

实现码字同步的方法、通信设备、芯片和芯片系统

本申请要求于 2022 年 01 月 30 日提交的申请号为 202210113601.8、发明名称为“一种基于状态机的分组码自同步实现方法”和 2022 年 05 月 12 日提交的申请号为 202210520888.6、发明名称为“实现码字同步的方法、通信设备、芯片和芯片系统”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

本申请涉及通信技术领域，特别涉及一种实现码字同步的方法、通信设备、芯片和芯片系统。

背景技术

目前随着通信技术的发展，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, FEC)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 中涉及分组码(block code)，分组码包括线性分组码(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, MAC)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(code word)进行，因而需要在数据序列中确定码字边界，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(code word 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 200/400GE 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发送的数据流中插入了额外数据，增加了冗余信息。

发明内容

本申请提供了一种实现码字同步的方法、通信设备、芯片和芯片系统，不需要插入 AM，就能够实现码字同步，能够节约传输资源。

第一方面，本申请提供了一种实现码字同步的方法，该方法包括：响应于启动信号，进入同步位置确定状态；在该同步位置确定状态下，确定接收到的数据序列中的同步位置，该同步位置用于指示该数据序列中码字的起始位置；响应于确定出该同步位置，进入失锁检测状态；在该失锁检测状态下，对基于该同步位置选取的多个码字进行验证，响应于验证失败，重新进入该同步位置确定状态。

本申请所示的方案中，响应于启动信号，进入同步位置确定状态，在同步位置确定状态下，确定出同步位置。在确定出同步位置后，进入失锁检测状态。在失锁检测状态下，一旦判断同步位置不满足验证条件，即重新确定同步位置。这样，该同步方法为自同步方法，无

需在发送端插入 AM, 就能够实现码字同步, 能够节约传输资源。而且在检测到同步位置不准确时, 可以重新确定同步位置, 因此, 也能够使得码字持续同步。而且还使用状态机实现码字持续同步, 能够更准确实现码字持续同步。

在一种可能的实现方式中, 在该同步位置确定状态下, 确定接收到的数据序列中的同步位置, 包括: 在该同步位置确定状态下, 在该数据序列中选取 N 个观察比特, 基于该 N 个观察比特所在的位置, 确定该同步位置, N 为大于或等于 1 的整数。

本申请所示的方案中, 观察比特属于数据序列, 观察比特所在的位置为数据序列中可能为同步位置的位置。

在一种可能的实现方式中, 该基于该 N 个观察比特所在的位置, 确定该同步位置, 包括: 在该 N 个观察比特所在的位置中, 选取该同步位置。

本申请所示的方案中, 可以在 N 个观察比特所在的位置中, 直接确定出同步位置。

在一种可能的实现方式中, 该同步位置确定状态包括第一计数器重置子状态和第一码字验证子状态; 该在该同步位置确定状态下, 在该数据序列中选取 N 个观察比特, 在该 N 个观察比特所在的位置中, 选取该同步位置, 包括: 在该第一计数器重置子状态下, 设置第一码字计数器的初始值, 在该数据序列中选取一个观察比特; 响应于基于当前的观察比特从该数据序列中选取出一个第一测试块, 进入该第一码字验证子状态, 该第一测试块的长度等于码字长度, 该第一测试块的起始位置与该当前的观察比特所在的位置之间间隔整数个该码字长度; 在该第一码字验证子状态下, 对该第一测试块进行验证; 响应于该第一测试块满足验证条件, 将该第一码字计数器的计数值增加 X, X 为大于或等于 1 的整数; 响应于该第一码字计数器的计数值等于第一阈值, 确定该当前的观察比特所在的位置为该同步位置。

本申请所示的方案中, 在使用当前的观察比特选取出的测试块中, 累计连续第一阈值个测试块满足验证条件, 确定当前的观察比特所在的位置为同步位置。这样, 由于状态机的状态比较少, 所以能够快速确定出同步位置。

在一种可能的实现方式中, 该在该同步位置确定状态下, 在该数据序列中选取 N 个观察比特, 在该 N 个观察比特所在的位置中, 选取该同步位置, 还包括: 响应于该第一码字计数器的计数值小于该第一阈值, 重新进入该第一码字验证子状态, 执行对基于该当前的观察比特选取的下一个第一测试块的验证操作。

在一种可能的实现方式中, 该同步位置确定状态还包括第一码字有效子状态; 该响应于该第一测试块满足该验证条件, 将该第一码字计数器的计数值增加 X, 包括: 响应于该第一测试块满足该验证条件, 进入该第一码字有效子状态; 在该第一码字有效子状态下, 将该第一码字计数器的计数值增加 X。

在一种可能的实现方式中, 该在该同步位置确定状态下, 在该数据序列中选取 N 个观察比特, 在该 N 个观察比特所在的位置中, 选取该同步位置, 还包括: 响应于该第一测试块不满足该验证条件, 执行对从该数据序列中选取的下一个观察比特的验证操作。

本申请所示的方案中, 在基于当前的观察比特选取出第一测试块后, 若第一测试块不满足验证条件, 则说明当前的观察比特所在的位置不是同步位置, 继续判断下一个观察比特是否是同步位置。这样, 可以快速确定出同步位置。

在一种可能的实现方式中, 该同步位置确定状态还包括第一移位子状态; 该响应于该第一测试块不满足该验证条件, 执行对从该数据序列中选取的下一个观察比特的验证操作, 包

括：响应于该第一测试块不满足验证条件，进入该第一移位子状态；在该第一移位子状态下，移位至该下一个观察比特；响应于移位至该下一个观察比特，重新进入该第一计数器重置子状态，执行对该下一个观察比特的验证操作。

本申请所示的方案中，在第一测试块不满足验证条件时，进入第一移位子状态，进行移位处理，移位至下一个观察比特，重新进入第一计数器重置子状态。

在一种可能的实现方式中，该同步位置确定状态还包括第一同步锁定的初始化子状态；该在该第一计数器重置子状态下，设置第一码字计数器的初始值之前，还包括：在该第一同步锁定的初始化子状态下，将第一同步锁定变量的值置为第一值。

在一种可能的实现方式中，该同步位置确定状态还包括第一同步锁定的成功子状态；该响应于该第一码字计数器的计数值等于第一阈值，确定该当前的观察比特所在的位置为该同步位置，包括：响应于该第一码字计数器的计数值等于该第一阈值，进入该第一同步锁定的成功子状态；在该第一同步锁定的成功子状态下，将该第一同步锁定变量的值置为第二值，该第一同步锁定变量的值为该第二值，用于指示该当前的观察比特所在位置为该同步位置。

在一种可能的实现方式中，该方法还包括：在该第一同步锁定的初始化子状态下，将该第一码字待验证变量的值置为第三值；该响应于基于当前的观察比特从该数据序列中选取出一个第一测试块，进入该第一码字验证子状态，包括：响应于基于当前的观察比特从该数据序列中选取出一个第一测试块，将该第一码字待验证变量的值置为第四值；响应于该第一码字待验证变量的值为第四值，且该第一同步锁定变量的值为第一值，进入该第一码字验证子状态。

在一种可能的实现方式中，该失锁检测状态包括第二码字验证子状态；该在该失锁检测状态下，对基于该同步位置选取的多个码字进行验证，包括：响应于确定出该同步位置，重新进入该第一计数器重置子状态；响应于基于该同步位置，选取出一个第一码字，进入该第二码字验证子状态；在该第二码字验证子状态下，对该第一码字进行验证。

在一种可能的实现方式中，该在该失锁检测状态下，对基于该同步位置选取的多个码字进行验证，还包括：在该第一计数器重置子状态下，设置该第一码字计数器的初始值和第一无效码字计数器的初始值；该响应于验证失败，重新进入该同步位置确定状态，包括：响应于该第一码字不满足验证条件，将该第一码字计数器的计数值增加 Y ，将该第一无效码字计数器的计数值增加 Z ， Y 和 Z 均为大于或等于 1 的整数；响应于在该第一码字计数器的计数值小于或等于第二阈值的情况下，该第一无效码字计数器的计数值等于第三阈值，重新进入该同步位置确定状态，该第三阈值小于该第二阈值。

本申请所示的方案中，同步位置确定过程和失锁检测过程，能够共用部分状态，因此，可以简化实现码字持续同步的流程。

在一种可能的实现方式中，该同步位置确定状态包括第一移位子状态；该重新进入该同步位置确定状态，包括：进入该第一移位子状态；在该第一移位子状态下，移位至下一个观察比特；响应于移位至下一个观察比特，重新进入该第一计数器重置子状态，执行对该下一个观察比特的验证操作。

在一种可能的实现方式中，该基于该 N 个观察比特所在的位置，确定该同步位置，包括：在该 N 个观察比特所在的位置中，选取该数据序列中的备选位置；基于该备选位置，确定该同步位置。

本申请所示的方案中，可以先在N个观察比特所在的位置中，选取出最有可能为同步位置的位置，即备选位置，然后再使用备选位置，确定同步位置，使得确定出的同步位置更准确。

在一种可能的实现方式中，该N大于1，该同步位置确定状态包括第二计数器重置子状态和第三码字验证子状态；该在该同步位置确定状态下，在该数据序列中选取N个观察比特，在该N个观察比特所在的位置中，选取该数据序列中的备选位置，包括：在该第二计数器重置子状态下，设置第二码字计数器的初始值和第一当前有效码字计数器的初始值，从该数据序列中选取一个观察比特；响应于基于当前的观察比特从该数据序列中选取出一个第二测试块，进入该第三码字验证子状态，该第二测试块的长度等于码字长度，该第二测试块的起始位置与该当前的观察比特所在的位置之间间隔整数个该码字长度；在该第三码字验证子状态下，对该第二测试块进行验证，将该第二码字计数器的计数值增加W，W为大于或等于1的整数，响应于该第二测试块满足验证条件，将该第一当前有效码字计数器的计数值增加P，P为大于或等于1的整数；响应于该第二码字计数器的计数值等于第四阈值，将观察比特数目变量的值增加1，响应于该第一当前有效码字计数器的计数值大于最大有效码字数目变量的当前值，更新备选同步位置变量的值为该当前的观察比特所在的位置，将该最大有效码字数目变量的当前值更新为该第一当前有效码字计数器的计数值；响应于该观察比特数目变量的值等于N，将该当前的备选同步位置变量的值确定为该备选位置。

本申请所示的方案中，连续验证N个观察比特，选取N个观察比特中满足验证条件的测试块最多的观察比特，将该观察比特所在的位置确定为备选位置。这样，由于对每个观察比特进行验证，选取备选位置，所以可以使得确定出的备选位置更准确，也就是说使得确定出的备选位置更有可能是同步位置。

在一种可能的实现方式中，该同步位置确定状态还包括备选位置选择子状态；该响应于该第二码字计数器的计数值等于第四阈值，将观察比特数目变量的值增加1，响应于该第一当前有效码字计数器的计数值大于最大有效码字数目变量的当前值，更新备选同步位置变量的值为该当前的观察比特所在的位置，将该最大有效码字数目变量的当前值更新为该第一当前有效码字计数器的计数值，包括：响应于该第二码字计数器的计数值等于第四阈值，进入该备选位置选择子状态；在该备选位置选择子状态下，响应于该第二码字计数器的计数值等于第四阈值，将观察比特数目变量的值增加1，响应于该第一当前有效码字计数器的计数值大于最大有效码字数目变量的当前值，更新备选同步位置变量的值为该当前的观察比特所在的位置，将该最大有效码字数目变量的当前值更新为该第一当前有效码字计数器的计数值。

在一种可能的实现方式中，该在该同步位置确定状态下，在该数据序列中选取N个观察比特，在该N个观察比特所在的位置中，选取该数据序列中的备选位置，还包括：响应于该观察比特数目变量的值小于N，执行对从该数据序列中选取的下一个观察比特的验证操作。

在一种可能的实现方式中，该同步位置确定状态还包括第二移位子状态；该响应于该观察比特数目变量的值小于N，执行对从该数据序列中选取的下一个观察比特的验证操作，包括：响应于该观察比特数目变量的值小于N，进入该第二移位子状态；在该第二移位子状态下，移位至该下一个观察比特；响应于移位至该下一个观察比特，重新进入该第二计数器重置子状态，执行对该下一个观察比特的验证操作。

在一种可能的实现方式中，该同步位置确定状态还包括第二同步锁定的初始化子状态；

该在该第二计数器重置子状态下，设置第二码字计数器的初始值和第一当前有效码字计数器的初始值之前，还包括：在该第二同步锁定的初始化子状态下，将该观察比特数目变量的值置为第五值，将该备选同步位置变量的值置为第六值，将该最大有效码字数目变量的值置为第七值。

在一种可能的实现方式中，该方法还包括：在该第二同步锁定的初始化子状态下，将第二码字待验证变量的值置为第三值；该响应于基于当前的观察比特从该数据序列中选取出一个第二测试块，进入该第三码字验证子状态，包括：响应于基于该当前的观察比特从该数据序列中选取一个第二测试块，将该第二码字待验证变量的值置为第四值；响应于该第二码字待验证变量的值为该第四值，进入该第三码字验证子状态。

在一种可能的实现方式中，该在该同步位置确定状态下，在该数据序列中选取N个观察比特，在该N个观察比特所在的位置中，选取该数据序列中的备选位置，还包括：响应于该第二码字计数器的计数值小于该第四阈值，重新进入该第三码字验证子状态，执行对基于该当前的观察比特选取的下一个第二测试块的验证操作。

在一种可能的实现方式中，该同步位置确定状态包括第三计数器重置子状态和第四码字验证子状态；该在该同步位置确定状态下，在该数据序列中选取N个观察比特，在该N个观察比特所在的位置中，选取该数据序列中的备选位置，包括：在该第三计数器重置子状态下，设置第三码字计数器的初始值和第一有效码字计数器的初始值，在该数据序列中选取一个观察比特；响应于基于当前的观察比特从该数据序列中选取出一个第三测试块，进入该第四码字验证子状态，该第三测试块的长度等于码字长度，该第三测试块的起始位置与该当前的观察比特所在的位置之间间隔整数个该码字长度；在该第四码字验证子状态下，对该第三测试块进行验证；响应于该第三测试块满足验证条件，将该第三码字计数器的计数值增加M，将该第一有效码字计数器的计数值增加Q，M和Q均为大于或等于1的整数；响应于在该第三码字计数器的计数值小于或等于第五阈值的情况下，该第一有效码字计数器的计数值等于第六阈值，确定该当前的观察比特所在的位置为该备选位置，该第五阈值大于或等于该第六阈值。

本申请所示的方案中，依次验证N个观察比特，在基于某个观察比特选取的测试块中，满足验证条件的测试块的数目累计达到一定数目时，确定该观察比特所在的位置为备选位置。采用这种方式，有可能不需要验证全部观察比特，所以可以快速确定出备选位置。

在一种可能的实现方式中，该同步位置确定状态还包括第二码字有效子状态；该响应于该第三测试块满足验证条件，将所述第三码字计数器的计数值增加M，将所述第一有效码字计数器的计数值增加Q，包括：响应于该第三测试块满足验证条件，进入该第二码字有效子状态；在该第二码字有效子状态下，将该第三码字计数器的计数值增加M，将该第一有效码字计数器的计数值增加Q。

在一种可能的实现方式中，该在该同步位置确定状态下，在该数据序列中选取N个观察比特，在该N个观察比特所在的位置中，选取该数据序列中的备选位置，还包括：在该第二码字有效子状态下，响应于该第三码字计数器的计数值等于该第五阈值，且该第一有效码字计数器的计数值小于该第六阈值，执行对从该数据序列中选取的下一个观察比特的验证操作。

在一种可能的实现方式中，该在该同步位置确定状态下，在该数据序列中选取N个观察比特，在该N个观察比特所在的位置中，选取该数据序列中的备选位置，还包括：在该第二

码字有效子状态下，响应于该第三码字计数器的计数值小于该第五阈值，且该第一有效码字计数器的计数值小于该第六阈值，重新进入该第四码字验证子状态，执行对基于该当前的观察比特选取的下一个第三测试块的验证操作。

在一种可能的实现方式中，该同步位置确定状态还包括第一码字无效子状态；在该同步位置确定状态下，在该数据序列中选取N个观察比特，在该N个观察比特所在的位置中，选取该数据序列中的备选位置，还包括：响应于该第三测试块不满足验证条件，进入该第一码字无效子状态；在该第一码字无效子状态下，将第三码字计数器的计数值增加M；响应于该第三码字计数器的计数值等于第五阈值，执行对从该数据序列中选取的下一个观察比特的验证操作。

在一种可能的实现方式中，该在该同步位置确定状态下，在该数据序列中选取N个观察比特，基于该N个观察比特，确定该数据序列中的备选位置，还包括：在该第一码字无效子状态下，响应于该第三码字计数器的计数值小于该第五阈值，重新进入该第四码字验证子状态，执行对基于该当前的观察比特选取的下一个第三测试块的验证操作。

在一种可能的实现方式中，该同步位置确定状态还包括第三移位子状态；该执行对从该数据序列中选取的下一个观察比特的验证操作，包括：进入该第三移位子状态；在该第三移位子状态下，移位至该下一个观察比特；响应于移位至该下一个观察比特，重新进入该第三计数器重置子状态，执行对该下一个观察比特的验证操作。

在一种可能的实现方式中，该同步位置确定状态还包括第三同步锁定的初始化子状态；在该第三计数器重置子状态下，设置第三码字计数器的初始值和第一有效码字计数器的初始值之前，还包括：在该第三同步锁定的初始化子状态下，将第三码字待验证变量的值置为第三值。

在一种可能的实现方式中，该响应于基于当前的观察比特从该数据序列中选取出一个第三测试块，进入该第四码字验证子状态，包括：响应于基于当前的观察比特从该数据序列中选取出一个第三测试块，将该第三码字待验证变量的值置为第四值；响应于该第三码字待验证变量的值为该第四值，进入该第四码字验证子状态。

在一种可能的实现方式中，该基于该备选位置，确定该同步位置，包括：对该备选位置进行验证，在验证成功后，确定该备选位置为该同步位置。

本申请所示的方案中，在选取出备选位置后，对备选位置进行验证，在验证成功后，将备选位置确定为同步位置。由于还对备选位置进行验证，所以能够使得确定出的同步位置更准确。

在一种可能的实现方式中，该同步位置确定状态还包括第四计数器重置子状态和第五码字验证子状态；该对该备选位置进行验证，在验证成功后，确定该备选位置为该同步位置，包括：在该第四计数器重置子状态下，设置第四码字计数器的初始值，基于该备选位置从该数据序列中选取一个第四测试块；响应于基于该备选位置从该数据序列中选取出一个第四测试块，进入该第五码字验证子状态；在该第五码字验证子状态下，对该第四测试块进行验证；响应于该第四测试块满足验证条件，将该第四码字计数器的计数值增加R，R为大于或等于1的整数；响应于该第四码字计数器的计数值等于第七阈值，确定该备选位置为该同步位置。

本申请所示的方案中，依次验证基于备选位置选取的第四测试块，连续满足验证条件的第四测试块的数目累计达到一定数目时，确定该备选位置为同步位置。采用这种方式，能够

判断备选位置是否为同步位置。

在一种可能的实现方式中，该同步位置确定状态还包括第三码字有效子状态；该响应于该第四测试块满足验证条件，将该第四码字计数器的计数值增加 R，包括：响应于该第四测试块满足验证条件，进入该第三码字有效子状态；在该第三码字有效子状态下，将该第四码字计数器的计数值增加 R。

在一种可能的实现方式中，该对该备选位置进行验证，在验证成功后，确定该备选位置为该同步位置，还包括：响应于该第四码字计数器的计数值小于该第七阈值，重新进入该第五码字验证子状态，执行对基于该备选位置选取的下一个第四测试块的验证操作。

在一种可能的实现方式中，该对该备选位置进行验证，在验证成功后，确定该备选位置为该同步位置，还包括：响应于该第四测试块不满足该验证条件，重新选取备选位置。

本申请所示的方案中，只要有一个第四测试块不满足验证条件，即重新选取备选位置，能够使得选取出的备选位置更可能为同步位置。

在一种可能的实现方式中，该同步位置确定状态还包括第一重新同步子状态；该响应于该第四测试块不满足验证条件，重新选取备选位置，包括：响应于该第四测试块不满足该验证条件，进入该第一重新同步子状态；在该第一重新同步子状态下，将第一重新同步锁定变量的值置为第二值，该第一重新同步锁定变量的值为该第二值，用于指示重新在该数据序列中选取备选位置。

在一种可能的实现方式中，该同步位置确定状态还包括第二同步锁定的成功子状态；该响应于该第四码字计数器的计数值等于第七阈值，确定该备选位置为该同步位置，包括：响应于该第四码字计数器的计数值等于第七阈值，进入该第二同步锁定的成功子状态；在该第二同步锁定的成功子状态下，将第二同步锁定变量的值置为第二值，该第二同步锁定变量的值为该第二值，用于指示该备选位置为该同步位置。

在一种可能的实现方式中，该基于该备选位置，确定该同步位置，包括：确定该备选位置为该同步位置。

本申请所示的方案中，直接将备选位置确定为同步位置，能够快速确定出同步位置。

在一种可能的实现方式中，该在该失锁检测状态下，对基于该同步位置选取的多个码字进行验证，响应于验证失败，重新进入该同步位置确定状态，包括：在该失锁检测状态下，对基于该同步位置选取的码字集合中的码字进行验证，该码字集合包括目标数值个码字；响应于该码字集合中不满足验证条件的码字数目达到第八阈值，重新进入该同步位置确定状态。

本申请所示的方案中，在选取出同步位置后，持续判断同步位置是否正确，能够使得码字持续同步。而且由于是判断不满足验证条件的码字的数目是否达到一定数值，不满足验证条件的码字的数目比较少，所以按照不满足验证条件的码字的数目判断同步位置是否准确，能够更快速的判断同步位置是否正确。

在一种可能的实现方式中，该失锁检测状态包括第五计数器重置子状态和第六码字验证子状态；该在该失锁检测状态下，对基于该同步位置选取的码字集合中的码字进行验证，该码字集合包括目标数值个码字；响应于该码字集合中不满足验证条件的码字数目达到第八阈值，重新进入该同步位置确定状态，包括：在该第五计数器重置子状态下，设置第五码字计数器的初始值和第二无效码字计数器的初始值；响应于基于该同步位置从该数据序列中选取出一个第二码字，进入该第六码字验证子状态；在该第六码字验证子状态下，对该第二码字

进行验证；响应于该第二码字不满足验证条件，将该第五码字计数器的计数值增加 1，将该第二无效码字计数器的计数值增加 1；响应于在该第五码字计数器的计数值小于目标数值的情况下，该第二无效码字计数器的计数值等于第八阈值，重新进入该同步位置确定状态。

本申请所示的方案中，在选取出同步位置后，持续判断同步位置是否正确，能够使得码字持续同步。

在一种可能的实现方式中，该失锁检测状态还包括第二码字无效子状态；该响应于该第二码字不满足验证条件，将第五码字计数器的计数值增加 1，将该第二无效码字计数器的计数值增加 1，包括：响应于该第二码字不满足验证条件，进入该第二码字无效子状态；在该第二码字无效子状态下，将该第五码字计数器的计数值增加 1，将该第二无效码字计数器的计数值增加 1。

在一种可能的实现方式中，该在该失锁检测状态下，对基于该同步位置选取的码字集合中的码字进行验证，该码字集合包括目标数值个码字；响应于该码字集合中不满足验证条件的码字数目达到第八阈值，重新进入该同步位置确定状态，还包括：响应于该第二码字满足验证条件，将该第五码字计数器的计数值增加 1；响应于该第五码字计数器的计数值等于该目标数值，重新进入该第五计数器重置子状态，对该同步位置进行下一次验证。

在一种可能的实现方式中，该失锁检测状态还包括第四码字有效子状态；该响应于该第二码字满足验证条件，将该第五码字计数器的计数值增加 1，包括：响应于该第二码字满足验证条件，进入该第四码字有效子状态；在该第四码字有效子状态下，将该第五码字计数器的计数值增加 1。

在一种可能的实现方式中，该在该失锁检测状态下，对基于该同步位置选取的码字集合中的码字进行验证，该码字集合包括目标数值个码字；响应于该码字集合中不满足验证条件的码字数目达到第八阈值，重新进入该同步位置确定状态，还包括：在该第三码字有效子状态下，响应于该第五码字计数器的计数值小于该目标数值，重新进入该第六码字验证子状态，执行对基于该同步位置选取的下一个第二码字的验证操作。

在一种可能的实现方式中，该在该失锁检测状态下，对基于该同步位置选取的码字集合中的码字进行验证，该码字集合包括目标数值个码字；响应于该码字集合中不满足验证条件的码字数目达到第八阈值，重新进入该同步位置确定状态，还包括：在该第二码字无效子状态下，响应于该第五码字计数器的计数值小于该目标数值，且该第二无效码字计数器的计数值小于该第八阈值，重新进入该第六码字验证子状态，执行对基于该同步位置选取的下一个第二码字的验证操作。

在一种可能的实现方式中，在该失锁检测状态下，对基于该同步位置选取的码字集合中的码字进行验证，该码字集合包括目标数值个码字；响应于该码字集合中不满足验证条件的码字数目达到第八阈值，重新进入该同步位置确定状态，还包括：在该第二码字无效子状态下，响应于该第五码字计数器的计数值等于该目标数值，且该第二无效码字计数器的计数值小于该第八阈值，重新进入该第五计数器重置子状态，对该同步位置进行下一次验证。

在一种可能的实现方式中，该验证条件为测试块或码字的校验序列的零元素的数目大于同步阈值；或者，该验证条件为测试块或码字的校验序列的非零元素的数目小于非同步阈值；或者，该验证条件为测试块或码字为可纠错测试块；或者，该验证条件为测试块或码字的重校验位与原始校验位相同。

本申请所示的方案中，存在多种方式能够验证测试块或码字，使得对测试块和码字的验证更灵活。

在一种可能的实现方式中，该启动信号包括系统重置或启动信号、数据接收失败信号或重新同步信号。

在一种可能的实现方式中，该数据序列是经过 FEC 编码的比特流。这样，码字同步的方法可以应用于 FEC 编码的比特流中。

第二方面，本申请提供了一种实现码字同步的装置，该装置包括：

同步位置确定模块，用于：响应于启动信号，进入同步位置确定状态；在所述同步位置确定状态下，确定接收到的数据序列中的同步位置，所述同步位置用于指示所述数据序列中码字的起始位置；

失锁检测模块，用于：响应于确定出所述同步位置，进入失锁检测状态；在所述失锁检测状态下，对基于所述同步位置选取的多个码字进行验证，响应于验证失败，重新进入所述同步位置确定状态。

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和存储器；该存储器中存储有至少一条计算机指令；该计算机指令由该处理器加载并执行以实现第一方面或者第一方面中可能的实现方式中的实现码字同步方法。

第四方面，本申请提供了一种芯片，该芯片用于实现第一方面或者第一方面中可能的实现方式中的实现码字同步方法。

第五方面，本申请提供了一种芯片系统，该芯片系统包括级联的第一芯片和第二芯片；该第一芯片用于实现第一方面或者第一方面中可能的实现方式中的实现码字同步方法；该第二芯片用于基于 AM 方式进行码字同步。

附图说明

图 1 是本申请一个示例性实施例提供的状态机的原理示意图；

图 2 是本申请一个示例性实施例提供的码字和同步位置的示意图；

图 3 是本申请一个示例性实施例提供的应用场景图；

图 4 是本申请一个示例性实施例提供的应用场景图；

图 5 是本申请一个示例性实施例提供的应用场景图；

图 6 是本申请一个示例性实施例提供的应用场景图；

图 7 是本申请一个示例性实施例提供的码字同步的逻辑示意图；

图 8 是本申请一个示例性实施例提供的基于状态机实现码字同步的方法流程图；

图 9 是本申请一个示例性实施例提供的确定同步位置的状态图；

图 10 是本申请一个示例性实施例提供的确定同步位置和失锁检测的状态图；

图 11 是本申请一个示例性实施例提供的确定备选位置的状态图；

图 12 是本申请一个示例性实施例提供的确定备选位置的状态图；

图 13 是本申请一个示例性实施例提供的确定备选位置的状态图；

图 14 是本申请一个示例性实施例提供的确定对备选位置进行验证的状态图；

图 15 是本申请一个示例性实施例提供的确定对备选位置进行验证的状态图；

图 16 是本申请一个示例性实施例提供的失锁检测过程的状态图；

图 17 是本申请一个示例性实施例提供的失锁检测过程的状态图；
图 18 是本申请一个示例性实施例提供的确定备选位置的状态图；
图 19 是本申请一个示例性实施例提供的实现码字同步的装置结构示意图；
图 20 是本申请一个示例性实施例提供的通信设备的结构示意图；
图 21 是本申请一个示例性实施例提供的通信设备的结构示意图。

具体实施方式

为使本申请的目的、技术方案和优点更加清楚，下面将结合附图对本申请实施方式作进一步地详细描述。

下面对本申请实施例涉及的一些术语概念做解释说明。

1、码字同步，指在数据序列中，确定完整码字的起始位置和结束位置的过程。数据序列也可以称为数据流或比特流，码字同步也可以称为帧同步。

2、状态机，全称为有限状态机 (finite state machine, FSM)，是一种数学模型，通过对实现某一特定功能所需的操作进行提炼和抽象得到。状态机将功能实现所需的全部操作 (operations) 分解为有限和相互关联的状态 (state)，每个状态内包括一个或多个子操作，状态之间的转换 (transition) 由输入信号或特定参数的变化触发，特定参数存储在寄存器中，认为是寄存器变量。在任意时刻，状态机只能处于一个状态。

状态机在硬件实现时包括组合逻辑电路和寄存器，组合逻辑电路用于执行操作，寄存器用于存储寄存器变量。

状态机的表达方式有多种，其中状态图 (state diagram 或 state transition diagram) 因直观性而较为常用。在状态图中，每个状态以一个矩形表示，且被其内一条水平线分为上下两部分，上面一部分为状态名称，下面一部分为本状态内需要执行的操作。状态之间的转换用单向箭头表示，箭头上的文字描述为进行此转换需要满足的条件 (condition)，称为转换条件。图 1 提供了单个状态及其进入和退出条件的示例，在状态机进入一个状态时，先执行状态内的操作，之后再检测是否满足转换条件，若满足转换条件则进入对应的下一个状态。

在图 1 中状态以矩形表示，此处仅是一种示例，状态也可以使用圆形等其他形状表示，状态转换也可以使用其它箭头或形式表示，本申请实施例不进行限定。

下面描述本申请的背景。

由于信号损耗和噪声的影响，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 (bit) 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，冗余信息和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 按照对数据序列的处理方式不同可分为分组码和卷积码 (convolutional code)。对于分组码，可再被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。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，该冗余信息称为校验位 (parity)，最终得到长度为 n 比特的码字， n 和 k 均为整数，这样，在长度为 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，一个完整的码字由信息位和校验位组成。码字从发送端发出，经过信道传输到达接收端之后，若码字内错误比特的数目在可纠错范围内，接收端通

过解码过程即可检查并纠正错误比特，将接收到的码字恢复为发送端发送的原始数据，从而降低信道带来的干扰，能够提高通信系统的可靠性。线性分组码字的检错以及纠错功能的实现必须基于完整的码字进行，因而，在对接收端所接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的边界，即找到一个完整码字的起始位置和结束位置，此过程称为码字同步或帧同步。若码字同步不正确，即未能确定正确的起始位置，则后续解码过程中就无法达到检错或纠错的应有效果，甚至可能增加误码率，造成通信系统的性能恶化，可见码字同步至关重要。

以系统码为例，图 2 提供了码字的同步位置(synchronization position)。如图 2 所示，整个码字的长度为 n 比特，其中前 k 比特为原始数据，后 $n-k$ 比特为校验位，数据序列中的同步位置为码字的起始位置，即码字中首个比特所在的位置。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，即相邻同步位置之间的间隔是固定，该间隔等于码字的长度，简称为码字长度。

在本申请实施例中，虽然后文中以系统码为例进行说明，但是本申请并不限于系统码，也可以适用于非系统码。本申请能够适用于所有线性分组码的通信系统。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。

下面描述本申请实施例的硬件装置以及应用场景。

本申请实施例可适用于使用 FEC 的通信设备，如路由器、交换机和服务器等。

以下给出三种结合以太网标准物理层实现的应用场景。示例性的，在两级级联 FEC 场景中，将以以太网标准的里德-所罗门码(Reed-Solomon code, RS)RS 作为级联的外层 FEC(FEC1)，由于以太网标准中已经使用 AM 方案进行接收端 RS-FEC 的码字同步，本申请实施例中码字同步方案可以应用于两级级联 FEC 的内层 FEC (FEC2)。内层 FEC 也可以称为第一 FEC，外层 FEC 也可以称为第二 FEC。

应用场景 1，两级级联 FEC 分别处于不同芯片内。FEC1 所在的芯片可沿用现有标准定义的发送端和接收端结构，解码时使用 AM 方案进行码字同步。参见图 3，在发送端和接收端，FEC1 常集成于集成电路(application specific integrated circuit, ASIC)中，解码时使用 AM 方案进行码字同步，FEC2 常集成于时钟数据恢复(clock and data recovery, CDR)、重定时器(retimer)或光数字信号处理器(optical digital signal processor, oDSP)等内部，解码时使用本申请实施例的方案进行码字同步。应用场景 1 多为远距离或高速率传输场景，需要在中继节点上使用 FEC2 进行纠错能力的增强。此处，在中继节点为 oDSP 时，链路可以传输光信号。

应用场景 2，两级级联 FEC 在发送端和接收端均处于同一芯片内。参见图 4，在发送端和接收端，FEC1 集成于 ASIC 中，解码时使用 AM 方案进行码字同步，FEC2 集成于 ASIC 中，解码时使用本申请实施例的码字同步方案进行码字同步。应用场景 2 中在发送端芯片和接收端芯片上额外增加 FEC2 的物理单元。

应用场景 3，两级级联 FEC 在发送端和接收端中的其中一端处于同一芯片内，在另一端处于不同芯片内。FEC1 中解码时使用 AM 方案进行码字同步，FEC2 中解码时使用本申请实

实施例的码字同步方案进行码字同步。在该应用场景3中，两级级联FEC处于同一芯片内的一端增加FEC2的物理单元，在另一端的FEC1沿用以太网标准定义的发送端结构或接收端结构，FEC2集成于CDR、retimer或oDSP等内部。参见图5，在发送端，FEC1和FEC2集成于ASIC中，在接收端，FEC1集成于ASIC中，FEC2均集成于CDR、retimer或oDSP中。参见图6，在发送端，FEC1集成于ASIC中，FEC2均集成于CDR、retimer或oDSP中，在接收端，FEC1和FEC2集成于ASIC中。

需要说明的是，这三种应用场景是基于两级级联FEC为例进行说明，本申请实施例中的码字同步方案也可以应用于多级级联FEC的场景，而且不仅可以应用于内层FEC技术中，还可以应用于各层FEC技术中，本申请实施例不进行限定。

另外，本申请实施例也可以应用于未级联的FEC的场景，也就是说发送端和接收端均进行一次FEC，本申请实施例中的码字同步方案也可以应用于该一次FEC中。

下面描述本申请实施例中基于状态机实现码字同步的方法流程。

本申请实施例中，完整的同步流程包括两个阶段，两个阶段为同步锁定阶段和失锁检测阶段，从而形成闭环，使得通信系统接收侧尽可能处于同步锁定状态，参见图7。示例性的，在同步锁定阶段，确定出同步位置，在失锁检测阶段，持续判断同步位置是否准确，在同步位置不正确时，重新确定同步位置。

在本申请实施例中，每个状态以一个矩形表示，且被其内一条水平线分为上下两部分，上面一部分为状态名称，下面一部分为本状态内需要执行的操作，状态的表示仅是一种示例，本申请实施例对此不进行限定。在文字描述中，{ }内表示单个状态的名称，[]内表示寄存器变量的名称。在状态图中，“+”为逻辑或，“*”为逻辑与，“!”为逻辑非，“<=”为将右侧变量赋值给左侧变量，“++”为将寄存器保存的寄存器变量的值加1。UCT为无需转换条件(unconditional transition)。

图8提供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通信设备，具体为用于接收数据的接收端设备。该通信设备可以是执行FEC的各种设备，包括但不限于路由器、交换机和服务器等。该方法的步骤包括步骤801至步骤804。

步骤801，响应于检测到启动信号，进入同步位置确定状态。

在本实施例中，在进行同步位置确定时，检测到[启动信号]后，触发无需转换条件进入{同步位置确定状态}。

示例性的，[启动信号]包括系统重置信号、系统启动信号、数据接收失败或重新同步信号。系统重置信号用于指示接收端中用于解码的部分重新配置，系统启动信号用于指示接收端中用于解码的部分启动，如接收到数据序列时，会检测到系统启动信号，数据接收失败用于指示接收端解码失败，重新同步信号用于指示重新确定同步位置。

此处，系统重置信号、系统启动信号、系统接收信号或重新同步信号均为寄存器变量，分别表示为[系统重置信号]、[系统启动信号]、[系统接收信号]和[重新同步信号]。

步骤802，状态机在同步位置确定状态下，确定数据序列中的同步位置，同步位置用于指示数据序列中码字的起始位置。

其中，数据序列接收自发送端，数据序列包括多个比特，每个比特为二进制数据，数据序列也可以称为比特序列。数据序列采用线性分组码。数据序列经过信道传输，可能存在误码，需要进行检错以及纠错等操作。

在本实施例中，在{同步位置确定状态}下，状态机确定数据序列中的同步位置，同步位置为数据序列中码字的起始位置。

步骤 803，状态机响应于确定出同步位置，进入失锁检测状态。

其中，失锁检测状态属于状态机的一个状态，表示为{失锁检测状态}。

步骤 804，状态机在失锁检测状态下，对基于同步位置选取的多个码字进行验证，响应于验证失败，更新同步位置。

采用本申请的方案，无需在发送端发送的数据序列中插入额外开销比特，即可在接收端实现数据序列的码字同步，同步可靠性比较高。而且在同步位置确定出后，还使用失锁检测机制判断同步位置是否还准确，更进一步提升同步可靠性。

针对步骤 802，确定同步位置的过程具体为：

在{同步位置确定状态}下，在数据序列中选取 N 个观察比特，基于观察比特所在的位置，确定同步位置， N 为大于或等于 1 的整数。其中，观察比特所在的位置可能为同步位置。选取观察比特的方式有多种，如下提供三种可能的实现方式。

在一种实现中，从数据序列中选取 N 个观察比特，在 N 大于 1 的情况下，多个观察比特中相邻的观察比特的间隔相同。

在另一种实现中，在码字同步过程中，引入测试块的概念，测试块属于数据序列，且测试块包括多个比特，测试块的长度与码字长度相同，可以认为每个测试块在模拟一个码字。在 N 大于 1 时， N 个观察比特可以位于同一个测试块内， N 个观察比特中每个观察比特在同一个测试块内的位置依次向后移动。例如， N 个观察比特的数目等于测试块的长度，也就是说 N 个观察比特的位置遍历单个测试块内的所有比特的位置，由于遍历单个测试块内的所有比特的位置，可以认为遍历单个码字内的所有比特的位置，所以可以使得确定出的同步位置更准确，但是随着观察比特的数目增加，码字同步过程消耗的资源也会增加。再例如， N 个观察比特的数目小于测试块长度，可以在测试块内每隔 M 个比特选取一个观察比特， M 为大于或等于 1 的整数，且 M 小于 n ， n 为码字包括的比特的数目。

在另一种实现中， N 个观察比特中每个观察比特在每个测试块内的位置依次向后移动。例如，每隔 $L*n+M$ 个比特选取一个比特作为观察比特， L 为间隔测试块的数目， L 为大于或等于 1 的整数，测试块的长度为 n ，即测试块包括 n 个比特。采用此种方式选取观察比特，相邻观察比特之间的间隔比较大，观察比特之间的相关性更低，能够减少突发误码带来的影响，进一步提升码字同步的准确性。此处观察比特的数目小于或等于 n ，在观察比特的数目等于 n 时，由于多个测试块中每个测试块的第一个比特所在的位置视为一个等同位置，第 i 个比特所在的位置视为一个等同位置，所以在观察比特的数目等于 n 时， N 个观察比特所在的位置即可遍历所有的等同位置，那么遍历 N 个观察比特即可遍历多个测试块中的所有位置， i 大于 1 且小于或等于 n 。

示例性的，下面提供了两种基于观察比特所在的位置，确定同步位置的可能实现方式。

在一种实现中，在 N 个观察比特所在的位置中，选取同步位置。

在该种实现中，一种选取同步位置的原理为：依次选取观察比特，基于每次选取的一个观察比特，连续选取多个测试块，若对于某个观察比特，多个测试块均满足验证条件，则将该观察比特所在的位置确定为同步位置，多个测试块在数据序列中的位置可以连续，也可以不连续，为方便描述在该种实现中称测试块为第一测试块。多个测试块的数目可以根据经验

值设置，也可以根据仿真分析结果设置。在本申请实施例中，测试块满足验证条件指测试块是可接受的分组码字，测试块不满足验证条件指测试块不是可接受分组码字。此处测试块是可接受的分组码字，是基于当前的验证方式认为测试块是可接受的分组码字。

具体的，图9提供了实现码字同步的示意图。如图9所示，{同步位置确定状态}包括{第一计数器重置子状态}和{第一码字验证子状态}，{第一计数器重置子状态}为对码字同步过程中所需的计数器进行重置的状态，{第一码字验证子状态}为对测试块进行验证的状态。基于N个观察比特所在的位置，选取同步位置的处理方式如下：

在{第一计数器重置子状态}下，设置[第一码字计数器]的初始值，该初始值可以为0或者其它值。在接收到的数据序列中选取一个观察比特，基于该观察比特在数据序列中选取一个第一测试块。第一测试块的起始位置与当前的观察比特之间间隔整数个码字长度。选取观察比特的方式参见前文的描述，此处不再赘述。

响应于选取出一个第一测试块，进入{第一码字验证子状态}。在{第一码字验证子状态}下，对第一测试块进行验证。响应于第一测试块满足验证条件，将[第一码字计数器]的计数值增加X，X为大于或等于1的值。

判断[第一码字计数器]的计数值与第一阈值的大小关系，响应于[第一码字计数器]的计数值等于第一阈值，确定当前的观察比特所在的位置为同步位置。

其中，在X等于1时，第一阈值为基于观察比特选取的第一测试块的数目。在X大于1时，第一阈值为基于观察比特选取的第一测试块的数目的X倍。

可选的，响应于[第一码字计数器]的计数值小于第一阈值，基于当前的观察比特选取下一个第一测试块，基于当前的观察比特从数据序列中选取下一个第一测试块，对该下一个第一测试块进行验证。此处是基于当前的观察比特选取下一个第一测试块时，是依次在数据序列中从前向后选取第一测试块，相邻的第一测试块之间的间隔可以相等，也可以不相等，相邻的第一测试块在数据序列中的位置可以相邻，也可以不相邻。

可选的，{同步位置确定状态}还包括{第一码字有效子状态}。响应于第一测试块满足验证条件，进入{第一码字有效子状态}。在{第一码字有效子状态}下，将[第一码字计数器]的计数值增加X。

可选的，响应于第一测试块满足验证条件，将[第一码字有效变量]的值置为第二值。响应于[第一码字有效变量]的值为第二值，进入{第一码字有效子状态}。

可选的，响应于第一测试块不满足验证条件，从数据序列中选取下一个观察比特，执行对该下一个观察比特的验证操作。每个观察比特的验证方式均相同。此处选取观察比特是依次选取，即连续两次选取的观察比特在数据序列中是依次向后移动。

可选的，{同步位置确定状态}还包括{第一移位子状态}，{第一移位子状态}为对观察比特进行移位的状态。响应于第一测试块不满足验证条件，进入{第一移位子状态}。在{第一移位子状态}下，移位至数据序列中的下一个观察比特，移位操作可以通过移位（SLIP）函数实现。响应于移位至该下一个观察比特，进入{第一计数器重置子状态}，执行对该下一个观察比特进行验证的操作。

可选的，在移位至下一个观察比特后，将[第一移位完成变量]的值置为真（true）或者其它值。响应于[第一移位完成变量]的值为true或者其它用于指示移位完成的值，进入{第一计数器重置子状态}。

每次在{第一计数器重置子状态}下,将[第一移位完成变量]的值置为假(false)或者其它值。

可选的,响应于第一测试块不满足验证条件,将[第一码字有效变量]的值置为第一值。响应于[第一码字有效变量]的值为第一值,进入{第一移位子状态}。

可选的,{同步位置确定状态}还包括{第一同步锁定的初始化子状态},{第一同步锁定的初始化子状态}为同步开始的状态。

响应于检测到[启动信号],进入{第一同步锁定的初始化子状态},在{第一同步锁定的初始化子状态}下,将[第一同步锁定变量]的值置为第一值,第一值可以为false,也可以为0等。

可选的,在{第一同步锁定的初始化子状态}下,将[第一同步锁定变量]的值置为第一值后,无需转换条件进入{第一计数器重置子状态}。

可选的,在{第一同步锁定的初始化子状态}下,将[第一码字待验证变量]的值置为第三值,第三值可以为false,也可以为0等。在选取出一个第一测试块后,将[第一码字待验证变量]的值置为第四值,第四值可以为true,也可以是1等。响应于[第一码字待验证变量]的值为第四值,且[第一同步锁定变量]的值置为第一值,进入{第一码字验证子状态}。在进入{第一码字验证子状态}下,将[第一码字待验证变量]置为第三值。

可选的,在{同步位置确定状态}还包括{第一同步锁定的初始化子状态}的情况下,响应于[第一码字计数器]的计数值等于第一阈值,将[第一同步锁定变量]的值置为第二值,第二值可以为true,也可以为1等。[第一同步锁定变量]的值为第二值,指示当前的观察比特所在的位置为同步位置。

可选的,在{同步位置确定状态}还包括{第一同步锁定的初始化子状态}的情况下,{同步位置确定状态}还包括{第一同步锁定的成功子状态},{第一同步锁定的成功子状态}为同步完成的状态。

响应于[第一码字计数器]的计数值等于第一阈值,进入{第一同步锁定的成功子状态}。在{第一同步锁定的成功子状态}下,将[第一同步锁定变量]的值置为第二值,第二值可以为true,也可以为1等。[第一同步锁定变量]的值为第二值,指示当前的观察比特所在的位置为同步位置。

可选的,可以基于第一测试块的特征值,判断第一测试块是否满足验证条件。第一测试块的特征值包括校验序列(syndrome)、重校验位(parity)或可纠错性。对应特征值为校验序列,验证的函数可以为计算机校验序列(CAL_SYNDROME)函数。对应特征值为重校验位,验证的函数可以为比较重校验位(COMPARE_PARITY)函数。对应特征值为可纠错性,验证的函数可以为可纠错(ERROR_CORRECTION)函数。在图9中示出的验证的函数是CAL_SYNDROME函数。

例如,在第一测试块的特征值为校验序列时,确定校验序列中的零元素的数目是否大于同步阈值,若大于同步阈值,则确定第一测试块满足验证条件,第一测试块是一个可接受的分组码字,否则确定第一测试块不满足验证条件,第一测试块不是一个可接受的分组码字。或者,确定校验序列中的非零元素的数目是否小于非同步阈值,若小于非同步阈值,则确定第一测试块满足验证条件,第一测试块是一个可接受的分组码字,否则确定第一测试块不满足验证条件,第一测试块不是一个可接受的分组码字。

在第一测试块的特征值为重校验位时,判断重校验位与原始校验位是否相同,若重校验

位与原始校验位相同，则可以确定第一测试块满足验证条件，第一测试块是一个可接受的分组码字，否则确定第一测试块不满足验证条件，第一测试块不是一个可接受的分组码字。

在第一测试块的特征值为可纠缠性，判断第一测试块是否可以被解码算法纠正，若可以被解码算法纠正，则确定第一测试块满足验证条件，第一测试块是一个可接受的分组码字，否则确定第一测试块不满足验证条件，第一测试块不是一个可接受的分组码字。

需要说明的是，上述是三种可能的验证方式均能对测试块进行验证，也可以使用上述三种可能的验证方式中两种或者三种对测试块进行验证。另外，也可以使用其它方式进行验证，本申请实施例不进行限定。

验证测试块满足验证条件，进而确定观察比特所在的位置是同步位置的原理：

对于每个线性分组码字，可以生成一个校验矩阵，该校验矩阵描述了线性分组码字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。当数据序列为线性分组码字时，该数据序列也存在校验矩阵，可以称为该数据序列的校验矩阵。具体的，数据序列与数据序列的校验矩阵之间的关系参见式（1）。

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

在式（1）中， C 为数据序列中的一个码字， S 为该码字的校验序列， H 为该数据序列的校验矩阵。由式（1）可知，数据序列中码字的校验序列与该数据序列的校验矩阵的转置矩阵相乘而得，且校验序列为一个零向量。

当该数据序列为接收端接收到的数据序列时，在数据序列中确定码字的起始位置，若没有以正确的起始位置从数据序列中选取码字，则得到的码字将不再满足式（1）。此时，数据序列中码字的校验序列可以表示为式（2）。

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

在式（2）中， R 为接收端接收到的数据序列中的一个码字， S_R 为该码字的校验序列， H 为发送端发送的原始数据序列的校验矩阵。接收端所接收的数据序列中码字的校验序列，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。从式（1）和式（2）可知，在码字包括错误比特时， S_R 不再为全零矩阵。这样，可以利用接收端所接收的数据序列中第一测试块的校验序列，判断第一测试块是否满足验证条件。

在图9所示的状态图中，{第一同步锁定的初始化子状态}表示为FEC_LOCK_INIT_1，{第一计数器重置子状态}表示为RESET_CNT_1，{第一码字验证子状态}表示为CW_CHECK_1，{第一移位子状态}表示为SLIP_1，{第一同步锁定的成功子状态}表示为FEC_LOCK_GOOD_1。[第一同步锁定变量]表示为fec_cw_sync_1，[第一码字待验证变量]表示为test_cw_1，[第一码字计数器]的计数值表示为cw_cnt_1，[第一码字有效变量]的值置为第二值表示为valid_cw_1，[第一码字有效变量]的值置为第一值表示为!valid_cw_1，[第一移位完成变量]表示为slip_done_1。

另一种选取同步位置的原理为：遍历 N 个观察比特，在 N 个观察比特所在的位置中确定同步位置。例如，遍历 N 个观察比特，基于每个观察比特连续选取第四阈值个测试块，确定第四阈值个测试块中满足验证条件的测试块的数目，第四阈值个测试块在数据序列中的位置可以连续，也可以不连续。将满足验证条件的测试块的数目最大的观察比特所在的位置，确定为同步位置，或者，将不满足验证条件的测试块的数目最小的观察比特所在的位置，确定为同步位置。此处，第四阈值为后文中 W 等于1时的取值。第四阈值可以根据经验值设置，

或者根据仿真分析结果设置。

具体的，确定同步位置的状态图参见后文中图 11 的描述，相当于图 11 输出的信号为第二转换信号，第二转换信号用于触发进入{失锁检测状态}。

再一种选取同步位置的原理为：从第一个观察比特开始，依次验证每个观察比特是否满足同步条件，同步条件可以为：基于观察比特连续选取的第五阈值个测试块中，至少存在第六阈值个测试块满足验证条件。第五阈值个测试块在数据序列中的位置可以连续，也可以不连续，第六阈值个测试块在数据序列中的位置可以连续，也可以不连续。若某个观察比特满足同步条件，则确定该观察比特所在的位置为同步位置。这样，有可能提前结束观察比特的验证，节约同步位置确定的资源。

此处，第五阈值为后文中 M 等于 1 时的取值，第六阈值为后文中 Q 等于 1 时的取值。第五阈值和第六阈值可以根据经验值设置，或者根据仿真分析结果设置。

具体的，确定同步位置的状态图参见后文中图 12 的描述，相当于图 12 输出的信号为第二转换信号，第二转换信号用于触发进入{失锁检测状态}。

示例性的，在确定出同步位置后，进入{失锁检测状态}，图 10 提供了由一个状态机实现码字同步过程的状态图。在图 10 所示的状态图中，同步位置确定过程和失锁检测过程共用{第一计数器重置子状态}和{第一移位子状态}。

具体的，参见图 10，{失锁检测状态}包括{第二码字验证子状态}，{第二码字验证子状态}为进行码字验证的状态。响应于确定出同步位置，进入{第一计数器重置子状态}，或者，[第一同步锁定变量]的值为第二值时，无需转换条件进入{第一计数器重置子状态}。在{第一计数器重置子状态}下，设置[第一码字计数器]的初始值和[第一无效码字计数器]的初始值，两个初始值均可以为 0。基于同步位置在数据序列中选取一个码字。此处，为了方便描述，基于同步位置选取的码字称为第一码字。

响应于选取出一个第一码字，进入{第二码字验证子状态}。在{第二码字验证子状态}下，对该第一码字进行验证。

响应于第一码字不满足验证条件，将[第一码字计数器]的计数值增加 Y，将[第一无效码字计数器]的计数值增加 Z，Y 和 Z 均为大于或等于 1 的整数。[第一码字计数器]的计数值小于或等于第二阈值的情况下，响应于[第一无效码字计数器]的计数值等于第三阈值，说明无效码字的数目已经达到更新同步位置的数目，重新进入{同步位置确定状态}，执行更新同步位置的操作。其中，在 Y 等于 1 时，第二阈值为基于同步位置选取的第一码字的数目。在 Y 大于 1 时，第二阈值为基于同步位置选取的第一码字的数目的 Y 倍。在 Z 等于 1 时，第三阈值为不满足验证条件的第一码字的数目。在 Z 大于 1 时，第三阈值为不满足验证条件的第一码字的数目的 Z 倍。第二阈值和第三阈值可以根据经验值设置，或者根据仿真分析结果设置。

可选的，可以基于第一码字的特征值，判断第一码字是否满足验证条件，原理参见式 (1) 和式 (2)。

可选的，{失锁检测状态}还包括{码字无效子状态}，{码字无效子状态}为第一码字不满足验证条件时进入的状态。响应于第一码字不满足验证条件，进入{码字无效子状态}。在{码字无效子状态}下，将[第一码字计数器]的计数值增加 Y，将[第一无效码字计数器]的计数值增加 Z。

可选的，响应于[第一码字计数器]的计数值小于第二阈值，且[第一无效码字计数器]的计

数值小于第三阈值, 重新进入{第二码字验证子状态}。基于该同步位置选取下一个第一码字, 对该下一个第一码字进行验证。此处[第一无效码字计数器]的计数值小于第三阈值说明无效码字的数目还比较少, 可以对下一个第一码字进行验证。

可选的, 响应于[第一码字计数器]的计数值等于第二阈值, 且[第一无效码字计数器]的计数值小于第三阈值, 重新进入{第一计数器重置子状态}, 继续对同步位置进行下一次验证。此处第二阈值个第一码字中存在的无效码字数目比较少, 说明同步位置当前还准确, 可以继续验证。

可选的, 响应于第一码字不满足验证条件, 将[码字有效变量]的值置为第一值。响应于[码字有效变量]的值为第一值, 进入{码字无效子状态}。

可选的, 响应于第一码字满足验证条件, 将[第一码字计数器]的计数值增加 Y。响应于[第一码字计数器]的计数值等于第二阈值, 重新进入{第一计数器重置子状态}, 说明同步位置准确, 继续对同步位置进行下一次验证。

响应于[第一码字计数器]的计数值小于第二阈值, 重新进入{第二码字验证子状态}。基于该同步位置选取下一个第一码字, 对该下一个第一码字进行验证。

可选的, {失锁检测状态}还包括{码字有效子状态}, {码字有效子状态}为第一码字满足验证条件时进入的状态。响应于第一码字满足验证条件, 进入{码字有效子状态}。

可选的, 响应于第一码字满足验证条件, 将[码字有效变量]的值置为第二值。响应于[码字有效变量]的值为第二值, 进入{码字有效子状态}。

可选的, 在{同步位置确定状态}还包括{第一移位子状态}的情况下, 重新进入{同步位置确定状态}的过程为: 响应于[第一无效码字计数器]的计数值等于第三阈值, 进入{第一移位子状态}。在{第一移位子状态}下, 移位至数据序列中的下一个观察比特。响应于移位至该下一个观察比特, 进入{第一计数器重置子状态}, 执行对下一个观察比特进行验证的操作。

可选的, 在移位至下一个观察比特后, 将[第一移位完成变量]的值置为第二值。响应于[第一移位完成变量]的值为第二值, 重新进入{第一计数器重置子状态}, 在{第一计数器重置子状态}下, 将[第一移位完成变量]的值置为第一值, 执行对下一个观察比特进行验证的操作。

可选的, 在选取出一个第一码字后, 将[第一码字待验证变量]的值置为第四值。响应于[第一码字待验证变量]的值为第四值, 且[第一同步锁定变量]的值为第二值, 进入{第二码字验证子状态}。在进入{第二码字验证子状态}后, 将[第一码字待验证变量]置为第三值。

在图 10 所示的状态图中, {第一同步锁定的初始化子状态}表示为 FEC_LOCK_INIT_1, {第一计数器重置子状态}表示为 RESET_CNT_1, {第二码字验证子状态}表示为 CW_CHECK_2, {码字有效子状态}表示为 VALID_CW_0, {第一移位子状态}表示为 SLIP_1, {第一同步锁定的成功子状态}表示为 FEC_LOCK_GOOD_1, {码字无效子状态}表示为 INVALID_CW_0。[第一同步锁定变量]表示为 fec_cw_sync_1, [第一码字待验证变量]表示为 test_cw_1, [第一码字计数器]的计数值表示为 cw_cnt_1, [码字有效变量]的值置为第二值表示为 valid_cw_0, [码字有效变量]的值置为第一值表示为 ! valid_cw_0, [第一无效码字计数器]的计数值表示为 invalid_cw_cnt_1, [第一移位完成变量]表示为 slip_done_1。

在图 10 所示的状态图中, 基于观察比特选取的第一测试块属于第一子序列, 基于同步位置选取的第一码字属于第二子序列, 第一子序列和第二子序列均属于数据序列, 且第二子序列位于第一子序列之后。

在另一种实现中，在{同步位置确定状态}下，在数据序列中选取N个观察比特，N大于或等于1，在N个观察比特所在的位置中，选取备选位置。基于备选位置确定同步位置。

其中，备选位置为最有可能为同步位置的位置。可以将备选位置确定为同步位置。或者，对基于备选位置选取的多个测试块进行验证，在验证通过后，将备选位置确定为同步位置。

示例性的，确定备选位置的方式有多种，如下给出三种可能的实现方式，参见方式一至方式三。

方式一，遍历N个观察比特，在N个观察比特所在的位置中确定备选位置。例如，遍历N个观察比特，基于每个观察比特连续选取第四阈值个测试块，确定第四阈值个测试块中满足验证条件的测试块的数目，第四阈值个测试块在数据序列中的位置可以连续，也可以不连续。将满足验证条件的测试块的数目最大的观察比特所在的位置，确定为备选位置，或者，将不满足验证条件的测试块的数目最小的观察比特所在的位置，确定为备选位置。这样，由于遍历N个观察比特，所以能够使得确定出的备选位置最有可能是同步位置。在此种实现中，N的取值可以是单个码字中包括的比特的数目。

具体的，参见图11所示的状态图，{同步位置确定状态}包括{第二计数器重置子状态}、{第三码字验证子状态}和{备选位置选择子状态}，{第二计数器重置子状态}为对计数器进行重置的状态，{第三码字验证子状态}为对测试块进行验证的状态，{备选位置选择子状态}为选取备选位置的状态。

在{第二计数器重置子状态}下，设置[第二码字计数器]的初始值和[第一当前有效码字计数器]的初始值，从接收到的数据序列中选取一个观察比特，基于当前的观察比特，从数据序列中选取一个测试块，为了方便描述，选取的测试块称为第二测试块，第二测试块的长度等于码字的长度，一个第二测试块在模拟一个码字。

响应于选取第二测试块，进入{第三码字验证子状态}。

在{第三码字验证子状态}下，对第二测试块进行验证，验证方式参见前文中第一测试块的验证方式，此处不再赘述。在{第三码字验证子状态}下，将[第二码字计数器]的计数值增加W。响应于第二测试块满足验证条件，将[第一当前有效码字计数器]的计数值增加P，W和P均为大于或等于1的整数。

响应于[第二码字计数器]的计数值等于第四阈值，进入{备选位置选择子状态}。此处[第二码字计数器]的计数值等于第四阈值，在W等于1的情况下，说明已经对当前的观察比特对应的第四阈值个第二测试块进行验证，可以确定下当前的观察比特所在的位置是否是备选位置，在W大于1的情况下，说明已经对当前的观察比特对应的多个第二测试块进行验证，可以确定下当前的观察比特所在的位置是否是备选位置。

在{备选位置选择子状态}下，将[观察比特数目变量]的值增加1。

在{备选位置选择子状态}下，响应于[第一当前有效码字计数器]的计数值大于[最大有效码字数目变量]的当前值，更新[备选同步位置变量]的值为当前的观察比特所在的位置，将[最大有效码字数目变量]的值更新为[第一当前有效码字计数器]的计数值。在[第一当前有效码字计数器]的计数值小于[最大有效码字数目变量]的当前值时，不更新[备选同步位置变量]，在[第一当前有效码字计数器]的计数值等于[最大有效码字数目变量]的当前值时，可以更新[备选同步位置变量]，也可以不更新[备选同步位置变量]。

响应于[观察比特数目变量]的值小于N，执行对数据序列中位于当前的观察比特之后的下

一个观察比特的验证操作。此处[观察比特数目变量]的值小于 N，说明 N 个观察比特还未全部验证完，应该选取下一个观察比特，对该下一个观察比特进行验证。

响应于[观察比特数目变量]的值等于 N，将[备选同步位置变量]的值确定为备选位置。可选的，在确定出备选位置后可以输出第一转换信号，第一转换信号可以是备选位置所在的位置，也可以是触发进入下一个状态的信号。

响应于第二测试块不满足验证条件，判断[第二码字计数器]的计数值与第四阈值的大小关系。响应于[第二码字计数器]的计数值小于第四阈值，基于当前的观察比特选取下一个第二测试块，对该下一个第二测试块进行验证。响应于[第二码字计数器]的计数值等于第四阈值，进入{备选位置选择子状态}。

可选的，{同步位置确定状态}还包括{第二同步锁定的初始化子状态}。

在进入{第二计数器重置子状态}之前，响应于检测到[启动信号]，进入{第二同步锁定的初始化子状态}。在{第二同步锁定的初始化子状态}下，将[观察比特数目变量]的值置为第五值，第五值可以为 0，将[备选同步位置变量]的值置为第六值，第六值可以为 0，将[最大有效码字数目变量]的值置为第七值，第七值可以为 0。

可选的，在设置这些变量后，无需转换条件进入{第二计数器重置子状态}。

可选的，在{第二同步锁定的初始化子状态}下，将[第二同步锁定变量]的值置为第一值。

可选的，在{第二同步锁定的初始化子状态}下，将[当前位置数目变量]的值置为 0 等。在{备选位置选择子状态}下，将[当前位置数目变量]的值增加 1。这样，记录当前位置变量，能够得到所遍历的当前观察比特所在的位置。

可选的，在{第二同步锁定的初始化子状态}下，将[第二码字待验证变量]的值置为第三值。

可选的，响应于基于当前的观察比特从数据序列中选取一个第二测试块，将[第二码字待验证变量]的值置为第四值，第四值可以是 true 或者其它值。响应于[第二码字待验证变量]的值为第四值，进入{第三码字验证子状态}。在{第三码字验证子状态}下，将[第二码字待验证变量]的值置为第三值，使得对第二测试块进行验证。

可选的，响应于[第二码字计数器]的计数值小于第四阈值，且[第二码字待验证变量]的值为第四值，重新进入{第三码字验证子状态}，基于当前的观察比特选取下一个第二测试块，对该下一个第二测试块进行验证。

可选的，{同步位置确定状态}还包括{第二移位子状态}，{第二移位子状态}为对观察比特进行移位的状态。响应于[观察比特数目变量]的值小于 N，说明还未遍历完 N 个观察比特，进入{第二移位子状态}。在{第二移位子状态}下，移位至数据序列中下一个观察比特所在的位置，响应于移位至下一个观察比特，进入{第二计数器重置子状态}，执行对下一个观察比特的验证操作。这样，按照此种方式能够遍历所有的观察比特，从中选取出最有可能的备选位置，能够使得选取的备选位置更准确。

其中，在移位至下一个观察比特后，将[第二移位完成变量]的值置为第二值，第二值为 true 或者其它值。响应于[第二移位完成变量]的值为第二值，进入{第二计数器重置子状态}，执行对下一个观察比特进行验证的操作，在{第二计数器重置子状态}下，将[第二移位完成变量]的值置为第一值，第一值为 false 或者其它值。

此处，执行每个观察比特的验证操作的处理相同。选取观察比特的方式参见前文中的描述，此处不再赘述。

在图 11 所示的状态图中，{第二同步锁定的初始化子状态}表示为 FEC_LOCK_INIT_2，{第二计数器重置子状态}表示为 RESET_CNT_2，{第三码字验证子状态}表示为 CW_CHECK_3，{备选位置选择子状态}表示为 SELECT_SYNC_LOCK，{第二移位子状态}表示为 SLIP_2。[第二同步锁定变量]表示为 fec_cw_sync_2，[第二码字待验证变量]表示为 test_cw_2，[第二码字计数器]的计数值表示为 cw_cnt_2，[第二移位完成变量]表示为 slip_done_2，[观察比特数目变量]表示为 location_num，[备选同步位置变量]表示为 sync_location，[最大有效码字数目变量]表示为 max_valid_cw_cnt，[当前位置数目变量]表示为 current_location，[第一当前有效码字计数器]的计数值表示为 current_valid_cw_cnt。码字有效表示为 valid_cw，“if”表示若或如果。

方式二，从第一个观察比特开始，依次验证每个观察比特是否满足同步条件，同步条件可以为：基于观察比特连续选取的多个测试块中，至少存在一定数目个测试块满足验证条件。多个测试块在数据序列中的位置可以连续，也可以不连续，一定数目个测试块在数据序列中的位置可以连续，也可以不连续。若某个观察比特满足同步条件，则确定该观察比特所在的位置为备选位置。这样，有可能提前结束观察比特的验证，节约备选位置确定的资源。

具体的，参见图 12 所示的状态图，{同步位置确定状态}包括{第三计数器重置子状态}、{第四码字验证子状态}和{第二码字有效子状态}。{第三计数器重置子状态}为对计数器进行重置的状态，{第四码字验证子状态}为对测试块进行验证的状态，{第二码字有效子状态}为测试块满足验证条件时进入的状态。

在{第三计数器重置子状态}下，设置[第三码字计数器]的初始值和[第一有效码字计数器]的初始值。在数据序列中选取一个观察比特，基于当前的观察比特，从数据序列中选取一个测试块，为了方便描述，选取的测试块称为第三测试块。第三测试块的长度等于码字长度，第三测试块的起始位置与观察比特所在的位置之间间隔整数个码字长度。

响应于选取出一个第三测试块，进入{第四码字验证子状态}。在{第四码字验证子状态}下，对第三测试块进行验证。验证方式参见对第一测试块验证的方式。

响应于第三测试块满足验证条件，进入{第二码字有效子状态}。在{第二码字有效子状态}下，将[第三码字计数器]的计数值增加 M，将[第一有效码字计数器]的计数值增加 Q，M 和 Q 均为大于或等于 1 的整数。

在[第三码字计数器]的计数值小于或等于第五阈值的情况下，响应于[第一有效码字计数器]的计数值等于第六阈值，将当前的观察比特所在的位置确定为备选位置。可选的，在确定出备选位置后可以输出第一转换信号，第一转换信号可以是备选位置所在的位置，也可以是触发进入下一个状态的信号。其中，在 M 等于 1 时，第五阈值为基于观察比特选取的第三测试块的数目。在 M 大于 1 时，第五阈值为基于观察比特选取的第三测试块的数目的 M 倍。在 Q 等于 1 时，第六阈值为满足验证条件的第三测试块的数目。在 Q 大于 1 时，第六阈值为满足验证条件的第三测试块的数目的 Q 倍。第五阈值和第六阈值可以根据经验值设置，或者根据仿真分析结果设置。

可选的，在{第二码字有效子状态}下，响应于[第三码字计数器]的计数值等于第五阈值，且[第一有效码字计数器]的计数值小于第六阈值，说明第五阈值个第三测试块不存在第六阈值个第三测试块满足验证条件，执行从数据序列中选取下一个观察比特进行验证的操作。

在{第二码字有效子状态}下，响应于[第三码字计数器]的计数值小于第五阈值，且[第一

有效码字计数器]的计数值小于第六阈值,说明第五阈值个第三测试块还未全部验证完毕。重新进入{第四码字验证子状态},对基于当前的观察比特选取的下一个第三测试块进行验证。

可选的,在响应于第三测试块满足验证条件时,将[第二码字有效变量]的值置第二值。响应于[第二码字有效变量]的值为第二值,进入{第二码字有效子状态}。

可选的,{同步位置确定状态}还包括{第一码字无效子状态},{第一码字无效子状态}为测试块不满足验证条件时进入的状态。响应于第三测试块不满足验证条件,进入{第一码字无效子状态}。在{第一码字无效子状态}下,将[第三码字计数器]的计数值增加M。

响应于[第三码字计数器]的计数值等于第五阈值,执行从数据序列中选取下一个观察比特进行验证的操作。响应于[第三码字计数器]的计数值小于第五阈值,说明第五阈值个第三测试块还未全部验证完毕。重新进入{第四码字验证子状态},对基于当前的观察比特选取的下一个第三测试块进行验证。

可选的,在响应于第三测试块不满足验证条件时,将[第二码字有效变量]的值置为第一值。响应于[第二码字有效变量]的值为第一值,进入{第一码字无效子状态}。

可选的,{同步位置确定状态}还包括{第三移位子状态},{第三移位子状态}为移位至下一个观察比特的状态。在{第一码字无效子状态}下,响应于[第三码字计数器]的计数值等于第五阈值,进入{第三移位子状态}。或者,在{第二码字有效子状态}下,响应于[第三码字计数器]的计数值等于第五阈值,且[第一有效码字计数器]的计数值小于第六阈值,进入{第三移位子状态}。在{第三移位子状态}下,将移位至下一个观察比特。响应于移位至下一个观察比特,重新进入{第三计数器重置子状态},执行对该下一个观察比特进行验证的操作。

其中,在移位至下一个观察比特后,将[第三移位完成变量]的值置为第二值,第二值为true 或者其它值。响应于[第三移位完成变量]的值为第二值,进入{第三计数器重置子状态},执行对下一个观察比特进行验证的操作,在{第三计数器重置子状态}下,将[第三移位完成变量]的值置为第一值,第一值为false 或者其它值。

可选的,{同步位置确定状态}还包括{第三同步锁定的初始化子状态},{第三同步锁定的初始化子状态}为初始化的状态。在进入{第三计数器重置子状态}之前,响应于检测到启动信号,进入{第三同步锁定的初始化子状态}。

在{第三同步锁定的初始化子状态}下,将[第三同步锁定变量]的值置为第一值,并且将[第三码字待验证变量]的值置为第三值,第三值为false 或其它值。在做完这些处理后无需转换条件进入{第三计数器重置子状态}。在基于当前的观察比特选取出一个第三测试块后,将[第三码字待验证变量]的值置为第四值,第四值为true 或其它值。

在{第三计数器重置子状态}下,响应于[第三码字待验证变量]的值为第四值,进入{第四码字验证子状态}。

在{第二码字有效子状态}下,响应于[第三码字待验证变量]的值为第四值,且[第三码字计数器]的计数值小于第五阈值,且[第一有效码字计数器]的计数值小于第六阈值,重新进入{第四码字验证子状态},对基于当前的观察比特选取的下一个第三测试块进行验证。

在{第一码字无效子状态}下,响应于[第三码字待验证变量]的值为第四值,且[第三码字计数器]的计数值小于第五阈值,重新进入{第四码字验证子状态},对基于当前的观察比特选取的下一个第三测试块进行验证。

在图12所示的流程中,执行每个观察比特的验证操作的处理相同。选取观察比特的方式

参见前文中的描述，此处不再赘述。

方式三，从第一个观察比特开始，依次验证每个观察比特是否满足同步条件，同步条件可以为基于观察比特连续选取的多个测试块满足验证条件，该多个测试块在数据序列中的位置可以连续，也可以不连续。若某个观察比特满足同步条件，则确定该观察比特所在的位置为备选位置。这样，有可能可以提前结束观察比特的验证，节约备选位置确定的资源。该种实现是方式二的特例。

具体的，参见图 13 所示的状态图，{同步位置确定状态}包括{第三计数器重置子状态}、{第四码字验证子状态}和{第二码字有效子状态}。

在{第三计数器重置子状态}下，设置第三码字计数器的初始值。在数据序列中选取一个观察比特，基于当前的观察比特，从数据序列中选取一个测试块，为了方便描述，选取的测试块被称为第三测试块。

响应于选取一个第三测试块，进入{第四码字验证子状态}。在{第四码字验证子状态}下，对第三测试块进行验证。例如，基于第三测试块的特征值对第三测试块进行验证。

响应于第三测试块满足验证条件，进入{第二码字有效子状态}。在{第二码字有效子状态}下，将[第三码字计数器]的计数值增加 M。

响应于[第三码字计数器]的计数值等于第五阈值，将当前的观察比特所在的位置确定为备选位置。可选的，在确定出备选位置后可以输出第一转换信号，第一转换信号可以是备选位置所在的位置，也可以是触发进入下一个状态的信号。

响应于[第三码字计数器]的计数值小于第五阈值，对基于当前的观察比特选取的下一个第三测试块进行验证。

响应于第三测试块不满足验证条件，进入{第三移位子状态}。在{第三移位子状态}下，将移位至下一个观察比特。响应于移位至下一个观察比特，进入{第三计数器重置子状态}，执行对该下一个观察比特进行验证的操作。

可选的，在响应于第三测试块满足验证条件时，将[第二码字有效变量]的值置为第二值，响应于[第二码字有效变量]的值为第二值，进入{第二码字有效子状态}。具体的移位描述参见针对图 12 的描述中。

可选的，{同步位置确定状态}还包括{第三同步锁定的初始化子状态}，描述参见前文中的描述，此处不再赘述。

在图 12 和图 13 所示的状态图中，{第三计数器重置子状态}表示为 RESET_CNT_3，{第四码字验证子状态}表示为 CW_CHECK_4，{第三移位子状态}表示为 SLIP_3，{第三同步锁定的初始化子状态}表示为 FEC_LOCK_INIT_3，{第一码字无效子状态}表示为 INVALID_CW_1，{第二码字有效子状态}表示为 VALID_CW_2。[第三同步锁定变量]表示为 fec_cw_sync_3，[第三码字待验证变量]表示为 test_cw_3，[第三码字计数器]的计数值表示为 cw_cnt_3，[第三移位完成变量]表示为 slip_done_3，[第一有效码字计数器]的计数值表示为 valid_cw_cnt_1，[第二码字有效变量]的值置为第二值表示为 valid_cw_2，[第二码字有效变量]的值置为第一值表示为 !valid_cw_2。

示例性的，在选取出备选位置后，基于备选位置确定同步位置。

在第一种基于备选位置确定同步位置的实现中，将备选位置确定为同步位置。具体的，{同步位置确定状态}还包括{第三同步锁定的成功子状态}，在基于上述方式一确定备选位置时，

响应于[观察比特数目变量]的值等于N,进入{第三同步锁定的成功子状态}。在基于上述方式二确定备选位置时,响应于[第二有效码字计数器]的计数值的值等于第六阈值,进入{第三同步锁定的成功子状态}。在基于上述方式三确定备选位置时,响应于[第三码字计数器]的计数值等于第五阈值,进入{第三同步锁定的成功子状态}。在{第三同步锁定的成功子状态}下,将[同步锁定变量]的值置为第二值。在该第一种基于备选位置确定同步位置的实现中,直接将备选位置确定为同步位置,能够节约同步位置的确定资源。

在第二种基于备选位置确定同步位置的实现中,对备选位置进行验证,在验证成功后,确定备选位置为同步位置。

具体的,在一种实现中,参见图14所示的状态图,基于备选位置连续选取多个测试块,为了方便描述,在该实现中选取的测试块称为第四测试块。该多个测试块均满足验证条件,确定备选位置为同步位置,该多个测试块在数据序列中的位置可以连续,也可以不连续。

{同步位置确定状态}还包括{第四计数器重置子状态}和{第五码字验证子状态},{第四计数器重置子状态}为对计数器进行重置的状态,{第五码字验证子状态}是对测试块进行验证的状态。

在{第四计数器重置子状态}下,设置[第四码字计数器]的初始值,对于备选位置,[第四码字计数器]用于记录基于该备选位置选取的测试块的数目。基于备选位置从数据序列中选取第四测试块。响应于选取出一个第四测试块,进入{第五码字验证子状态}。

在{第五码字验证子状态}下,对第四测试块进行验证,验证方式参见前文中对第一测试块验证的方式,此处不再赘述。

响应于第四测试块满足验证条件,将[第四码字计数器]的计数值增加R,R为大于或等于1的整数。响应于[第四码字计数器]的计数值等于第七阈值,确定该备选位置为同步位置。

其中,在R等于1时,第七阈值为基于备选位置选取的第四测试块的数目。在R大于1时,第七阈值为基于备选位置选取的第四测试块的数目的R倍。第七阈值可以根据经验值设置,也可以根据仿真分析结果设置。

可选的,同步位置确定状态还包括{第三码字有效子状态},{第三码字有效子状态}为测试块满足验证条件时进行的状态。响应于第四测试块满足验证条件,进入{第三码字有效子状态}。在{第三码字有效子状态}下,将[第四码字计数器]的计数值增加R。响应于[第四码字计数器]的计数值等于第七阈值,确定该备选位置为同步位置。响应于[第四码字计数器]的计数值小于第七阈值,重新进入{第五码字验证子状态},对基于备选位置选取的下一个第四测试块进行验证。

可选的,响应于第四测试块不满足验证条件,执行重新选取备选位置的操作。

可选的,响应于第四测试块满足验证条件,将[第三码字有效变量]的值置为第二值,第二值为true或其它值。响应于[第三码字有效变量]的值为第二值,进入{第三码字有效子状态}。

响应于第四测试不满足验证条件,将[第三码字有效变量]的值置为第一值,第一值为false或其它值。响应于[第三码字有效变量]的值为第一值,执行重新选取备选位置的操作。

可选的,{同步位置确定状态}还包括{第一重新开始子状态}。响应于[第三码字有效变量]的值为第一值,进入{第一重新开始子状态}。在{第一重新开始子状态}下,将[重新同步锁定变量]的值置为第二值。响应于[重新同步锁定变量]的值为第二值,确定检测到启动信号,执行重新选取备选位置的操作。

可选的, {同步位置确定状态}还包括{第三同步锁定的成功子状态}, {第三同步锁定的成功子状态}为确定出同步位置的状态。响应于[第四码字计数器]的计数值等于第七阈值, 进入{第三同步锁定的成功子状态}。在{第三同步锁定的成功子状态}下, 将[第三同步锁定变量]的值置为第二值。响应于[第三同步锁定变量]的值为第二值, [第三同步锁定变量]的值为第二值用于指示备选位置为同步位置, 可以无需转换条件输出第二转换信号。该第二转换信号用于触发进入{失锁检测状态}, 第二转换信号可以是同步位置或者其它触发信号。

可选的, 响应于基于同步位置选取一个第四测试块后, 将[第四码字待验证变量]的值置为第二值, 响应于[第四码字待验证变量]的值为第二值, 进入{第五码字验证子状态}。或者, 响应于[第四码字待验证变量]的值为第二值, 且[第四码字计数器]的计数值小于第七阈值, 进入{第五码字验证子状态}。在{第五码字验证子状态}下, 将[第四码字待验证变量]的值置为第一值。

这样, 由于在确定出备选位置后, 还继续进行确认备选位置是否为同步位置, 进一步提升同步位置的准确性。

在图 14 所示的状态图中, {第三同步锁定的成功子状态}表示为 FEC_LOCK_GOOD_3, {第四计数器重置子状态}表示为 RESET_CNT_4, {第五码字验证子状态}表示为 CW_CHECK_5, {第三码字有效子状态}表示为 VALID_CW_3, {第一重新开始子状态}表示为 RESTART_1。[第三同步锁定变量]表示为 fec_cw_sync_3, [第四码字待验证变量]表示为 test_cw_4, [第四码字计数器]的计数值表示为 cw_cnt_4, [重新同步锁定变量]表示为 restart_fec_lock, [第三码字有效变量]的值为第一值表示为 !valid_cw_3, [第三码字有效变量]的值为第二值表示为 valid_cw_3。

在另一种实现中, 基于备选位置连续选取多个测试块, 为了方便描述, 在该实现中, 选取的测试块称为第五测试块。该多个测试块中有一定数目个测试块满足验证条件, 确定备选位置为同步位置, 该多个测试块在数据序列中的位置可以连续, 也可以不连续, 该一定数目个测试块在数据序列中的位置可以连续, 也可以不连续。

具体的, 参见图 15 所示的状态图, {同步位置确定状态}还包括{第六计数器重置子状态}、{第七码字验证子状态}和{第五码字有效子状态}, {第六计数器重置子状态}为对计数器进行重置的状态, {第七码字验证子状态}为对测试块进行验证的状态, {第五码字有效子状态}为测试块满足验证条件时进行的状态。

在{第六计数器重置子状态}下, 设置[第六码字计数器]的初始值和[第四有效码字计数器]的初始值。对于备选位置, [第六码字计数器]用于记录基于该备选位置选取的测试块的数目, [第四有效码字计数器]用于记录测试块满足验证条件的数目。基于备选位置从数据序列中选取一个第五测试块。响应于选取出一个第五测试块, 进入{第七码字验证子状态}。

可选的, 在{第六计数器重置子状态}下, 还可以将[第四移位完成变量]的值置为第一值, 用于指示当前不需要进行移位。

在{第七码字验证子状态}下, 对第五测试块进行验证, 验证方式参见前文中对第一测试块验证的方式, 此处不再赘述。

响应于第五测试块满足验证条件, 进入{第五码字有效子状态}。在{第五码字有效子状态}下, 将[第六码字计数器]的计数值增加 E, E 为大于或等于 1 的整数, 将[第四有效码字计数器]的计数值增加 F, F 为大于或等于 1 的整数。在[第六码字计数器]的计数值小于或等于第九

阈值的情况下, 响应于[第四有效码字计数器]的计数值等于第十阈值, 确定该备选位置为同步位置。响应于[第六码字计数器]的计数值等于第九阈值, 且[第四有效码字计数器]的计数值小于第十阈值, 执行重新选取备选位置的操作。响应于[第六码字计数器]的计数值小于第九阈值, 且[第四有效码字计数器]的计数值小于第十阈值, 重新进入{第七码字验证子状态}, 继续对基于备选位置选取的下一个第五测试块进行验证。其中, 在E等于1时, 第九阈值为基于备选位置选取的第五测试块的数目。在E大于1时, 第九阈值为基于备选位置选取的第五测试块的数目的E倍。在F等于1时, 第十阈值为满足验证条件的第五测试块的数目。在F大于1时, 第十阈值为满足验证条件的第五测试块的数目的F倍。第九阈值和第十阈值可以根据经验值设置, 也可以根据仿真分析结果设置。

可选的, {同步位置确定状态}还包括{第三码字无效子状态}, {第三码字无效子状态}为测试块不满足验证条件时进入的状态。响应于第五测试块不满足验证条件, 进入{第三码字无效子状态}。在{第三码字无效子状态}下, 将[第六码字计数器]的计数值增加E。响应于[第六码字计数器]的计数值等于第九阈值, 执行重新确定备选位置的操作。响应于[第六码字计数器]的计数值小于第九阈值, 重新进入{第七码字验证子状态}, 继续对基于备选位置选取的下一个第五测试块进行验证。

可选的, 响应于第五测试块满足验证条件, 将[第四码字有效变量]的值置为第二值。响应于[第四码字有效变量]的值为第二值, 进入{第五码字有效子状态}。

响应于第五测试块不满足验证条件, 将[第四码字有效变量]的值置为第一值。响应于[第四码字有效变量]的值为第一值, 进入{第三码字无效子状态}。

可选的, {同步位置确定状态}还包括{第二重新开始子状态}, {第二重新开始子状态}为输出重新确定备选位置的信号的状态。在{第三码字无效子状态}下, 响应于[第六码字计数器]的计数值等于第九阈值, 进入{第二重新开始子状态}。或者, 在{第五码字有效子状态}下, 响应于[第六码字计数器]的计数值等于第九阈值, 且[第四有效码字计数器]的计数值小于第十阈值, 进入{第二重新开始子状态}。在{第二重新开始子状态}下, 将[重新同步锁定变量]的值置为第二值。响应于[重新同步锁定变量]的值置为第二值, 确定检测到启动信号, 执行重新选取备选位置的操作。

可选的, {同步位置确定状态}还包括{第四同步锁定的成功子状态}, {第四同步锁定的成功子状态}为同步完成的状态。响应于[第四有效码字计数器]的计数值等于第十阈值, 进入{第四同步锁定的成功子状态}。在{第四同步锁定的成功子状态}下, 将[第四同步锁定变量]的值置为第二值。[第四同步锁定变量]的值为第二值用于指示备选位置为同步位置, 可以无需转换条件输出第二转换信号。该第二转换信号用于触发进入{失锁检测状态}。

可选的, 响应于基于备选位置选取一个第五测试块后, 将[第五码字待验证变量]的值置为第二值, 响应于[第五码字待验证变量]的值为第二值, 进入{第七码字验证子状态}。或者, 响应于[第五码字待验证变量]的值为第二值, 且[第六码字计数器]的计数值小于第九阈值, 重新进入{第七码字验证子状态}。或者, 响应于[第五码字待验证变量]的值为第二值, 且[第六码字计数器]的计数值小于第九阈值, 且[第四有效码字计数器]的计数值小于第十阈值, 重新进入{第七码字验证子状态}。在{第七码字验证子状态}下, 将[第五码字待验证变量]的值置为第一值。

在图 15 所示的状态图中, {第六计数器重置子状态}表示为 RESET_CNT_6, {第七码字

验证子状态}表示为 CW_CHECK_7, {第四同步锁定的成功子状态}表示为 FEC_LOCK_GOOD_4, {第五码字有效子状态}表示为 VALID_CW_5, {第三码字无效子状态}表示为 INVALID_CW_3, {第二重新开始子状态}表示为 RESTART_2。[第四同步锁定变量]表示为 fec_cw_sync_4, [第五码字待验证变量]表示为 test_cw_5, [第六码字计数器]的计数值表示为 cw_cnt_6, [第四有效码字计数器]的计数值表示为 valid_cw_cnt_4, [重新同步锁定变量]表示为 restart_fec_lock, [第四码字有效变量]的值是第一值对应! valid_cw_4, [第四码字有效变量]的值是第二值对应 valid_cw_4。

在图 9 和图 11 至图 13 所示的状态图中, 基于观察比特选取的测试块属于第一子序列。在图 14 与图 15 所示的状态图中, 基于备选位置选取的测试块属于第三子序列, 第一子序列和第三子序列均属于接收序列, 且第三子序列位于第一子序列之后。另外, 第一子序列与第三子序列也可以存在部分重叠。

在另一种实现中, 可以基于备选位置连续选取出第九阈值个第五测试块, 遍历第九阈值个第五测试块, 在第九阈值个第五测试块中存在第十阈值个第五测试块满足验证条件, 确定备选位置为同步位置。

示例性的, 在步骤 804 中, 进入{失锁检测状态}, 进行失锁检测。有多种方式能够进行失锁检测, 如下提供三种可行的方式。

方式一, 在{失锁检测状态}下, 对基于同步位置选取多个码字进行验证, 响应于该多个码字中不满足验证条件的码字数目达到一定数值, 重新进入所述同步位置确定状态。例如, 基于同步位置连续选取目标数值个码字, 在目标数值个码字中存在第八阈值个码字不满足验证条件, 确定同步位置错误, 更新同步位置, 否则确定同步位置正确。在同步位置正确的情况下, 不满足验证条件的码字的数目比较少, 所以确定不满足验证条件的码字的数目是否达到第八阈值, 可以更快速的确定同步位置是否错误。此处, 目标数值个码字在数据序列中可以是连续的, 也可以是不连续的。

具体的, 参见图 16 所示的状态图, {失锁检测状态}包括{第五计数器重置子状态}和{第六码字验证子状态}, {第五计数器重置子状态}为对计数器进行重置的状态, {第六码字验证子状态}为对码字进行验证的状态。

在确定出同步位置后, 进入{第五计数器重置子状态}。在{第五计数器重置子状态}下, 设置[第五码字计数器]的初始值和[第二无效码字计数器]的初始值, 此处两个初始值可以为 0。

基于同步位置从数据序列中选取一个码字, 为了方便描述基于同步位置选取的码字称为第二码字。可选的, 基于同步位置从数据序列的第四子序列中选取一个码字。第四子序列与前文中的第一子序列以及第三子序列均不相同, 且第四子序列位于第三子序列之后。

响应于选取出第二码字, 进入{第六码字验证子状态}。在{第六码字验证子状态}下, 对第二码字进行验证, 对第二码字验证的方式与对第一码字验证的方式相同, 此处不再赘述。

响应于第二码字不满足验证条件, 将[第五码字计数器]的计数值增加 1, 将[第二无效码字计数器]的计数值增加 1。响应于[第五码字计数器]的计数值等于目标数值, 且[第二无效码字计数器]的计数值小于第八阈值, 执行对同步位置进行下一次验证的操作。此处, [第五码字计数器]的计数值等于目标数值, 且[第二无效码字计数器]的计数值小于第八阈值, 说明目标数值个码字中存在不满足验证条件的码字比较少, 同步位置正确。

响应于[第二无效码字计数器]的计数值等于第八阈值,执行重新确定数据序列的同步位置的操作。此处,[第二无效码字计数器]的计数值等于第八阈值,说明不满足验证条件的码字数目比较多,同步位置不准确。

可选的,{失锁检测状态}还包括{第二码字无效子状态},{第二码字无效子状态}为码字不满足验证条件时进入的状态。响应于第二码字不满足验证条件,进入{第二码字无效子状态}。在{第二码字无效子状态}下,将[第五码字计数器]的计数值增加1,将[第二无效码字计数器]的计数值增加1。

可选的,在{第二码字无效子状态}下,响应于[第五码字计数器]的计数值小于目标数值,且[第一无效码字计数器]的计数值小于第八阈值,重新进入{第六码字验证子状态},执行对基于同步位置选取的下一个第二码字进行验证的操作。

可选的,{失锁检测状态}还包括{第四码字有效子状态},{第四码字有效子状态}为码字满足验证条件时进入的状态。响应于第二码字满足验证条件,进入{第四码字有效子状态}。在{第四码字有效子状态}下,将[第五码字计数器]的计数值增加1。响应于[第五码字计数器]的计数值等于目标数值,执行对同步位置进行下一次验证的操作。响应于[第五码字计数器]的计数值小于目标数值,执行对基于同步位置选取的下一个第二码字进行验证的操作。

可选的,响应于第二码字满足验证条件,将[第五码字有效变量]的值置为第二值。响应于[第五码字有效变量]的值为第二值,进入{第四码字有效子状态}。

响应于第二码字不满足验证条件,将[第五码字有效变量]的值置为第一值。响应于[第五码字有效变量]的值为第一值,进入{第二码字无效子状态}。

可选的,响应于基于同步位置选取一个第二码字后,将[第六码字待验证变量]的值置为第二值,响应于[第六码字待验证变量]的值为第二值,进入{第六码字验证子状态}。或者,在{第四码字有效子状态}下,响应于[第六码字待验证变量]的值为第二值,且[第五码字计数器]的计数值小于目标数值,重新进入{第六码字验证子状态}。或者,在{第二码字无效子状态}下,响应于[第六码字待验证变量]的值为第二值,且[第五码字计数器]的计数值小于目标数值,且[第二无效码字计数器]的计数值小于第八阈值,重新进入{第六码字验证子状态}。在{第六码字验证子状态}下,将[第六码字待验证变量]的值置为第一值。

可选的,响应于[第二同步锁定参数]的值为第二值,进入{第五计数器重置子状态}。或者,响应于[第三同步锁定参数]的值为第二值,进入{第五计数器重置子状态}。响应于[第四同步锁定参数]的值为第二值,进入{第五计数器重置子状态}。或者,响应于第二转换信号,进入{第五计数器重置子状态},第二转换信号为确定位置确定完成时输出的信号。

可选的,响应于[第二无效码字计数器]的计数值等于第八阈值,输出第三转换信号,第三转换信号用于指示执行重新确定数据序列的同步位置的操作。

可选的,失锁检测状态还包括{第三重新开始子状态},{第三重新开始子状态}为输出重新确定同步位置的信号的状态。响应于[第一无效码字计数器]的计数值等于第八阈值,进入{第三重新开始子状态}。在{第三重新开始子状态}下,将[重新同步锁定变量]的值置为第二值。[重新同步锁定变量]的值为第二值指示重新确定同步位置。

在图16所示的状态图中,{第五计数器重置子状态}表示为 RESET_CNT_5,{第六码字验证子状态}表示为 CW_CHECK_6,{第四码字有效子状态}表示为 VALID_CW_4,{第二码字无效子状态}表示为 INVALID_CW_2。[第六码字待验证变量]表示为 test_cw_6,[第五码字

计数器]的计数值表示为 `cw_cnt_5`, [第二无效码字计数器]的计数值表示为 `invalid_cw_cnt_2`, [重新同步锁定变量]表示为 `restart_fec_lock`, [第五码字有效变量]的值是第一值对应 `valid_cw_5`, [第五码字有效变量]的值是第二值对应 `valid_cw_5`。

方式二, 基于同步位置连续选取目标数值个码字, 遍历目标数值个码字, 在目标数值个码字中存在第十一阈值个码字满足验证条件, 确定同步位置正确, 否则确定同步位置错误, 第十一阈值大于第八阈值。

具体的, 参见图 17 所示的状态图, {失锁检测状态}包括{第七计数器重置子状态}和{第八码字验证子状态}, {第七计数器重置子状态}为对计数器进行重置的状态, {第八码字验证子状态}为对码字进行验证的状态。

在确定出同步位置后, 进入{第七计数器重置子状态}。在{第七计数器重置子状态}下, 设置[第七码字计数器]的初始值和[第五有效码字计数器]的初始值, 此处两个初始值可以为 0。

基于同步位置从数据序列中选取一个码字, 为了方便描述, 基于同步位置选取的码字称为第二码字。可选的, 基于同步位置从数据序列的第四子序列中选取一个码字。第四子序列与前文中的第一子序列以及第三子序列均不相同, 且第四子序列位于第三子序列之后。

响应于选取第二码字, 进入{第八码字验证子状态}。在{第八码字验证子状态}下, 对第二码字进行验证, 将[第七码字计数器]的计数值增加 1, 对第二码字验证的方式与对第一测试块验证的方式相同, 此处不再赘述。

在{第八码字验证子状态}下, 响应于第二码字满足验证条件, 将[第五有效码字计数器]的计数值增加 1。

响应于[第七码字计数器]的计数值小于目标数值, 对基于同步位置选取的下一个第二码字进行验证。

响应于[第七码字计数器]的计数值等于目标数值, 且[第五有效码字计数器]的计数值大于或等于第十一阈值, 继续对同步位置进行下一次验证。

响应于[第七码字计数器]的计数值等于目标数值, 且[第五有效码字计数器]的计数值小于第十一阈值, 输出第三转换信号, 第三转换信号用于指示执行重新确定数据序列中的同步位置的操作。或者, 失锁检测状态还包括{第四重新开始子状态}。响应于[第七码字计数器]的计数值等于目标数值, 且[第五有效码字计数器]的计数值小于第十一阈值, 重新进入{第四重新开始子状态}。在{第四重新开始子状态}下, 将[重新同步锁定变量]的值置为第二值。[重新同步锁定变量]的值为第二值指示重新确定同步装置。

可选的, 响应于基于同步位置选取一个第二码字后, 将[第七码字待验证变量]的值置为第二值(如 `true`), 响应于[第七码字待验证变量]的值为第二值, 进入{第八码字验证子状态}。或者, 响应于[第七码字待验证变量]的值为第二值, 且[第七码字计数器]的计数值小于目标数值, 重新进入{第八码字验证子状态}。在{第八码字验证子状态}下, 将[第七码字待验证变量]的值置为第一值(如 `false`)。

在图 17 所示的状态图中, {第七计数器重置子状态}表示为 `RESET_CNT_7`, {第八码字验证子状态}表示为 `CW_CHECK_8`。[第七码字待验证变量]表示为 `test_cw_7`, [第七码字计数器]的计数值表示为 `cw_cnt_7`, [第五有效码字计数器]的计数值表示为 `valid_cw_cnt_5`, 第二码字满足验证条件表示为 `valid_cw`, “if” 表示若或如果。

方式三, 逐个验证基于同步位置选取的码字, 确定累计满足验证条件的码字数目, 直到

最多目标数值个码字。在这个过程中，满足验证条件的码字数目达到第十一阈值，确定同步位置正确。

需要说明的是，上述同步位置确定过程与失锁检测过程可以任意组合。另外，上述同步位置确定以及失锁检测可以使用同一个状态机实现。或者，通过两个子状态机实现，两个子状态机分别为子状态机 1 和子状态机 2，子状态机 1 用于实现确定同步位置的过程，子状态机 2 用于实现失锁检测的过程。或者，通过三个子状态机实现，三个子状态机分别为子状态机 3、子状态机 4 和子状态机 5，子状态机 3 用于实现确定备选位置的过程，子状态机 4 用于实现基于备选位置确定同步位置的过程，子状态机 5 用于实现失锁检测的过程。

例如，图 18 提供了一种由一个状态机实现确定同步位置过程的状态图。图 18 所示的状态图的描述参见图 13 和图 14 所示的状态图中的描述，相比图 13 和图 14 增加了{第三同步锁定的成功子状态}，表示为 (FEC_LOCK_GOOD_3)，且将{第一重新开始子状态}替换为{第三移位子状态}。

需要说明的是，在本申请实施例的状态图中，寄存器变量增加的值都是以 1 为例进行示例。

下面描述实现码字同步的装置的结构。

图 19 是本申请实施例提供的实现码字同步的装置的结构图。该装置可以通过硬件的结合实现成为装置中的部分或者全部。本申请实施例提供的装置可以实现本申请实施例图 8 所述的流程，该装置包括：同步位置确定模块 1910 和失锁检测模块 1920，其中：

同步位置确定模块 1910，用于：

响应于启动信号，进入同步位置确定状态；

在所述同步位置确定状态下，确定接收到的数据序列中的同步位置，所述同步位置用于指示所述数据序列中码字的起始位置，具体可以用于实现步骤 801 和步骤 802 的同步位置确定功能以及执行步骤 801 和步骤 802 包含的隐含步骤；

失锁检测模块 1920，用于：响应于确定出所述同步位置，进入失锁检测状态；

在所述失锁检测状态下，对基于所述同步位置选取的多个码字进行验证，响应于验证失败，重新进入所述同步位置确定状态，具体可以用于实现步骤 803 和步骤 804 的失锁检测功能以及执行步骤 803 和步骤 804 包含的隐含步骤。

在一种可能的实现方式中，所述同步位置确定模块 1910，用于：

在所述同步位置确定状态下，在所述数据序列中选取 N 个观察比特，基于所述 N 个观察比特所在的位置，确定所述同步位置，N 为大于或等于 1 的整数。

在一种可能的实现方式中，所述同步位置确定模块 1910，用于：在所述 N 个观察比特所在的位置中，选取所述同步位置。

在一种可能的实现方式中，所述同步位置确定状态包括第一计数器重置子状态和第一码字验证子状态；

所述同步位置确定模块 1910，用于：

在所述第一计数器重置子状态下，设置第一码字计数器的初始值，在所述数据序列中选取一个观察比特；

响应于基于当前的观察比特从所述数据序列中选取出一个第一测试块，进入所述第一码字验证子状态，所述第一测试块的长度等于码字长度，所述第一测试块的起始位置与所述当

前的观察比特所在的位置之间间隔整数个所述码字长度；

在所述第一码字验证子状态下，对所述第一测试块进行验证；

响应于所述第一测试块满足验证条件，将所述第一码字计数器的计数值增加 X，X 为大于或等于 1 的整数；

响应于所述第一码字计数器的计数值等于第一阈值，确定所述当前的观察比特所在的位置为所述同步位置。

在一种可能的实现方式中，所述同步位置确定模块 1910，还用于：

响应于所述第一码字计数器的计数值小于所述第一阈值，重新进入所述第一码字验证子状态，执行对基于所述当前的观察比特选取的下一个第一测试块的验证操作。

在一种可能的实现方式中，所述同步位置确定状态还包括第一码字有效子状态；

所述同步位置确定模块 1910，还用于：响应于所述第一测试块满足所述验证条件，进入所述第一码字有效子状态；

在所述第一码字有效子状态下，将所述第一码字计数器的计数值增加 X。

在一种可能的实现方式中，所述同步位置确定模块 1910，还用于：

响应于所述第一测试块不满足所述验证条件，执行对从所述数据序列中选取的下一个观察比特的验证操作。

在一种可能的实现方式中，所述同步位置确定状态还包括第一移位子状态；

所述同步位置确定模块 1910，还用于：响应于所述第一测试块不满足验证条件，进入所述第一移位子状态；

在所述第一移位子状态下，移位至所述下一个观察比特；

响应于移位至所述下一个观察比特，重新进入所述第一计数器重置子状态，执行对所述下一个观察比特的验证操作。

在一种可能的实现方式中，所述同步位置确定状态还包括第一同步锁定的初始化子状态；

所述同步位置确定模块 1910，还用于：在所述第一计数器重置子状态下，设置第一码字计数器的初始值之前，在所述第一同步锁定的初始化子状态下，将第一同步锁定变量的值置为第一值。

在一种可能的实现方式中，所述同步位置确定状态还包括第一同步锁定的成功子状态；

所述同步位置确定模块 1910，还用于：

响应于所述第一码字计数器的计数值等于所述第一阈值，进入所述第一同步锁定的成功子状态；

在所述第一同步锁定的成功子状态下，将所述第一同步锁定变量的值置为第二值，所述第一同步锁定变量的值为所述第二值，用于指示所述当前的观察比特所在的位置为所述同步位置。

在一种可能的实现方式中，所述同步位置确定模块 1910，还用于：

在所述第一同步锁定的初始化子状态下，将第一码字待验证变量的值置为第三值；

响应于基于当前的观察比特从所述数据序列中选取出一个第一测试块，将所述第一码字待验证变量的值置为第四值；

响应于所述第一码字待验证变量的值为第四值，且所述第一同步锁定变量的值为第一值，进入所述第一码字验证子状态。

在一种可能的实现方式中，所述失锁检测状态包括第二码字验证子状态；

所述失锁检测模块 1920，用于：

响应于确定出所述同步位置，重新进入所述第一计数器重置子状态；

响应于基于所述同步位置，选取出一个第一码字，进入所述第二码字验证子状态；

在所述第二码字验证子状态下，对所述第一码字进行验证。

在一种可能的实现方式中，所述失锁检测模块 1920，还用于：

在所述第一计数器重置子状态下，设置所述第一码字计数器的初始值和第一无效码字计数器的初始值；

所述失锁检测模块 1920，用于：

响应于所述第一码字不满足验证条件，将所述第一码字计数器的计数值增加 Y，将所述第一无效码字计数器的计数值增加 Z，Y 和 Z 均为大于或等于 1 的整数；

响应于在所述第一码字计数器的计数值小于或等于第二阈值的情况下，所述第一无效码字计数器的计数值等于第三阈值，重新进入所述同步位置确定状态，所述第三阈值小于所述第二阈值。

在一种可能的实现方式中，所述同步位置确定状态包括第一移位子状态；

所述失锁检测模块 1920，用于：

进入所述第一移位子状态；

在所述第一移位子状态下，移位至下一个观察比特；

响应于移位至所述下一个观察比特，重新进入所述第一计数器重置子状态，执行对所述下一个观察比特的验证操作。

在一种可能的实现方式中，所述同步位置确定模块 1910，用于：

在所述 N 个观察比特所在的位置中，选取所述数据序列中的备选位置；

基于所述备选位置，确定所述同步位置。

在一种可能的实现方式中，所述 N 大于 1，所述同步位置确定状态包括第二计数器重置子状态和第三码字验证子状态；

所述同步位置确定模块 1910，用于：

在所述第二计数器重置子状态下，设置第二码字计数器的初始值和第一当前有效码字计数器的初始值，从所述数据序列中选取一个观察比特；

响应于基于当前的观察比特从所述数据序列中选取出一个第二测试块，进入所述第三码字验证子状态，所述第二测试块的长度等于码字长度，所述第二测试块的起始位置与所述当前的观察比特所在的位置之间间隔整数个所述码字长度；

在所述第三码字验证子状态下，对所述第二测试块进行验证，将所述第二码字计数器的计数值增加 W，W 为大于或等于 1 的整数，响应于所述第二测试块满足验证条件，将所述第一当前有效码字计数器的计数值增加 P，P 为大于或等于 1 的整数；

响应于所述第二码字计数器的计数值等于第四阈值，将观察比特数目变量的值增加 1，响应于所述第一当前有效码字计数器的计数值大于最大有效码字数目变量的当前值，更新备选同步位置变量的值为所述当前的观察比特所在的位置，将所述最大有效码字数目变量的当前值更新为所述第一当前有效码字计数器的计数值；

响应于所述观察比特数目变量的值等于 N，将所述当前的备选同步位置变量的值确定为

所述备选位置。

在一种可能的实现方式中，所述同步位置确定状态还包括备选位置选择子状态；

所述同步位置确定模块 1910，用于：

响应于所述第二码字计数器的计数值等于第四阈值，进入所述备选位置选择子状态；

在所述备选位置选择子状态下，响应于所述第二码字计数器的计数值等于第四阈值，将观察比特数目变量的值增加 1，响应于所述第一当前有效码字计数器的计数值大于最大有效码字数目变量的当前值，更新备选同步位置变量的值为所述当前的观察比特所在的位置，将所述最大有效码字数目变量的当前值更新为所述第一当前有效码字计数器的计数值。

在一种可能的实现方式中，所述同步位置确定模块 1910，还用于：响应于所述观察比特数目变量的值小于 N，执行对从所述数据序列中选取的下一个观察比特的验证操作。

在一种可能的实现方式中，所述同步位置确定状态还包括第二移位子状态；

所述同步位置确定模块 1910，用于：

响应于所述观察比特数目变量的值小于 N，进入所述第二移位子状态；

在所述第二移位子状态下，移位至所述下一个观察比特；

响应于移位至所述下一个观察比特，重新进入所述第二计数器重置子状态，执行对所述下一个观察比特的验证操作。

在一种可能的实现方式中，所述同步位置确定状态还包括第二同步锁定的初始化子状态；

所述同步位置确定模块 1910，还用于：

在所述第二同步锁定的初始化子状态下，将所述观察比特数目变量的值置为第五值，将所述备选同步位置变量的值置为第六值，将所述最大有效码字数目变量的值置为第七值。

在一种可能的实现方式中，所述同步位置确定模块 1910，还用于：

在所述第二同步锁定的初始化子状态下，将第二码字待验证变量的值置为第三值；

所述同步位置确定模块 1910，用于：

响应于基于所述当前的观察比特从所述数据序列中选取一个第二测试块，将所述第二码字待验证变量的值置为第四值；

响应于所述第二码字待验证变量的值为所述第四值，进入所述第三码字验证子状态。

在一种可能的实现方式中，所述同步位置确定模块 1910，还用于：

响应于所述第二码字计数器的计数值小于所述第四阈值，重新进入所述第三码字验证子状态，执行对基于所述当前的观察比特选取的下一个第二测试块的验证操作。

在一种可能的实现方式中，所述同步位置确定状态包括第三计数器重置子状态和第四码字验证子状态；

所述同步位置确定模块 1910，用于：

在所述第三计数器重置子状态下，设置第三码字计数器的初始值和第一有效码字计数器的初始值，在所述数据序列中选取一个观察比特；

响应于基于当前的观察比特从所述数据序列中选取出一个第三测试块，进入所述第四码字验证子状态，所述第三测试块的长度等于码字长度，所述第三测试块的起始位置与所述当前的观察比特所在的位置之间间隔整数个所述码字长度；

在所述第四码字验证子状态下，对所述第三测试块进行验证；

响应于所述第三测试块满足验证条件，将所述第三码字计数器的计数值增加 M，将所述

第一有效码字计数器的计数值增加 Q , M 和 Q 均为大于或等于 1 的整数;

响应于在所述第三码字计数器的计数值小于或等于第五阈值的情况下, 所述第一有效码字计数器的计数值等于第六阈值, 确定所述当前的观察比特所在的位置为所述备选位置, 所述第五阈值大于或等于所述第六阈值。

在一种可能的实现方式中, 所述同步位置确定状态还包括第二码字有效子状态;

所述同步位置确定模块 1910, 用于:

响应于所述第三测试块满足验证条件, 进入所述第二码字有效子状态;

在所述第二码字有效子状态下, 将所述第三码字计数器的计数值增加 M , 将所述第一有效码字计数器的计数值增加 Q 。

在一种可能的实现方式中, 所述同步位置确定模块 1910, 还用于:

在所述第二码字有效子状态下, 响应于所述第三码字计数器的计数值等于所述第五阈值, 且所述第一有效码字计数器的计数值小于所述第六阈值, 执行对从所述数据序列中选取的下一个观察比特的验证操作。

在一种可能的实现方式中, 所述同步位置确定模块 1910, 还用于:

在所述第二码字有效子状态下, 响应于所述第三码字计数器的计数值小于所述第五阈值, 且所述第一有效码字计数器的计数值小于所述第六阈值, 重新进入所述第四码字验证子状态, 执行对基于所述当前的观察比特选取的下一个第三测试块的验证操作。

在一种可能的实现方式中, 所述同步位置确定状态还包括第一码字无效子状态;

所述同步位置确定模块 1910, 还用于:

响应于所述第三测试块不满足验证条件, 进入所述第一码字无效子状态;

在所述第一码字无效子状态下, 将第三码字计数器的计数值增加 M ;

响应于所述第三码字计数器的计数值等于第五阈值, 执行对从所述数据序列中选取的下一个观察比特的验证操作。

在一种可能的实现方式中, 所述同步位置确定模块 1910, 还用于: 在所述第一码字无效子状态下, 响应于所述第三码字计数器的计数值小于所述第五阈值, 重新进入所述第四码字验证子状态, 执行对基于所述当前的观察比特选取的下一个第三测试块的验证操作。

在一种可能的实现方式中, 所述同步位置确定状态还包括第三移位子状态;

所述同步位置确定模块 1910, 用于:

进入所述第三移位子状态;

在所述第三移位子状态下, 移位至所述下一个观察比特;

响应于移位至所述下一个观察比特, 重新进入所述第三计数器重置子状态, 执行对所述下一个观察比特的验证操作。

在一种可能的实现方式中, 所述同步位置确定状态还包括第三同步锁定的初始化子状态;

所述同步位置确定模块 1910, 还用于:

在所述第三同步锁定的初始化子状态下, 将第三码字待验证变量的值置为第三值。

在一种可能的实现方式中, 所述同步位置确定模块 1910, 用于:

响应于基于当前的观察比特从所述数据序列中选取出一个第三测试块, 将所述第三码字待验证变量的值置为第四值;

响应于所述第三码字待验证变量的值为所述第四值, 进入所述第四码字验证子状态。

在一种可能的实现方式中，所述同步位置确定模块 1910，用于：

对所述备选位置进行验证，在验证成功后，确定所述备选位置为所述同步位置。

在一种可能的实现方式中，所述同步位置确定状态还包括第四计数器重置子状态和第五码字验证子状态；

所述同步位置确定模块 1910，用于：

在所述第四计数器重置子状态下，设置第四码字计数器的初始值，基于所述备选位置从所述数据序列中选取一个第四测试块；

响应于基于所述备选位置从所述数据序列中选取出一个第四测试块，进入所述第五码字验证子状态；

在所述第五码字验证子状态下，对所述第四测试块进行验证；

响应于所述第四测试块满足验证条件，将所述第四码字计数器的计数值增加 R ， R 为大于或等于 1 的整数；

响应于所述第四码字计数器的计数值等于第七阈值，确定所述备选位置为所述同步位置。

在一种可能的实现方式中，所述同步位置确定状态还包括第三码字有效子状态；

所述同步位置确定模块 1910，用于：

响应于所述第四测试块满足验证条件，进入所述第三码字有效子状态；

在所述第三码字有效子状态下，将所述第四码字计数器的计数值增加 R 。

在一种可能的实现方式中，所述同步位置确定模块 1910，还用于：

响应于所述第四码字计数器的计数值小于所述第七阈值，重新进入所述第五码字验证子状态，执行对基于所述备选位置选取的下一个第四测试块的验证操作。

在一种可能的实现方式中，所述同步位置确定模块 1910，还用于：

响应于所述第四测试块不满足所述验证条件，重新选取备选位置。

在一种可能的实现方式中，所述同步位置确定状态还包括第一重新同步子状态；

所述同步位置确定模块 1910，还用于：

响应于所述第四测试块不满足所述验证条件，进入所述第一重新同步子状态；

在所述第一重新同步子状态下，将第一重新同步锁定变量的值置为第二值，所述第一重新同步锁定变量的值为所述第二值，用于指示重新在所述数据序列中选取备选位置。

在一种可能的实现方式中，所述同步位置确定状态还包括第二同步锁定的成功子状态；

所述同步位置确定模块 1910，用于：

响应于所述第四码字计数器的计数值等于第七阈值，进入所述第二同步锁定的成功子状态；

在所述第二同步锁定的成功子状态下，将第二同步锁定变量的值置为第二值，所述第二同步锁定变量的值为所述第二值，用于指示所述备选位置为所述同步位置。

在一种可能的实现方式中，所述同步位置确定模块 1910，用于：

确定所述备选位置为所述同步位置。

在一种可能的实现方式中，所述失锁检测模块 1920，用于：

在所述失锁检测状态下，对基于所述同步位置选取的码字集合中的码字进行验证，码字集合包括目标数值个码字；响应于所述码字集合中不满足验证条件的码字数目达到第八阈值，重新进入所述同步位置确定状态。

在一种可能的实现方式中，所述失锁检测状态包括第五计数器重置子状态和第六码字验证子状态；

所述失锁检测模块 1920，用于：

在所述第五计数器重置子状态下，设置第五码字计数器的初始值和第二无效码字计数器的初始值；

响应于基于所述同步位置从所述数据序列中选取出一个第二码字，进入所述第六码字验证子状态；

在所述第六码字验证子状态下，对所述第二码字进行验证；

响应于所述第二码字不满足验证条件，将所述第五码字计数器的计数值增加 1，将所述第二无效码字计数器的计数值增加 1；

响应于在所述第五码字计数器的计数值小于目标数值的情况下，所述第二无效码字计数器的计数值等于第八阈值，重新进入所述同步位置确定状态。

在一种可能的实现方式中，所述失锁检测状态还包括第二码字无效子状态；

所述失锁检测模块 1920，用于：

响应于所述第二码字不满足验证条件，进入所述第二码字无效子状态；

在所述第二码字无效子状态下，将所述第五码字计数器的计数值增加 1，将所述第二无效码字计数器的计数值增加 1。

在一种可能的实现方式中，所述失锁检测模块 1920，还用于：

响应于所述第二码字满足验证条件，将所述第五码字计数器的计数值增加 1；

响应于所述第五码字计数器的计数值等于所述目标数值，重新进入所述第五计数器重置子状态，对所述同步位置进行下一次验证。

在一种可能的实现方式中，所述失锁检测状态还包括第四码字有效子状态；

所述失锁检测模块 1920，还用于：

响应于所述第二码字满足验证条件，进入所述第四码字有效子状态；

在所述第四码字有效子状态下，将所述第五码字计数器的计数值增加 1。

在一种可能的实现方式中，所述失锁检测模块 1920，还用于：

在所述第四码字有效子状态下，响应于所述第五码字计数器的计数值小于所述目标数值，重新进入所述第六码字验证子状态，执行对基于所述同步位置选取的下一个第二码字的验证操作。

在一种可能的实现方式中，所述失锁检测模块 1920，还用于：

在所述第二码字无效子状态下，响应于所述第五码字计数器的计数值小于所述目标数值，且所述第二无效码字计数器的计数值小于所述第八阈值，重新进入所述第六码字验证子状态，执行对基于所述同步位置选取的下一个第二码字的验证操作。

在一种可能的实现方式中，所述失锁检测模块 1920，还用于：

在所述第二码字无效子状态下，响应于所述第五码字计数器的计数值等于所述目标数值，且所述第二无效码字计数器的计数值小于所述第八阈值，重新进入所述第五计数器重置子状态，对所述同步位置进行下一次验证。

在一种可能的实现方式中，所述验证条件为测试块或码字的校验序列的零元素的数目大于同步阈值；或者，

所述验证条件为测试块或码字的校验序列的非零元素的数目小于非同步阈值；或者，
所述验证条件为测试块或码字为可纠错测试块；或者，
所述验证条件为测试块或码字的重校验位与原始校验位相同。

在一种可能的实现方式中，所述启动信号包括系统重置信号、系统启动信号、数据接收失败信号或重新同步信号。

在一种可能的实现方式中，所述数据序列是经过 FEC 编码的比特流。

下面描述通信设备的结构。

图 20 本申请实施例所涉及的通信设备的一种结构示意图。参见图 20，通信设备 2000 可选地由一般性的总线体系结构来实现。通信设备 2000 包括至少一个处理器 2001、通信总线 2002、存储器 2003 以及至少一个网络接口 2004。图 20 所示结构的通信设备可以是前文中提到的路由器或交换机等。

处理器 2001 例如是通用中央处理器 (central processing unit, CPU)、网络处理器 (network processor, NP)、图形处理器 (graphics processing unit, GPU)、神经网络处理器 (neural-network processing units, NPU)、数据处理单元 (data processing unit, DPU)、微处理器或者一个或多个用于实现本申请方案的集成电路。例如，处理器 2001 包括 ASIC，可编程逻辑器件 (programmable logic device, PLD) 或其组合。PLD 例如是复杂可编程逻辑器件 (complex programmable logic device, CPLD)、现场可编程逻辑门阵列 (field-programmable gate array, FPGA)、通用阵列逻辑 (generic array logic, GAL) 或其任意组合。

通信总线 2002 用于在上述组件之间传送信息。通信总线 2002 可以分为地址总线、数据总线、控制总线等。为便于表示，图 20 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。

存储器 2003 例如是只读存储器 (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，又如是随机存取存储器 (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，又如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)、磁盘存储介质或其它磁存储设备，或者是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 2003 例如是独立存在，并通过通信总线 2002 与处理器 2001 相连接。存储器 2003 也可以和处理器 2001 集成在一起。

网络接口 2004 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。网络接口 2004 包括有线网络接口，还可以包括无线网络接口。其中，有线网络接口例如可以为以太网接口。以太网接口可以是光接口，电接口或其组合。无线网络接口可以为无线局域网 (wireless local area networks, WLAN) 接口，蜂窝网络的网络接口或其组合等。

在具体实现中，作为一种示例，处理器 2001 可以包括一个或多个 CPU。

在具体实现中，作为一种示例，通信设备 2000 可以包括多个处理器。这些处理器中的一个可以是一个单核处理器 (single-CPU)，也可以是一个多核处理器 (multi-CPU)。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 (如计算机程序指令) 的处理核。

图 21 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 21 所示的设备结构示意图。设备 2100 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 2100 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 2100 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 2100 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 2100 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 ASIC 或 FPGA 实现。

在逻辑上,设备 2100 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 2100 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。

主控板用于确定数据序列中的同步位置,同步位置用于指示所述数据序列中码字的起始位置,以及进行失锁检测。详细处理过程请参考上述图 8 中的描述,此处不再赘述。

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(interprocess communication, IPC)通道,主控板和接口板之间通过 IPC 通道进行通信。

本申请实施例中,还提供了一种芯片,芯片用于实现前文中码字同步的方法。

本申请实施例中,还提供了一种芯片系统,该芯片系统包括级联的第一芯片和第二芯片,该第一芯片用于实现前文中码字同步的方法,该第二芯片用于基于 AM 方法进行码字同步。即内层 FEC 采用本申请中的码字同步方法,外层 FEC 采用基于 AM 方式的码字同步方法。

在本申请所提供的几个实施例中,应该理解到,所揭露的系统架构、装置和方法,可以通过其它的方式实现。例如,以上所描述的装置实施例仅仅是示意性的,例如,该模块的划分,仅仅为一种逻辑功能划分,实际实现时可以有另外的划分方式,例如多个模块或组件可以结合或者可以集成到另一个系统,或一些特征可以忽略,或不执行。另外,所显示或讨论的相互之间的耦合或直接耦合或通信连接可以是通过一些接口、装置或模块的间接耦合或通

信连接，也可以是电的，机械的或其它的形式连接。

本申请中术语“第一”和“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”和“第二”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，在不脱离各种示例的范围的情况下，第一测试块可以被称为第二测试块，并且类似地，第二测试块可以被称为第一测试块。第一测试块和第二测试块都可以是测试块，并且在某些情况下，可以是单独且不同的测试块。

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。

以上描述，仅为本申请的示例性的实施方式，但本申请的保护范围并不局限于此，任何熟悉本技术领域的技术人员在本申请揭露的技术范围内，可轻易想到各种等效的修改或替换，这些修改或替换都应涵盖在本申请的保护范围之内。因此，本申请的保护范围应以权利要求的保护范围为准。

权 利 要 求 书

1.一种实现码字同步的方法，其特征在于，所述方法包括：

响应于启动信号，进入同步位置确定状态；

在所述同步位置确定状态下，确定接收到的数据序列中的同步位置，所述同步位置用于指示所述数据序列中码字的起始位置；

响应于确定出所述同步位置，进入失锁检测状态；

在所述失锁检测状态下，对基于所述同步位置选取的多个码字进行验证，响应于验证失败，重新进入所述同步位置确定状态。

2.根据权利要求1所述的方法，其特征在于，所述在所述同步位置确定状态下，确定接收到的数据序列中的同步位置，包括：

在所述同步位置确定状态下，在所述数据序列中选取N个观察比特，基于所述N个观察比特所在的位置，确定所述同步位置，N为大于或等于1的整数。

3.根据权利要求2所述的方法，其特征在于，所述基于所述N个观察比特所在的位置，确定所述同步位置，包括：

在所述N个观察比特所在的位置中，选取所述同步位置。

4.根据权利要求3所述的方法，其特征在于，所述同步位置确定状态包括第一计数器重置子状态和第一码字验证子状态；

所述在所述同步位置确定状态下，在所述数据序列中选取N个观察比特，在所述N个观察比特所在的位置中，选取所述同步位置，包括：

在所述第一计数器重置子状态下，设置第一码字计数器的初始值，在所述数据序列中选取一个观察比特；

响应于基于当前的观察比特从所述数据序列中选取出一个第一测试块，进入所述第一码字验证子状态，所述第一测试块的长度等于码字长度，所述第一测试块的起始位置与所述当前的观察比特所在的位置之间间隔整数个所述码字长度；

在所述第一码字验证子状态下，对所述第一测试块进行验证；

响应于所述第一测试块满足验证条件，将所述第一码字计数器的计数值增加X，X为大于或等于1的整数；

响应于所述第一码字计数器的计数值等于第一阈值，确定所述当前的观察比特所在的位置为所述同步位置。

5.根据权利要求4所述的方法，其特征在于，所述在所述同步位置确定状态下，在所述数据序列中选取N个观察比特，在所述N个观察比特所在的位置中，选取所述同步位置，还包括：

响应于所述第一码字计数器的计数值小于所述第一阈值，重新进入所述第一码字验证子状态，执行对基于所述当前的观察比特选取的下一个第一测试块的验证操作。

6.根据权利要求4或5所述的方法，其特征在于，所述同步位置确定状态还包括第一码字有效子状态；

所述响应于所述第一测试块满足所述验证条件，将所述第一码字计数器的计数值增加X，包括：

响应于所述第一测试块满足所述验证条件, 进入所述第一码字有效子状态;

在所述第一码字有效子状态下, 将所述第一码字计数器的计数值增加 X。

7. 根据权利要求 4 至 6 任一项所述的方法, 其特征在于, 所述在所述同步位置确定状态下, 在所述数据序列中选取 N 个观察比特, 在所述 N 个观察比特所在的位置中, 选取所述同步位置, 还包括:

响应于所述第一测试块不满足所述验证条件, 执行对从所述数据序列中选取的下一个观察比特的验证操作。

8. 根据权利要求 7 所述的方法, 其特征在于, 所述同步位置确定状态还包括第一移位子状态;

所述响应于所述第一测试块不满足所述验证条件, 执行对从所述数据序列中选取的下一个观察比特的验证操作, 包括:

响应于所述第一测试块不满足验证条件, 进入所述第一移位子状态;

在所述第一移位子状态下, 移位至所述下一个观察比特;

响应于移位至所述下一个观察比特, 重新进入所述第一计数器重置子状态, 执行对所述下一个观察比特的验证操作。

9. 根据权利要求 4 至 8 任一项所述的方法, 其特征在于, 所述同步位置确定状态还包括第一同步锁定的初始化子状态;

所述在所述第一计数器重置子状态下, 设置第一码字计数器的初始值之前, 还包括:

在所述第一同步锁定的初始化子状态下, 将第一同步锁定变量的值置为第一值。

10. 根据权利要求 9 所述的方法, 其特征在于, 所述同步位置确定状态还包括第一同步锁定的成功子状态;

所述响应于所述第一码字计数器的计数值等于第一阈值, 确定所述当前的观察比特所在的位置为所述同步位置, 包括:

响应于所述第一码字计数器的计数值等于所述第一阈值, 进入所述第一同步锁定的成功子状态;

在所述第一同步锁定的成功子状态下, 将所述第一同步锁定变量的值置为第二值, 所述第一同步锁定变量的值为所述第二值, 用于指示所述当前的观察比特所在的位置为所述同步位置。

11. 根据权利要求 9 或 10 所述的方法, 其特征在于, 所述方法还包括:

在所述第一同步锁定的初始化子状态下, 将第一码字待验证变量的值置为第三值;

所述响应于基于当前的观察比特从所述数据序列中选取出一个第一测试块, 进入所述第一码字验证子状态, 包括:

响应于基于当前的观察比特从所述数据序列中选取出一个第一测试块, 将所述第一码字待验证变量的值置为第四值;

响应于所述第一码字待验证变量的值为第四值, 且所述第一同步锁定变量的值为第一值, 进入所述第一码字验证子状态。

12. 根据权利要求 4 至 11 任一项所述的方法, 其特征在于, 所述失锁检测状态包括第二码字验证子状态;

所述在所述失锁检测状态下, 对基于所述同步位置选取的多个码字进行验证, 包括:

响应于确定出所述同步位置，重新进入所述第一计数器重置子状态；

响应于基于所述同步位置，选取出一个第一码字，进入所述第二码字验证子状态；

在所述第二码字验证子状态下，对所述第一码字进行验证。

13. 根据权利要求 12 所述的方法，其特征在于，所述在所述失锁检测状态下，对基于所述同步位置选取的多个码字进行验证，还包括：

在所述第一计数器重置子状态下，设置所述第一码字计数器的初始值和第一无效码字计数器的初始值；

所述响应于验证失败，重新进入所述同步位置确定状态，包括：

响应于所述第一码字不满足验证条件，将所述第一码字计数器的计数值增加 Y，将所述第一无效码字计数器的计数值增加 Z，Y 和 Z 均为大于或等于 1 的整数；

响应于在所述第一码字计数器的计数值小于或等于第二阈值的情况下，所述第一无效码字计数器的计数值等于第三阈值，重新进入所述同步位置确定状态，所述第三阈值小于所述第二阈值。

14. 根据权利要求 13 所述的方法，其特征在于，所述同步位置确定状态包括第一移位子状态；

所述重新进入所述同步位置确定状态，包括：

进入所述第一移位子状态；

在所述第一移位子状态下，移位至下一个观察比特；

响应于移位至所述下一个观察比特，重新进入所述第一计数器重置子状态，执行对所述下一个观察比特的验证操作。

15. 根据权利要求 2 所述的方法，其特征在于，所述基于所述 N 个观察比特所在的位置，确定所述同步位置，包括：

在所述 N 个观察比特所在的位置中，选取所述数据序列中的备选位置；

基于所述备选位置，确定所述同步位置。

16. 根据权利要求 15 所述的方法，其特征在于，所述 N 大于 1，所述同步位置确定状态包括第二计数器重置子状态和第三码字验证子状态；

所述在所述同步位置确定状态下，在所述数据序列中选取 N 个观察比特，在所述 N 个观察比特所在的位置中，选取所述数据序列中的备选位置，包括：

在所述第二计数器重置子状态下，设置第二码字计数器的初始值和第一当前有效码字计数器的初始值，从所述数据序列中选取一个观察比特；

响应于基于当前的观察比特从所述数据序列中选取出一个第二测试块，进入所述第三码字验证子状态，所述第二测试块的长度等于码字长度，所述第二测试块的起始位置与所述当前的观察比特所在的位置之间间隔整数个所述码字长度；

在所述第三码字验证子状态下，对所述第二测试块进行验证，将所述第二码字计数器的计数值增加 W，W 为大于或等于 1 的整数，响应于所述第二测试块满足验证条件，将所述第一当前有效码字计数器的计数值增加 P，P 为大于或等于 1 的整数；

响应于所述第二码字计数器的计数值等于第四阈值，将观察比特数目变量的值增加 1，响应于所述第一当前有效码字计数器的计数值大于最大有效码字数目变量的当前值，更新备选同步位置变量的值为所述当前的观察比特所在的位置，将所述最大有效码字数目变量的当

前值更新为所述第一当前有效码字计数器的计数值；

响应于所述观察比特数目变量的值等于 N，将所述当前的备选同步位置变量的值确定为所述备选位置。

17.根据权利要求 16 所述的方法，其特征在于，所述同步位置确定状态还包括备选位置选择子状态；

所述响应于所述第二码字计数器的计数值等于第四阈值，将观察比特数目变量的值增加 1，响应于所述第一当前有效码字计数器的计数值大于最大有效码字数目变量的当前值，更新备选同步位置变量的值为所述当前的观察比特所在的位置，将所述最大有效码字数目变量的当前值更新为所述第一当前有效码字计数器的计数值，包括：

响应于所述第二码字计数器的计数值等于第四阈值，进入所述备选位置选择子状态；

在所述备选位置选择子状态下，响应于所述第二码字计数器的计数值等于第四阈值，将观察比特数目变量的值增加 1，响应于所述第一当前有效码字计数器的计数值大于最大有效码字数目变量的当前值，更新备选同步位置变量的值为所述当前的观察比特所在的位置，将所述最大有效码字数目变量的当前值更新为所述第一当前有效码字计数器的计数值。

18.根据权利要求 16 或 17 所述的方法，其特征在于，所述在所述同步位置确定状态下，在所述数据序列中选取 N 个观察比特，在所述 N 个观察比特所在的位置中，选取所述数据序列中的备选位置，还包括：

响应于所述观察比特数目变量的值小于 N，执行对从所述数据序列中选取的下一个观察比特的验证操作。

19.根据权利要求 18 所述的方法，其特征在于，所述同步位置确定状态还包括第二移位子状态；

所述响应于所述观察比特数目变量的值小于 N，执行对从所述数据序列中选取的下一个观察比特的验证操作，包括：

响应于所述观察比特数目变量的值小于 N，进入所述第二移位子状态；

在所述第二移位子状态下，移位至所述下一个观察比特；

响应于移位至所述下一个观察比特，重新进入所述第二计数器重置子状态，执行对所述下一个观察比特的验证操作。

20.根据权利要求 16 至 19 任一项所述的方法，其特征在于，所述同步位置确定状态还包括第二同步锁定的初始化子状态；

所述在所述第二计数器重置子状态下，设置第二码字计数器的初始值和第一当前有效码字计数器的初始值之前，还包括：

在所述第二同步锁定的初始化子状态下，将所述观察比特数目变量的值置为第五值，将所述备选同步位置变量的值置为第六值，将所述最大有效码字数目变量的值置为第七值。

21.根据权利要求 20 所述的方法，其特征在于，所述方法还包括：

在所述第二同步锁定的初始化子状态下，将第二码字待验证变量的值置为第三值；

所述响应于基于当前的观察比特从所述数据序列中选取出一个第二测试块，进入所述第三码字验证子状态，包括：

响应于基于所述当前的观察比特从所述数据序列中选取一个第二测试块，将所述第二码字待验证变量的值置为第四值；

响应于所述第二码字待验证变量的值为所述第四值，进入所述第三码字验证子状态。

22.根据权利要求 16 至 21 任一项所述的方法，其特征在于，所述在所述同步位置确定状态下，在所述数据序列中选取 N 个观察比特，在所述 N 个观察比特所在的位置中，选取所述数据序列中的备选位置，还包括：

响应于所述第二码字计数器的计数值小于所述第四阈值，重新进入所述第三码字验证子状态，执行对基于所述当前的观察比特选取的下一个第二测试块的验证操作。

23.根据权利要求 15 所述的方法，其特征在于，所述同步位置确定状态包括第三计数器重置子状态和第四码字验证子状态；

所述在所述同步位置确定状态下，在所述数据序列中选取 N 个观察比特，在所述 N 个观察比特所在的位置中，选取所述数据序列中的备选位置，包括：

在所述第三计数器重置子状态下，设置第三码字计数器的初始值和第一有效码字计数器的初始值，在所述数据序列中选取一个观察比特；

响应于基于当前的观察比特从所述数据序列中选取出一个第三测试块，进入所述第四码字验证子状态，所述第三测试块的长度等于码字长度，所述第三测试块的起始位置与所述当前的观察比特所在的位置之间间隔整数个所述码字长度；

在所述第四码字验证子状态下，对所述第三测试块进行验证；

响应于所述第三测试块满足验证条件，将所述第三码字计数器的计数值增加 M，将所述第一有效码字计数器的计数值增加 Q，M 和 Q 均为大于或等于 1 的整数；

响应于在所述第三码字计数器的计数值小于或等于第五阈值的情况下，所述第一有效码字计数器的计数值等于第六阈值，确定所述当前的观察比特所在的位置为所述备选位置，所述第五阈值大于或等于所述第六阈值。

24.根据权利要求 23 所述的方法，其特征在于，所述同步位置确定状态还包括第二码字有效子状态；

所述响应于所述第三测试块满足验证条件，将所述第三码字计数器的计数值增加 M，将所述第一有效码字计数器的计数值增加 Q，包括：

响应于所述第三测试块满足验证条件，进入所述第二码字有效子状态；

在所述第二码字有效子状态下，将所述第三码字计数器的计数值增加 M，将所述第一有效码字计数器的计数值增加 Q。

25.根据权利要求 24 所述的方法，其特征在于，所述在所述同步位置确定状态下，在所述数据序列中选取 N 个观察比特，在所述 N 个观察比特所在的位置中，选取所述数据序列中的备选位置，还包括：

在所述第二码字有效子状态下，响应于所述第三码字计数器的计数值等于所述第五阈值，且所述第一有效码字计数器的计数值小于所述第六阈值，执行对从所述数据序列中选取的下一个观察比特的验证操作。

26.根据权利要求 24 或 25 所述的方法，其特征在于，所述在所述同步位置确定状态下，在所述数据序列中选取 N 个观察比特，在所述 N 个观察比特所在的位置中，选取所述数据序列中的备选位置，还包括：

在所述第二码字有效子状态下，响应于所述第三码字计数器的计数值小于所述第五阈值，且所述第一有效码字计数器的计数值小于所述第六阈值，重新进入所述第四码字验证子状态，

执行对基于所述当前的观察比特选取的下一个第三测试块的验证操作。

27.根据权利要求 23 至 26 任一项所述的方法,其特征在于,所述同步位置确定状态还包括第一码字无效子状态;

所述在所述同步位置确定状态下,在所述数据序列中选取 N 个观察比特,在所述 N 个观察比特所在的位置中,选取所述数据序列中的备选位置,还包括:

响应于所述第三测试块不满足验证条件,进入所述第一码字无效子状态;

在所述第一码字无效子状态下,将所述第三码字计数器的计数值增加 M;

响应于所述第三码字计数器的计数值等于第五阈值,执行对从所述数据序列中选取的下一个观察比特的验证操作。

28.根据权利要求 27 所述的方法,其特征在于,所述在所述同步位置确定状态下,在所述数据序列中选取 N 个观察比特,基于所述 N 个观察比特,确定所述数据序列中的备选位置,还包括:

在所述第一码字无效子状态下,响应于所述第三码字计数器的计数值小于所述第五阈值,重新进入所述第四码字验证子状态,执行对基于所述当前的观察比特选取的下一个第三测试块的验证操作。

29.根据权利要求 25 或 27 所述的方法,其特征在于,所述同步位置确定状态还包括第三移位子状态;

所述执行对从所述数据序列中选取的下一个观察比特的验证操作,包括:

进入所述第三移位子状态;

在所述第三移位子状态下,移位至所述下一个观察比特;

响应于移位至所述下一个观察比特,重新进入所述第三计数器重置子状态,执行对所述下一个观察比特的验证操作。

30.根据权利要求 23 至 29 任一项所述的方法,其特征在于,所述同步位置确定状态还包括第三同步锁定的初始化子状态;

所述在所述第三计数器重置子状态下,设置第三码字计数器的初始值和第一有效码字计数器的初始值之前,还包括:

在所述第三同步锁定的初始化子状态下,将第三码字待验证变量的值置为第三值。

31.根据权利要求 30 所述的方法,其特征在于,所述响应于基于当前的观察比特从所述数据序列中选取出一个第三测试块,进入所述第四码字验证子状态,包括:

响应于基于当前的观察比特从所述数据序列中选取出一个第三测试块,将所述第三码字待验证变量的值置为第四值;

响应于所述第三码字待验证变量的值为所述第四值,进入所述第四码字验证子状态。

32.根据权利要求 15 至 31 任一项所述的方法,其特征在于,所述基于所述备选位置,确定所述同步位置,包括:

对所述备选位置进行验证,在验证成功后,确定所述备选位置为所述同步位置。

33.根据权利要求 32 所述的方法,其特征在于,所述同步位置确定状态还包括第四计数器重置子状态和第五码字验证子状态;

所述对所述备选位置进行验证,在验证成功后,确定所述备选位置为所述同步位置,包括:

在所述第四计数器重置子状态下，设置第四码字计数器的初始值，基于所述备选位置从所述数据序列中选取一个第四测试块；

响应于基于所述备选位置从所述数据序列中选取出一个第四测试块，进入所述第五码字验证子状态；

在所述第五码字验证子状态下，对所述第四测试块进行验证；

响应于所述第四测试块满足验证条件，将所述第四码字计数器的计数值增加 R ， R 为大于或等于 1 的整数；

响应于所述第四码字计数器的计数值等于第七阈值，确定所述备选位置为所述同步位置。

34. 根据权利要求 33 所述的方法，其特征在于，所述同步位置确定状态还包括第三码字有效子状态；

所述响应于所述第四测试块满足验证条件，将所述第四码字计数器的计数值增加 R ，包括：

响应于所述第四测试块满足验证条件，进入所述第三码字有效子状态；

在所述第三码字有效子状态下，将所述第四码字计数器的计数值增加 R 。

35. 根据权利要求 33 或 34 所述的方法，其特征在于，所述对所述备选位置进行验证，在验证成功后，确定所述备选位置为所述同步位置，还包括：

响应于所述第四码字计数器的计数值小于所述第七阈值，重新进入所述第五码字验证子状态，执行对基于所述备选位置选取的下一个第四测试块的验证操作。

36. 根据权利要求 33 至 35 任一项所述的方法，其特征在于，所述对所述备选位置进行验证，在验证成功后，确定所述备选位置为所述同步位置，还包括：

响应于所述第四测试块不满足所述验证条件，重新选取备选位置。

37. 根据权利要求 36 所述的方法，其特征在于，所述同步位置确定状态还包括第一重新同步子状态；

所述响应于所述第四测试块不满足所述验证条件，重新选取备选位置，包括：

响应于所述第四测试块不满足所述验证条件，进入所述第一重新同步子状态；

在所述第一重新同步子状态下，将第一重新同步锁定变量的值置为第二值，所述第一重新同步锁定变量的值为所述第二值，用于指示重新在所述数据序列中选取备选位置。

38. 根据权利要求 33 至 37 任一项所述的方法，其特征在于，所述同步位置确定状态还包括第二同步锁定的成功子状态；

所述响应于所述第四码字计数器的计数值等于第七阈值，确定所述备选位置为所述同步位置，包括：

响应于所述第四码字计数器的计数值等于第七阈值，进入所述第二同步锁定的成功子状态；

在所述第二同步锁定的成功子状态下，将第二同步锁定变量的值置为第二值，所述第二同步锁定变量的值为所述第二值，用于指示所述备选位置为所述同步位置。

39. 根据权利要求 15 至 31 任一项所述的方法，其特征在于，所述基于所述备选位置，确定所述同步位置，包括：

确定所述备选位置为所述同步位置。

40. 根据权利要求 1 至 39 任一项所述的方法，其特征在于，所述在所述失锁检测状态下，

对基于所述同步位置选取的多个码字进行验证，响应于验证失败，重新进入所述同步位置确定状态，包括：

在所述失锁检测状态下，对基于所述同步位置选取的码字集合中的码字进行验证，所述码字集合包括目标数值个码字；响应于所述码字集合中不满足验证条件的码字数目达到第八阈值，重新进入所述同步位置确定状态。

41.根据权利要求 40 任一项所述的方法，其特征在于，所述失锁检测状态包括第五计数器重置子状态和第六码字验证子状态；

所述在所述失锁检测状态下，对基于所述同步位置选取的码字集合中的码字进行验证，所述码字集合包括目标数值个码字；响应于所述码字集合中不满足验证条件的码字数目达到第八阈值，重新进入所述同步位置确定状态，包括：

在所述第五计数器重置子状态下，设置第五码字计数器的初始值和第二无效码字计数器的初始值；

响应于基于所述同步位置从所述数据序列中选取出一个第二码字，进入所述第六码字验证子状态；

在所述第六码字验证子状态下，对所述第二码字进行验证；

响应于所述第二码字不满足验证条件，将所述第五码字计数器的计数值增加 1，将所述第二无效码字计数器的计数值增加 1；

响应于在所述第五码字计数器的计数值小于目标数值的情况下，所述第二无效码字计数器的计数值等于第八阈值，重新进入所述同步位置确定状态。

42.根据权利要求 41 所述的方法，其特征在于，所述失锁检测状态还包括第二码字无效子状态；

所述响应于所述第二码字不满足验证条件，将所述第五码字计数器的计数值增加 1，将所述第二无效码字计数器的计数值增加 1，包括：

响应于所述第二码字不满足验证条件，进入所述第二码字无效子状态；

在所述第二码字无效子状态下，将所述第五码字计数器的计数值增加 1，将所述第二无效码字计数器的计数值增加 1。

43.根据权利要求 41 或 42 所述的方法，其特征在于，所述在所述失锁检测状态下，对基于所述同步位置选取的码字集合中的码字进行验证，所述码字集合包括目标数值个码字；响应于所述码字集合中不满足验证条件的码字数目达到第八阈值，重新进入所述同步位置确定状态，还包括：

响应于所述第二码字满足验证条件，将所述第五码字计数器的计数值增加 1；

响应于所述第五码字计数器的计数值等于所述目标数值，重新进入所述第五计数器重置子状态，对所述同步位置进行下一次验证。

44.根据权利要求 43 所述的方法，其特征在于，所述失锁检测状态还包括第四码字有效子状态；

所述响应于所述第二码字满足验证条件，将所述第五码字计数器的计数值增加 1，包括：

响应于所述第二码字满足验证条件，进入所述第四码字有效子状态；

在所述第四码字有效子状态下，将所述第五码字计数器的计数值增加 1。

45.根据权利要求 43 或 44 所述的方法，其特征在于，所述在所述失锁检测状态下，对基

于所述同步位置选取的码字集合中的码字进行验证,所述码字集合包括目标数值个码字;响应于所述码字集合中不满足验证条件的码字数目达到第八阈值,重新进入所述同步位置确定状态,还包括:

在所述第四码字有效子状态下,响应于所述第五码字计数器的计数值小于所述目标数值,重新进入所述第六码字验证子状态,执行对基于所述同步位置选取的下一个第二码字的验证操作。

46.根据权利要求 41 至 45 任一项所述的方法,其特征在于,所述在所述失锁检测状态下,对基于所述同步位置选取的码字集合中的码字进行验证,所述码字集合包括目标数值个码字;响应于所述码字集合中不满足验证条件的码字数目达到第八阈值,重新进入所述同步位置确定状态,还包括:

在所述第二码字无效子状态下,响应于所述第五码字计数器的计数值小于所述目标数值,且所述第二无效码字计数器的计数值小于所述第八阈值,重新进入所述第六码字验证子状态,执行对基于所述同步位置选取的下一个第二码字的验证操作。

47.根据权利要求 41 至 46 任一项所述的方法,其特征在于,所述在所述失锁检测状态下,对基于所述同步位置选取的码字集合中的码字进行验证,所述码字集合包括目标数值个码字;响应于所述码字集合中不满足验证条件的码字数目达到第八阈值,重新进入所述同步位置确定状态,还包括:

在所述第二码字无效子状态下,响应于所述第五码字计数器的计数值等于所述目标数值,且所述第二无效码字计数器的计数值小于所述第八阈值,重新进入所述第五计数器重置子状态,对所述同步位置进行下一次验证。

48.根据权利要求 4 至 14、16 至 31、33 至 38 和 40 至 47 中任一项所述的方法,其特征在于,所述验证条件为测试块或码字的校验序列的零元素的数目大于同步阈值;或者,所述验证条件为测试块或码字的校验序列的非零元素的数目小于非同步阈值;或者,所述验证条件为测试块或码字为可纠错测试块;或者,所述验证条件为测试块或码字的重校验位与原始校验位相同。

49.根据权利要求 1 至 48 任一项所述的方法,其特征在于,所述启动信号包括系统重置信号、系统启动信号、数据接收失败信号或重新同步信号。

50.根据权利要求 1 至 49 任一项所述的方法,其特征在于,所述数据序列是经过前向纠错 FEC 编码的比特流。

51.一种通信设备,其特征在于,所述通信设备包括处理器和存储器;

所述存储器中存储有至少一条计算机指令;

所述计算机指令由所述处理器加载并执行以实现权利要求 1 至 50 任一项所述的方法。

52.一种芯片,其特征在于,所述芯片用于实现权利要求 1 至 50 任一项所述的方法。

53.一种芯片系统,其特征在于,所述芯片系统包括级联的第一芯片和第二芯片;

所述第一芯片用于实现权利要求 1 至 50 任一项所述的方法;

所述第二芯片用于基于对齐标记 AM 方式进行码字同步。