值为所述多组第一测试数据块中每组第一测试数据块中可纠错测试数据块的数量，或者所述多组第一测试数据块的特性值为所述多组第一测试数据块中每组第一测试数据块中重检验位与原始校验位相同的测试数据块的数量时，所述同步可能性指标可以与所述零元素的数量或者可纠错测试数据块的数量或者重检验位与原始校验位相同的测试数据块的数量正相关；当所述多组第一测试数据块的特性值为所述多组第一测试数据块中每组第一测试数据块的校验序列中非零元素的数量，或者所述多组第一测试数据块的特性值为所述多组第一测试数据块中每组第一测试数据块中不可纠错测试数据块的数量，或者所述多组第一测试数据块的特性值为所述多组第一测试数据块中每组第一测试数据块中重检验位与原始校验位不相同的测试数据块的数量时，所述同步可能性指标可以与所述非零元素的数量或者不可纠错测试数据块的数量或者重检验位与原始校验位不相同的测试数据块的数量负相关；在此不再赘述。

可选的，当所述多组第一测试数据块中，同步可能性指标的数值最大的观察比特的数量大于1个时，可以根据其他条件选择其中一个观察比特作为所述最可能位于同步位置的观察比特。例如，从同步可能性指标的数值最大的观察比特中，随机选取一个观察比特作为所述最可能位于同步位置的观察比特。例如，从同步可能性指标的数值最大的观察比特

中,选取在所述数据序列中排序最前的观察比特作为所述最可能位于同步位置的观察比特。

S230, 第二阶段, 对所述备选比特进行验证, 当验证成功时, 确定所述备选比特的位置为所述同步位置。具体的, S230 可以包括 S231 和 S232 两个步骤。

5 S231, 根据所述备选比特, 在数据序列中划分出一组第二测试数据块, 该组第二测试数据块包括至少一个第二测试数据块。所述备选比特即所述最可能位于同步位置的观察比特。

10 可选的, 在第二阶段中, 该方法在根据所述备选比特在所述数据序列中划分出至少一个第二测试数据块时, 可以仅利用所述数据序列中的一部分, 例如所述数据序列中的第二子序列。根据所述备选比特在数据序列中划分出一组第二测试数据块, 也就是根据所述备选比特在第二子序列中划分出一组第二测试数据块。

15 可选的, 该第二子序列与第一阶段中的第一子序列可以是相同的, 即第二子序列与第一子序列的所有比特均相同, 两个子序列完全重合。可选的, 该第二子序列与第一阶段中的第一子序列也可以是部分相同的, 即第二子序列和第一子序列的部分比特相同, 两个子序列部分重合。可选的, 该第二子序列与第一阶段中的第一子序列也可以是不同的, 即第二子序列和第一子序列的所有比特均不同, 两个子序列没有交集。

当所述第一阶段和所述第二阶段所使用的数据之间交集越少时, 所述第一阶段和所述第二阶段所使用的数据之间的独立性越强, 码字同步结果的准确性更高。因此, 当第一子序列与第二子序列不同时, 码字同步结果的准确性更高。

20 在第二阶段中, 根据所述备选比特, 即所述最可能位于同步位置的观察比特, 在第二子序列中划分出一组第二测试数据块, 该组第二测试数据块包括至少一个第二测试数据块。

其中, 根据所述备选比特, 在第二子序列中划分出一组第二测试数据块, 可以是将所述备选比特作为所述至少一个第二测试数据块的起始比特, 从而划分出所述至少一个第二测试数据块。

25 根据所述最可能位于同步位置的观察比特, 在第二子序列中划分出一组第二测试数据块, 也可以是将所述最可能位于同步位置的观察比特作为测试数据块的起始比特, 以将所述第二子序列划分为多个测试数据块, 在所述多个测试数据块选取至少一个测试数据块作为所述至少一个第二测试数据块。此时, 所述至少一个第二测试数据块与所述最可能位于同步位置的观察比特之间间隔若干测试数据块, 且所述至少一个第二测试数据块可以位于所述最可能位于同步位置的观察比特之前, 也可以位于所述最可能位于同步位置的观察比特之后。

30 可选的, 所述至少一个第二测试数据块可以是连续的, 即至少一个第二测试数据块包括连续的 N 个数据块。当每个测试数据块的长度为 n 比特时, 即选取其后的连续 $N*n$ 个比特, 即获得所述至少一个第二测试数据块。

35 可选的, 所述至少一个第二测试数据块也可以是不连续的, 所述至少一个第二测试数据块中每个测试数据块的起始位置均与备选比特相关联。具体的, 每个测试数据块的起始位置与备选比特的间隔均为码字的长度的整数倍, 即所述至少一个第二测试数据块中每两个测试数据块的间隔均为码字的长度的整数倍。并且, 所述至少一个第二测试数据块中每两个测试数据块的间隔可以相同, 也可以不同。

S232, 对所述至少一个第二测试数据块的特性值进行验证, 当验证成功时, 确定所述

备选比特的位置为所述同步位置。

可选的，对所述至少一个第二测试数据块的特性值进行验证，验证方式可以是提前终止的验证方式。具体的，依次累计所述至少一个第二测试数据块中各个第二测试数据块的特性值以获得累计值，直至所述累计值满足同步条件时，验证成功，确定所述备选比特的

5 位置为所述同步位置。

其中，所述同步条件和所述特性值相关联，具体的，

所述特性值为校验序列中全零序列的数量，所述同步条件为所述累计值大于或等于同步阈值；或者，

10 所述特性值为校验序列中零元素的数量，所述同步条件为所述累计值大于或等于同步阈值；或者，

所述特性值为可纠错测试数据块的数量，所述同步条件为所述累计值大于或等于同步阈值；或者，

所述特性值为重检验位与原始校验位相同的测试数据块的数量，所述同步条件为所述累计值大于或等于同步阈值；

15 其中，所述同步阈值的值也可以通过仿真分析获得。并且，该步骤 230 中的同步阈值与上述步骤 2222 中的同步阈值，可以为相同取值，也可以为不相同取值。

可选的，对所述至少一个第二测试数据块的特性值进行验证，验证方式可以是遍历的验证方式。具体的，将所述至少一个第二测试数据块中所有第二测试数据块的特性值相加以获得总值，当所述总值满足同步条件时，验证成功，确定所述备选比特的位置为所述同

20 步位置。

其中，所述同步条件和所述特性值相关联，具体的，

所述特性值为校验序列中全零序列的数量，所述同步条件为所述总值大于或等于同步阈值；或者，

25 所述特性值为校验序列中非全零序列的数量，所述同步条件为所述总值小于或等于同步阈值；或者，

所述特性值为校验序列中零元素的数量，所述同步条件为所述总值大于或等于同步阈值；或者，

所述特性值为校验序列中非零元素的数量，所述同步条件为所述总值小于或等于同步阈值；或者，

30 所述特性值为可纠错测试数据块的数量，所述同步条件为所述总值大于或等于同步阈值；或者，

所述特性值为不可纠错测试数据块的数量，所述同步条件为所述总值小于或等于同步阈值；或者，

35 所述特性值为重检验位与原始校验位相同的测试数据块的数量，所述同步条件为所述总值大于或等于同步阈值；或者，

所述特性值为重检验位与原始校验位不相同的测试数据块的数量，所述同步条件为所述总值小于或等于同步阈值。

其中，所述同步阈值的值也可以通过仿真分析获得。并且，该步骤 230 中的同步阈值与上述步骤 2222 中的同步阈值，可以为相同取值，也可以为不相同取值。

在所述提前终止的验证方式和所述遍历的验证方式下,针对特性值的计算都可以并行进行,即采用多个计算单元同时计算多个测试数据块的特性值。例如,采用 10 个计算单元同时计算 10 个测试数据块的特性值。并行的计算方式能够节约计算时间,降低时延。

图 8 至图 9 示出了确定所述数据序列中最可能位于同步位置的观察比特,以及验证所述最可能位于同步位置的观察比特的位置是否为同步位置的一个示例性过程。图 8 所示的线性分组码为 BCH(360, 340)码,即其中每个码字的长度为 360 比特,每个码字内的信息数据的长度为 340 比特。该 BCH(360, 340)码的调制方式为非归零码 (Non-Return-to-Zero, NRZ),调制阶数 M 为 2,则每个码元包括 1 个比特。针对该 BCH(360, 340)码的校验矩阵如下所示:

$$H = \begin{bmatrix} \alpha^{359} & \alpha^{358} & \dots & \alpha^2 & \alpha^1 & 1 \\ (\alpha^2)^{359} & (\alpha^2)^{358} & \dots & (\alpha^2)^2 & (\alpha^2)^1 & 1 \\ (\alpha^3)^{359} & (\alpha^3)^{358} & \dots & (\alpha^3)^2 & (\alpha^3)^1 & 1 \\ (\alpha^4)^{359} & (\alpha^4)^{358} & \dots & (\alpha^4)^2 & (\alpha^4)^1 & 1 \end{bmatrix},$$

其中 α 为伽罗华域 $GF(2^{10})$ 的本原元 (primitive element)。

图 8 示出了接收端所接收到的数据序列的一部分,需要确认该数据序列中码字的起始位置,即同步位置。结合以上描述,在所述第一阶段,确定该数据序列中最可能位于同步位置的观察比特。如图 8 所示,在第一阶段中,每间隔 $T \cdot \log_2 M$ 个比特,选取调制码元的起始比特,作为观察比特,所述 T 为 1,即每间隔 1 个比特选取调制码元的起始比特,作为观察比特。因而,选取连续的 360 个比特作为观察比特,即所述观察比特的数量 P 为 360,如图 8 所示的 $b_1, b_2, b_3, \dots, b_{359}, b_{360}$ 。根据这些观察比特,从数据序列中划分出多组第一测试数据块,每组第一测试数据块中测试数据块的数量 N 为 2,即每组第一测试数据块包括 2 个测试数据块,如图 8 所示的多组 B_1, B_2 。

结合以上描述,图 8 所示的数据序列中多个观察比特的位置中,只有一个观察比特的位置是该数据序列真正的同步位置。将所述每组第一测试数据块的校验序列中全零序列的数量,作为选取所述最可能位于同步位置的观察比特的依据。如图 8 所示,校验序列 S 是一个长度为 4 的行向量,例如,以观察比特 b_1 为起始比特的一组第一测试数据块的校验序列集合为 $S^{(1)}$,包括 $S_1^{(1)}, S_2^{(1)}$,分别对应测试数据块 B_1, B_2 ,类似的,以观察比特 $b_2, b_3, \dots, b_{359}, b_{360}$ 为起始比特的一组第一测试数据块的校验序列集合,分别为 $S^{(2)}, S^{(3)}, \dots, S^{(359)}, S^{(360)}$ 。统计每组第一测试数据块的校验序列中全零序列的数量,分别记为 $Q_1, Q_2, Q_3, \dots, Q_{359}, Q_{360}$ 。如图 8 所示,其中 $Q_1=0$,即以观察比特 b_1 为起始比特的一组第一测试数据块的校验序列中没有全零序列,类似的, $Q_2=0, Q_3=1, \dots, Q_{359}=2, Q_{360}=0$ 。假设最终在所有的 360 个 Q 值中 Q_{359} 的值最大,则将观察比特 b_{359} 作为所述最可能位于同步位置的观察比特。即认为观察比特 b_{359} 为一个测试数据块的起始比特,其后每隔 360 比特均为后续测试数据块的起始比特。

在选取所述最可能位于同步位置的观察比特为 b_{359} 之后,在第一阶段所使用的数据序列的部分之后,即在所述第一子序列之后,根据该 b_{359} 划分出一组第二测试数据块。即在第二子序列中划分出一组第二测试数据块,所述第二子序列与所述第一子序列不同。如图 9 所示,该组第二测试数据块包括 4 个测试数据块。对所述至少一个第二测试数据块的特性值进行验证,所述特性值为所述至少一个第二测试数据块的校验序列中全零序列的数量。

采用上述遍历的验证方式,即将所述至少一个第二测试数据块中所有第二测试数据块的校验序列中全零序列的数量相加以获得总值,当所述总值大于或等于同步阈值,验证成功,所述同步阈值为2。如图9所示,4个测试数据块的校验序列中,全零序列的数量为3,大于2,因而确定 b_{359} 所在的位置为同步位置。

- 5 可选的,当验证未成功时,重新执行上述步骤S220,即重新执行所述第一阶段,重新选取所述最可能位于同步位置的观察比特。

步骤S230为可选步骤,即该方法可以仅包括所述第一阶段。此时,在步骤S220之后直接进入步骤S240。此时,在该方法中,在确定所述备选比特后,直接将该备选比特的位置确定为所述同步位置。即直接将所述最可能的同步位置作为所述同步位置。

- 10 S240,确定所述数据序列为同步锁定状态。

接收端确定同步位置以后,可以根据该同步位置将接收到的数据序列划分为多个同步码字。当数据序列根据同步位置被划分为多个同步码字时,可以认为该数据序列处于同步锁定位置。也就是说,在确定同步位置以后,接收端确定所述数据序列为同步锁定状态。当数据序列处于同步锁定状态时,接收端可以对该数据序列进行检错以及纠错等操作。

- 15 S250,失锁判断。

在接收端开始对所述数据序列进行检错、纠错等操作以后,仍然需要持续观察所述数据序列中的码字划分是否准确,即所述同步位置是否准确。当同步位置不准确时,根据该同步位置划分出的每个测试数据块的内容不再对应真正的码字,此时,也可以称为该数据序列的同步锁定状态丢失。因此,判断数据序列的同步位置是否准确的过程,也可以称为

- 20 失锁判断过程。

在失锁判断过程中,根据所述同步位置从所述数据序列中划分出多个同步码字,所述同步位置为所述多个同步码字的起始位置;对所述多个同步码字的特性值进行验证,当验证失败时,确定所述数据序列为所述失锁状态。

- 25 可选的,所述多个同步码字可以是连续的,即多个同步码字包括连续的N个码字。当每个码字的长度为n比特时,即选取其后的连续 $N*n$ 个比特,即获得所述多个同步码字。

可选的,所述多个同步码字也可以是不连续的,所述多个同步码字中每个码字的起始位置均与同步位置相关联。具体的,每个同步码字的起始位置与同步位置的间隔均为码字的长度的整数倍,即所述多个同步码字中每两个同步码字的间隔均为码字的长度的整数倍。并且,所述多个同步码字中每两个同步码字的间隔可以相同,也可以不同。

- 30 可选的,对所述多个同步码字的特性值进行验证,验证方式可以是提前终止的验证方式。具体的,依次累计所述多个同步码字中各个同步码字的特性值以获得累计值,直至所述累计值满足失锁条件时,验证失败,确定所述备选比特的位置为所述同步位置。

其中,所述失锁条件和所述特性值相关联,具体的,

- 35 所述特性值为校验序列中非全零序列的数量,所述失锁条件为所述累计值大于或等于同步阈值;或者,

所述特性值为校验序列中非零元素的数量,所述失锁条件为所述累计值大于或等于同步阈值;或者,

所述特性值为不可纠错码字的数量,所述失锁条件为所述累计值大于或等于同步阈值;或者,

所述特性值为重检验位与原始校验位不相同的码字的数量,所述失锁条件为所述累计值大于或等于同步阈值;

其中,所述码字的长度为 n 比特,所述码字的前 k 比特为信息位,所述码字的后 $n-k$ 比特为所述原始校验位,所述重检验位是根据所述信息位获得的,所述重校验位的长度为 $n-k$ 比特,所述 n 和所述 k 为整数。

其中,所述同步阈值的值也可以通过仿真分析获得。并且,该步骤 250 中的同步阈值与上述步骤 230, 2222 中的同步阈值,可以为相同取值,也可以为不相同取值。

可选的,对所述多个同步码字的特性值进行验证,验证方式可以是遍历的验证方式。具体的,将所述多个同步码字中所有同步码字的特性值相加以获得总值,当所述总值满足失锁条件时,验证失败,确定所述备选比特的位置为所述同步位置。

其中,所述同步条件和所述特性值相关联,具体的,

所述特性值为校验序列中全零序列的数量,所述失锁条件为所述总值小于或等于同步阈值;或者,

所述特性值为校验序列中非全零序列的数量,所述失锁条件为所述总值大于或等于同步阈值;或者,

所述特性值为校验序列中零元素的数量,所述失锁条件为所述总值小于或等于同步阈值;或者,

所述特性值为校验序列中非零元素的数量,所述失锁条件为所述总值大于或等于同步阈值;或者,

所述特性值为可纠错码字的数量,所述失锁条件为所述总值小于或等于同步阈值;或者,

所述特性值为不可纠错码字的数量,所述失锁条件为所述总值大于或等于同步阈值;或者,

所述特性值为重检验位与原始校验位相同的码字的数量,所述失锁条件为所述总值小于或等于同步阈值;或者,

所述特性值为重检验位与原始校验位不相同的码字的数量,所述失锁条件为所述总值大于或等于同步阈值;

其中,所述码字的长度为 n 比特,所述码字的前 k 比特为信息位,所述码字的后 $n-k$ 比特为所述原始校验位,所述重检验位是根据所述信息位获得的,所述重校验位的长度为 $n-k$ 比特,所述 n 和所述 k 为整数。

其中,所述同步阈值的值也可以通过仿真分析获得。并且,该步骤 250 中的同步阈值与上述步骤 230, 2222 中的同步阈值,可以为相同取值,也可以为不相同取值。

在所述提前终止的验证方式和所述遍历的验证方式下,针对特性值的计算都可以并行进行,即采用多个计算单元同时计算多个测试数据块的特性值。例如,采用 10 个计算单元同时计算 10 个测试数据块的特性值。并行的计算方式能够节约计算时间,降低时延。

图 10 示出了失锁判断的一种示例性过程。如图 10 所示,根据所述同步位置从数据序列中划分出 5 个同步码字,统计其中可纠错码字的数量。例如,在解码状态(decoding state)中,0 表示不可纠错,1 表示可纠错,则在图 10 所示的 5 个同步码字中,可纠错码字的数量为 1。若失锁阈值为 3,则可纠错码字的数量小于失锁阈值,确定 5 个同步码字满足失

锁条件，所述数据序列为失锁状态。

可选的，当验证未失败时，确定所述数据序列的同步锁定状态未丢失。

可选的，当确定所述数据序列的同步锁定状态未丢失以后，可以在所述数据序列中继续向后移动一定间隔，选取多个同步码字，继续判断该向后选取的多个同步码字是否满足失锁条件。可选的，所述向后移动一定间隔可以是移动一个测试数据块，即对每一个同步码字均以其为开始向后观察多个同步码字，以进行失锁判断。

S260，确定所述数据序列为失锁状态。

可选的，当确定所述数据序列为失锁状态时，继续确定所述数据序列的更新同步位置。具体的，重新执行所述第一阶段和所述第二阶段，将重新执行的所述第二阶段中确定的同步位置作为所述更新同步位置。如此便形成了一个同步锁定-失锁-同步锁定的闭环操作，保证了通信系统在尽可能多的时间内处于同步锁定的正常工作状态。

通过该方法能够解决 AM 同步方案中增加额外数据、级联扩展性差的技术问题。并且，无须在发送端数据流中插入额外数据，从而也无须在以太接口上需要引入空闲（Idle）码块增删机制，无须设计相应的逻辑处理单元，无须为插入的 AM 序列提前预留带宽。如果需要原有 FEC 基础上，增加一级 FEC，采用两级 FEC 级联时，可以避免原有的空闲码块增删机制无法适用于对第二级 FEC 的 AM 序列、以及针对第二级 FEC 增加 AM 序列影响高精度时钟同步性能的技术问题。通过该方法可以实现接收端数据流高精度码字同步的技术效果，其同步性能达到较高可靠性。

步骤 S240、S250 和 S260 均为可选步骤，即该方法可以仅用于确定同步位置，可以不对所述数据序列进行同步锁定状态下的检错或纠错处理，也可以不对该数据序列后续是否进入失锁状态进行判断。

通过仿真比较，本申请所提供的码字同步方法与现有的 AM 同步方案相比，在平均同步锁定时间、误锁定平均出现时间以及误失锁平均出现时间等各项性能指标上，本申请所提供的码字同步方法均优于现有的 AM 同步方案。因此，从整体来看，本申请所提供的码字同步方法可以达到比 AM 同步方案更优的同步性能。

图 11 示出了本申请实施例的码字同步方法。该方法包括如下步骤：

步骤一，接收数据序列，所述数据序列包括多个比特。所述步骤一的具体过程可以参见上述对步骤 S210 的描述。

步骤二，确定所述数据序列中的备选比特，所述备选比特包括在所述多个比特中。所述备选比特为所述最可能位于同步位置的观察比特。所述步骤二的具体过程可以参见上述对步骤 S220 的描述。

步骤三，根据所述备选比特确定同步位置，所述同步位置用于指示所述码字的起始位置。

可选的，根据所述备选比特确定同步位置可以是在选取所述最可能位于同步位置的观察比特后，进一步验证所述最可能位于同步位置的观察比特的位置是否为同步位置。所述备选比特为所述最可能位于同步位置的观察比特。此时，所述步骤三的具体过程可以参见上述对步骤 S230 的描述。此时，该方法至少包括图 2 所示的 S210、S220、S230。

可选的，根据所述备选比特确定同步位置也可以是在选取所述最可能位于同步位置的观察比特后，直接将该最可能位于同步位置的观察比特作为同步位置。所述备选比特为所

述最可能位于同步位置的观察比特。此时，该方法至少包括图 2 所示的 S210, S220。

可以看出，在上述两种根据所述备选比特确定同步位置的方式中，位于所述同步位置的比特包括在所述备选比特中，所述备选比特为所述最可能位于同步位置的观察比特。

- 可选的，在所述步骤三之后，所述方法还包括：确定所述数据序列为同步锁定状态。
5 具体过程可以参见上述对步骤 S240 的描述。

可选的，在所述步骤三之后，所述方法还包括：失锁判断。具体过程可以参见上述对步骤 S250 的描述。

可选的，所述方法还可以包括步骤四，响应于所述数据序列为失锁状态，确定所述数据序列的更新同步位置。具体过程可以参见上述对步骤 S260 的描述。

10

本申请实施例的码字同步方法由通信设备执行，所述通信设备可以是任何执行 FEC 的设备，包括但不限于路由器、交换机、服务器、以及终端设备。

- 图 12 示出了本申请实施例所涉及的通信设备的一种可能的结构示意图，参阅图 12，通信设备 1200 包括：接收单元 1201 和处理单元 1202。这些单元可以执行上述图 2-11 所示的方法的相应步骤。举例来说，
15

接收单元 1201，用于接收数据序列，所述数据序列包括多个比特。

处理单元 1202，用于确定所述数据序列中的备选比特，所述备选比特包括在所述多个比特中；根据所述备选比特确定同步位置，所述同步位置用于指示所述数据序列中码字的起始位置。

- 20 图 13 是本申请实施例所涉及的通信设备的另一种结构示意图。参见图 13，该通信设备 1300 包括至少一个处理器 1301 以及至少一个通信接口 1304，可选地，该设备 1300 还可以包括存储器 1303。

- 处理器 1301 可以是中央处理器（central processing unit, CPU）、通用处理器，数字信号处理器（digital signal processor, DSP）、专用集成电路（application-specific integrated circuit, ASIC）、现场可编程门阵列（field programmable gate array, FPGA）或者其他可编程逻辑器件、晶体管逻辑器件、硬件部件或者其任意组合。其可以实现或执行结合本申请实施例公开内容所描述的各种逻辑方框、模块和电路。所述处理器也可以是实现计算功能的组合，例如包括一个或多个微处理器组合，DSP 和微处理器的组合等等。处理器可以用于确定所述数据序列中的备选比特，以及根据所述备选比特确定同步位置。以实现本申请实施例中
30 提供的方法。

- 可选的，通信总线 1302 用于在处理器 1301、通信接口 1304 和存储器、1303 之间传递信息。总线可以是外设部件互连标准（peripheral component interconnect, 简称 PCI）总线或扩展工业标准结构（extended industry standard architecture, 简称 EISA）总线等。所述总线可以分为地址总线、数据总线、控制总线等。为便于表示，图 13 中仅用一条粗线表示，但并不表示仅有一根总线或一种类型的总线。
35

存储器 1303 可以是只读存储器（read-only memory, ROM）或可存储静态信息和指令的其它类型的静态存储设备，随机存取存储器（random access memory, RAM）或者可存储信息和指令的其它类型的动态存储设备，也可以是电可擦可编程只读存储器（electrically erasable programmable read-only memory, EEPROM）、只读光盘（compact disc read-only

Memory, CD-ROM) 或其它光盘存储、光碟存储 (包括压缩光碟、激光碟、光碟、数字通用光碟、蓝光光碟等)、磁盘存储介质或者其它磁存储设备、或者能够用于携带或存储具有指令或数据结构形式的期望的程序代码并能够由计算机存取的任何其它介质, 但不限于此。存储器 1303 可以是独立存在, 通过通信总线 1302 与处理器 1301 相连接。存储器 1303 也可以和处理器 1301 集成在一起。

5 可选地, 存储器 1303 用于存储执行本申请方案的程序代码或指令, 并由处理器 1301 来控制执行。在实现图 12 所示实施例的情况下, 且图 12 实施例中所描述的各单元为通过软件实现的情况下, 执行图 12 中各个单元的功能所需的软件或程序代码存储在存储器 1303 中。处理器 1301 用于执行存储器 1303 中存储的程序代码。程序代码中可以包括一个或多个软件模块。可选地, 处理器 1301 自身也可以存储执行本申请方案的程序代码或指令。

10 通信接口 1304, 使用任何收发器一类的装置, 用于与其它设备或通信网络通信, 通信网络可以为以太网、无线接入网(RAN)或无线局域网(wireless local area networks, WLAN)等。在本申请实施例中, 通信接口 1304 可以用于接收分段路由网络中的其他节点发送的报文, 也可以向分段路由网络中的其他节点发送报文。通信接口 1304 可以为以太网接口(Ethernet)接口、快速以太(Fast Ethernet, FE)接口、千兆以太(Gigabit Ethernet, GE)接口或异步传输模式(Asynchronous Transfer Mode, ATM)接口。

在具体实现中, 作为一种实施例, 设备 1300 可以包括多个处理器, 例如图 13 中所示的处理器 1301 和处理器 1305。这些处理器中的每一个可以是一个单核(single-CPU)处理器, 也可以是一个多核(multi-CPU)处理器。这里的处理器可以指一个或多个设备、电路、和/或用于处理数据(例如计算机程序指令)的处理核。

在具体实施例中, 网络设备 1300 中的处理器 1301 用于通过通信接口接收数据序列, 所述数据序列包括多个比特; 确定所述数据序列中的备选比特, 所述备选比特包括在所述多个比特中; 根据所述备选比特确定同步位置, 所述同步位置用于指示所述数据序列中码字的起始位置。该处理器的详细处理过程请参考上述图 2 所示实施例中的步骤 S210, S220, S230, S240, S250, S260, 图 3 所示实施例中的步骤 S221, S222, 以及图 11 所示步骤一至三的详细描述, 这里不再赘述。

25 网络设备 1300 中的通信接口用于网络设备 1300 通过网络系统接收和发送数据序列。具体的过程请参考上述图 2 所示实施例中 S210, 以及图 11 所示步骤一的详细描述, 这里不再赘述。

30 图 14 本申请实施例所涉及的通信设备的另一种结构示意图。当通信设备为网络中的转发设备, 例如路由器或交换机时, 该通信设备可以参见图 14 所示的设备结构示意图。设备 1400 包括主控板和一个或多个接口板, 主控板与接口板通信连接。主控板也称为主处理单元(main processing unit, MPU)或路由处理卡(route processor card), 主控板负责对设备 1400 中各个组件的控制和管理, 包括路由计算、设备管理和维护功能。接口板也称为线卡(line processing unit, LPU)或线卡(line card), 用于转发数据。在一些实施例中, 设备 1400 也可以包括交换网板, 交换网板与主控板、接口板通信连接, 交换网板用于转发接口板之间的数据, 交换网板也可以称为交换网板单元(switch fabric unit, SFU)。接口板包括中央处理器、存储器、转发芯片和物理接口卡(physical interface card, PIC)。

中央处理器与存储器、网络处理器和物理接口卡分别通信连接。存储器用于存储转发表。转发芯片用于基于存储器中保存的转发表转发接收到的数据帧，如果数据帧的目的地址为设备 1400 的地址，则将该数据帧上送 CPU 处理；如果数据帧的目的地址不是设备 1400 的地址，则根据该目的地址从转发表中查找到该目的地址对应的下一跳和出接口，将该数据帧转发到该目的地址对应的出接口。转发芯片可以是网络处理器（network processor, NP）。PIC 也称为子卡，可安装在接口板上，负责将光电信号转换为数据帧并对数据帧进行合法性检查后转发给转发芯片处理。在一些实施例中，中央处理器也可执行转发芯片的功能，比如基于通用 CPU 实现软件转发，从而接口板中不需要转发芯片。主控板、接口板、交换网板之间的通信连接可以通过总线来实现。在一些实施例中，转发芯片可以通过专用集成电路（application-specific integrated circuit, ASIC）或现场可编程门阵列（field programmable gate array, FPGA）实现。

在逻辑上，设备 1400 包括控制面和转发面，控制面包括主控板和中央处理器，转发面包括执行转发的各个组件，比如存储器、PIC 和 NP。控制面执行路由器、生成转发表、处理信令和协议报文、配置与维护 PE1 的状态等功能，控制面将生成的转发表下发给转发面，在转发面，NP 基于控制面下发的转发表对设备 1400 的 PIC 收到的报文查表转发。控制面下发的转发表可以保存在存储器中。在有些实施例中，控制面和转发面可以完全分离，不在同一设备上。

在具体实施例中，所述接口板用于接收数据序列，所述数据序列包括多个比特。具体的过程请参考上述图 2 所示实施例中 S210，以及图 11 所示步骤一的详细描述，这里不再赘述。所述主控板用于确定所述数据序列中的备选比特，所述备选比特包括在所述多个比特中；根据所述备选比特确定同步位置，所述同步位置用于指示所述数据序列中码字的起始位置。详细处理过程请参考上述图 2 所示实施例中的步骤 S210, S220, S230, S240, S250, S260，图 3 所示实施例中的步骤 S221, S222，以及图 11 所示步骤一至三的详细描述，这里不再赘述。

在一种可能的实现方式中，主控板和接口板之间建立进程间通信协议（inter-process communication, IPC）通道，主控板和接口板之间通过 IPC 通道进行通信。

本申请实施例提供一种芯片，该芯片包括存储器和处理器，存储器用于存储计算机程序，处理器用于从存储器中调用并运行该计算机程序，以执行上述方法实施例中的方法，即图 2-11 所示实施例中的方法。

可选地，上述芯片仅包括处理器，处理器用于读取并执行存储器中存储的计算机程序，当计算机程序被执行时，处理器执行上述方法实施例中的方法，即图 2-11 所示实施例中的方法。

可选的，所述芯片可以用于两级 FEC 级联的应用场景。具体的，两级 FEC 级联包括内层 FEC 和外层 FEC，本申请实施例提供的芯片用于所述内层 FEC。内层 FEC 采用本申请所述的码字同步方法，也称为 FEC1；外层 FEC 可以采用 AM 同步方案，也称为 FEC2。此时，所述 FEC1 与所述 FEC2 可以位于同一芯片上，也可位于不同芯片上。即本申请实施例所提供的芯片可以仅包括所述 FEC1，也可以包括所述 FEC1 和所述 FEC2。

本申请实施例提供一种网络系统。该网络系统包括接收设备和发送设备。所述接收设备用于接收所述发送设备发送的数据序列。所述接收设备可以执行上述图 2-11 所示实施

例中的步骤。

本申请实施例还提供了一种非瞬态存储介质，用于储存前述实施例中所用的软件指令，其包括用于执行前述实施例所示的方法的程序，当其在计算机或网络设备上执行时，使得所示计算机或网络设备执行前述方法实施例中的方法。

- 5 本申请实施例还提供了一种包括计算机程序指令的计算机程序产品，当该计算机程序产品在计算机上运行时，使得网络节点执行前述方法实施例中的方法。

需说明的是，以上描述的任意装置实施例都仅仅是示意性的，其中所述作为分离部件说明的单元可以是或者也可以不是物理上分开的，作为单元显示的部件可以是或者也可以不是物理单元，即可以位于一个地方，或者也可以分布到多个网络单元上。可以根据实际的
10 的需要选择其中的部分或者全部模块来实现本实施例方案的目的。另外，本申请提供的网络设备或主机的实施例附图中，模块之间的连接关系表示它们之间具有通信连接，具体可以实现为一条或多条通信总线或信号线。本领域普通技术人员在不付出创造性劳动的情况下，即可以理解并实施。

本申请实施例公开内容所描述的方法或者算法的步骤可以硬件的方式来实现，也可以
15 是由处理器执行软件指令的方式来实现。软件指令可以由相应的软件模块组成，软件模块可以被存放于随机存取存储器（random access memory, RAM）、闪存、只读存储器（read-only memory, ROM）、可擦除可编程只读存储器（erasable programmable ROM, EPROM）、电可擦可编程只读存储器（electrically EPROM, EEPROM）、硬盘、移动硬盘、光盘或者本领域熟知的任何其它形式的存储介质中。存储介质耦合至处理器，从而使处理器能够从
20 该存储介质读取信息。当然，存储介质也可以是处理器的组成部分。

以上所述的具体实施方式，对本申请的目的、技术方案和有益效果进行了进一步详细说明，所应理解的是，以上所述仅为本申请的具体实施方式而已，并不用于限定本申请的保护范围，凡在本申请的技术方案的基础之上，所做的任何修改、替换、改进等，均应包括在本申请的保护范围之内。

25 本申请中术语“第一”“第二”等字样用于对作用和功能基本相同的相同项或相似项进行区分，应理解，“第一”、“第二”、“第n”之间不具有逻辑或时序上的依赖关系，也不对数量和执行顺序进行限定。还应理解，尽管以下描述使用术语第一、第二等来描述各种元素，但这些元素不应受术语的限制。这些术语只是用于将一元素与另一元素区别分开。例如，在不脱离各种所述示例的范围的情况下，第一图像可以被称为第二图像，并且类似地，
30 第二图像可以被称为第一图像。第一图像和第二图像都可以是图像，并且在某些情况下，可以是单独且不同的图像。

本申请中术语“至少一个”的含义是指一个或多个，本申请中术语“多个”的含义是指两个或两个以上，例如，多个第二报文是指两个或两个以上的第二报文。本文中术语“系统”和“网络”经常可互换使用。

35 应理解，在本文中对各种所述示例的描述中所使用的术语只是为了描述特定示例，而并非旨在进行限制。如在对各种所述示例的描述和所附权利要求书中所使用的那样，单数形式“一个（“a”，“an”）”和“该”旨在也包括复数形式，除非上下文另外明确地指示。

还应理解，本文中所使用的术语“和/或”是指并且涵盖相关联的所列出的项目中的一个或多个项目的任何和全部可能的组合。术语“和/或”，是一种描述关联对象的关联关系，

表示可以存在三种关系，例如，A 和/或 B，可以表示：单独存在 A，同时存在 A 和 B，单独存在 B 这三种情况。另外，本申请中的字符“/”，一般表示前后关联对象是一种“或”的关系。

还应理解，在本申请的各个实施例中，各个过程的序号的大小并不意味着执行顺序的先后，各过程的执行顺序应以其功能和内在逻辑确定，而不应对本申请实施例的实施过程构成任何限定。

5 应理解，根据 A 确定 B 并不意味着仅仅根据 A 确定 B，还可以根据 A 和/或其它信息确定 B。

还应理解，术语“包括”(也称“includes”、“including”、“comprises”和/或“comprising”)当
10 在本说明书中使用指定存在所陈述的特征、整数、步骤、操作、元素、和/或部件，但是并不排除存在或添加一个或多个其他特征、整数、步骤、操作、元素、部件、和/或其分组。

还应理解，术语“如果”可被解释为意指“当 ...时”(“when”或“upon”)或“响应于确定”或
15 “响应于检测到”。类似地，根据上下文，短语“如果确定 ...”或“如果检测到[所陈述的条件或事件]”可被解释为意指“在确定 ...时”或“响应于确定 ...”或“在检测到[所陈述的条件或事件] 时”或“响应于检测到[所陈述的条件或事件]”。

应理解，说明书通篇中提到的“一个实施例”、“一实施例”、“一种可能的实现方式”意味着与实施例或实现方式有关的特定特征、结构或特性包括在本申请的至少一个实施例中。
20 因此，在整个说明书各处出现的“在一个实施例中”或“在一实施例中”、“一种可能的实现方式”未必一定指相同的实施例。此外，这些特定的特征、结构或特性可以任意适合的方式结合在一个或多个实施例中。

25