



(12) 发明专利申请

(10) 申请公布号 CN 115913459 A

(43) 申请公布日 2023. 04. 04

(21) 申请号 202211306073.4

(22) 申请日 2022.10.24

(71) 申请人 山东浪潮科学研究院有限公司

地址 250101 山东省济南市高新区浪潮路
1036号浪潮科技园

(72) 发明人 崔秦龙 景利 尚国旗 林海
田炜

(74) 专利代理机构 北京路浩知识产权代理有限
公司 11002

专利代理师 张文玄

(51) Int. Cl.

H04L 1/00 (2006.01)

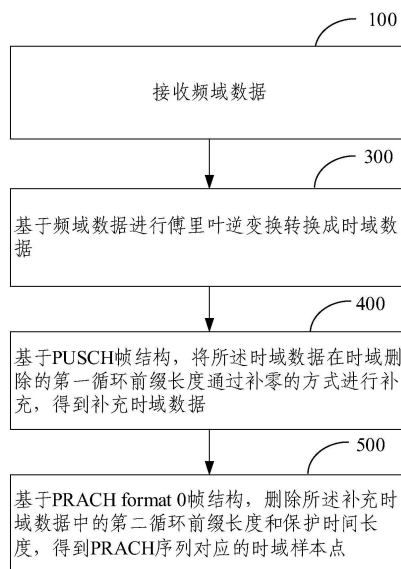
权利要求书2页 说明书10页 附图5页

(54) 发明名称

恢复时域数据的方法、装置和电子设备

(57) 摘要

本发明涉及通信技术领域,提供一种恢复时域数据的方法、装置和电子设备。恢复时域数据的方法包括:接收频域数据;基于频域数据进行傅里叶逆变换转换成时域数据;基于PUSCH帧结构,将所述时域数据在时域删除的第一循环前缀长度通过补零的方式进行补充,得到补充时域数据;基于PRACH format 0帧结构,删除所述补充时域数据中的第二循环前缀长度和保护时间长度,得到PRACH序列对应的时域样本点。本发明实施例用以解决现有技术中进行PRACH format 0格式的时域数据恢复的处理复杂度高,且时域数据恢复会引入其他的干扰数据的缺陷。



1. 一种恢复时域数据的方法,其特征在于,包括:
 - 接收频域数据;
 - 基于频域数据进行傅里叶逆变换转换成时域数据;
 - 基于PUSCH帧结构,将所述时域数据在时域删除的第一循环前缀长度通过补零的方式进行补充,得到补充时域数据;
 - 基于PRACH format 0帧结构,删除所述补充时域数据中的第二循环前缀长度和保护时间长度,得到PRACH序列对应的时域样本点。
2. 根据权利要求1所述的恢复时域数据的方法,其特征在于,所述基于PUSCH帧结构,将所述时域数据在时域删除的第一循环前缀长度通过补零的方式进行补充,得到补充时域数据;
 - 确定PRACH format 0帧结构的时间长度;
 - 基于PUSCH帧结构,将所述时域数据在时域删除的第一循环前缀长度通过补零的方式进行补充,得到所述时间长度的补充时域数据。
3. 根据权利要求2所述的恢复时域数据的方法,其特征在于,所述基于PUSCH帧结构,将所述时域数据在时域删除的第一循环前缀长度通过补零的方式进行补充,得到所述时间长度的补充时域数据,包括:
 - 根据OFDM符号与第一循环前缀长度计算公式,保持每个OFDM符号对应的数值不动,在每个OFDM符号之前把第一循环前缀长度对应的采样点通过补零的方式进行补充,直至得到所述时间长度的补充时域数据。
4. 根据权利要求1所述的恢复时域数据的方法,其特征在于,所述基于PRACH format 0帧结构,删除所述补充时域数据中的第二循环前缀长度和保护时间长度,得到PRACH序列对应的时域样本点,包括:
 - 基于基站采样率和PRACH format 0帧结构,确定第二循环前缀长度和保护时间长度;
 - 删除所述补充时域数据中的所述第二循环前缀长度和所述保护时间长度,得到PRACH序列对应的时域样本点。
5. 根据权利要求1所述的恢复时域数据的方法,其特征在于,所述基于频域数据进行傅里叶逆变换转换成时域数据之前,还包括:
 - 将多个资源元素数据经IFFT_shift操作放置所述频域数据两端。
6. 根据权利要求1所述的恢复时域数据的方法,其特征在于,所述基于PRACH format 0帧结构,删除所述补充时域数据中的第二循环前缀长度和保护时间长度,得到PRACH序列对应的时域样本点之后,还包括:
 - 基于所述PRACH序列对应的时域样本点进行降采样处理。
7. 一种恢复时域数据的装置,其特征在于,包括:
 - 接收模块,用于接收频域数据;
 - 傅里叶逆变换模块,用于基于频域数据进行傅里叶逆变换转换成时域数据;
 - 补零模块,用于基于PUSCH帧结构,将所述时域数据在时域删除的第一循环前缀长度通过补零的方式进行补充,得到补充时域数据;
 - 删除模块,用于基于PRACH format 0帧结构,删除所述补充时域数据中的第二循环前缀长度和保护时间长度,得到PRACH序列对应的时域样本点。

8. 一种电子设备,包括存储器、处理器及存储在所述存储器上并可在所述处理器上运行的计算机程序,其特征在于,所述处理器执行所述程序时实现如权利要求1至6任一项所述的恢复时域数据的方法。

9. 一种非暂态计算机可读存储介质,其上存储有计算机程序,其特征在于,所述计算机程序被处理器执行时实现如权利要求1至6任一项所述的恢复时域数据的方法。

10. 一种计算机程序产品,包括计算机程序,其特征在于,所述计算机程序被处理器执行时实现如权利要求1至6任一项所述的恢复时域数据的方法。

恢复时域数据的方法、装置和电子设备

技术领域

[0001] 本发明涉及通信技术领域,尤其涉及一种恢复时域数据的方法、装置和电子设备。

背景技术

[0002] 5gNR(全称5G New Radio)是第三代合作伙伴计划(3gpp)发布的连续移动宽带演进的一部分,用以满足与等待时间、可靠性、安全性、可扩展性(例如,物联网(iot))和其它要求相关的新要求。5gNR的一些方面是基于4g长期演进(lte)标准的。具体地,用户设备(ue)和基站被配置为执行rach过程以便在ue和基站之间提供上行链路(ul)同步。为此,由ue发送rach前导码序列。

[0003] 然而,rach前导码序列可能对其它ul信道中的上行链路信号引起显著的干扰问题,需要根据rach相关配置周期性在固定时频域资源位置上发送rach序列,考虑到在5gNR中,rach的帧结构与其他上行链路ul信道不一样,要进行单独处理。但是目前部分RU厂家,为了降低实现复杂度,针对前端数据,统一按照122.88M采样率,根据PUSCH帧格式进行处理(100M带宽,30kHz子载波间隔),并未对PRACH进行单独处理,需要BBU(全称Building Base band Unit,译为基带处理单元)对PRACH进行特殊处理和检测。

[0004] 针对PRACH format 0格式,如果要在BBU检测PRACH,需要进行专门的算法研究和设计,目前频域降采样的复杂程度比较高,暂时没有一个可行的频域降采样方案。如果要做降采样处理,需要先将从FPGA得到的频域数据恢复到时域数据后,再进行降采样处理,在这个过程中就涉及到时域样点恢复,在恢复prach时域样本点时,可能会引入其他的干扰数据进来。

[0005] 即现有方法进行PRACH format 0格式的时域数据恢复的处理复杂度高,且时域数据恢复会引入其他的干扰数据。

发明内容

[0006] 本发明提供一种恢复时域数据的方法、装置和电子设备,用以解决现有技术中进行PRACH format 0格式的时域数据恢复的处理复杂度高,且时域数据恢复会引入其他的干扰数据的缺陷。

[0007] 本发明提供一种恢复时域数据的方法,包括:

[0008] 接收频域数据;

[0009] 基于频域数据进行傅里叶逆变换转换成时域数据;

[0010] 基于PUSCH帧结构,将所述时域数据在时域删除的第一循环前缀长度通过补零的方式进行补充,得到补充时域数据;

[0011] 基于PRACH format 0帧结构,删除所述补充时域数据中的第二循环前缀长度和保护时间长度,得到PRACH序列对应的时域样本点。

[0012] 根据本发明提供的一种恢复时域数据的方法,所述基于PUSCH帧结构,将所述时域数据在时域删除的第一循环前缀长度通过补零的方式进行补充,得到补充时域数据;

- [0013] 确定PRACH format 0帧结构的时间长度；
- [0014] 基于PUSCH帧结构,将所述时域数据在时域删除的第一循环前缀长度通过补零的方式进行补充,得到所述时间长度的补充时域数据。
- [0015] 根据本发明提供一种恢复时域数据的方法,所述基于PUSCH帧结构,将所述时域数据在时域删除的第一循环前缀长度通过补零的方式进行补充,得到所述时间长度的补充时域数据,包括:
- [0016] 根据OFDM符号与第一循环前缀长度计算公式,保持每个OFDM符号对应的数值不动,在每个OFDM符号之前把第一循环前缀长度对应的采样点通过补零的方式进行补充,直至得到所述时间长度的补充时域数据。
- [0017] 根据本发明提供一种恢复时域数据的方法,所述基于PRACH format 0帧结构,删除所述补充时域数据中的第二循环前缀长度和保护时间长度,得到PRACH序列对应的时域样本点,包括:
- [0018] 基于基站采样率和PRACH format 0帧结构,确定第二循环前缀长度和保护时间长度；
- [0019] 删除所述补充时域数据中的所述第二循环前缀长度和所述保护时间长度,得到PRACH序列对应的时域样本点。
- [0020] 根据本发明提供一种恢复时域数据的方法,所述基于频域数据进行傅里叶逆变换转换成时域数据之前,还包括:
- [0021] 将多个资源元素数据经IFFT_shift操作放置所述频域数据两端。
- [0022] 根据本发明提供一种恢复时域数据的方法,所述基于PRACH format 0帧结构,删除所述补充时域数据中的第二循环前缀长度和保护时间长度,得到PRACH序列对应的时域样本点之后,还包括:
- [0023] 基于所述PRACH序列对应的时域样本点进行降采样处理。
- [0024] 本发明还提供一种恢复时域数据的装置,包括:
- [0025] 接收模块,用于接收频域数据；
- [0026] 傅里叶逆变换模块,用于基于频域数据进行傅里叶逆变换转换成时域数据；
- [0027] 补零模块,用于基于PUSCH帧结构,将所述时域数据在时域删除的第一循环前缀长度通过补零的方式进行补充,得到补充时域数据；
- [0028] 删除模块,用于基于PRACH format 0帧结构,删除所述补充时域数据中的第二循环前缀长度和保护时间长度,得到PRACH序列对应的时域样本点。
- [0029] 本发明还提供一种电子设备,包括存储器、处理器及存储在存储器上并可在处理器上运行的计算机程序,所述处理器执行所述程序时实现如上述任一种所述恢复时域数据的方法。
- [0030] 本发明还提供一种非暂态计算机可读存储介质,其上存储有计算机程序,该计算机程序被处理器执行时实现如上述任一种所述恢复时域数据的方法。
- [0031] 本发明还提供一种计算机程序产品,包括计算机程序,所述计算机程序被处理器执行时实现如上述任一种所述恢复时域数据的方法。
- [0032] 本发明提供的恢复时域数据的方法、装置和电子设备,通过基于PUSCH帧结构,将所述时域数据在时域删除的第一循环前缀长度通过补零的方式进行补充,得到补充时域数

据;以及基于PRACH format0帧结构,删除所述补充时域数据中的第二循环前缀长度和保护时间长度,得到PRACH序列对应的时域样本点。从而本发明实施例通过对时域数据补零再去循环前缀的方法来恢复PRACH format 0格式数据,实现不同信道在帧格式不同的情况下的时域样本点恢复,既避免引入多余的干扰数据,又降低处理复杂度。

附图说明

[0033] 为了更清楚地说明本发明或现有技术中的技术方案,下面将对实施例或现有技术描述中所需要使用的附图作一简单地介绍,显而易见地,下面描述中的附图是本发明的一些实施例,对于本领域普通技术人员来讲,在不付出创造性劳动的前提下,还可以根据这些附图获得其他的附图。

[0034] 图1是本发明提供的恢复时域数据的方法的流程示意图之一;

[0035] 图2是本发明提供的恢复时域数据的方法的流程示意图之二;

[0036] 图3是本发明提供的PUSCH帧结构的结构示意图;

[0037] 图4是本发明提供的PRACH帧结构的结构示意图;

[0038] 图5是本发明提供的恢复时域数据的方法的流程示意图之三;

[0039] 图6是本发明提供的恢复时域数据的装置的结构示意图;

[0040] 图7是本发明提供的电子设备的结构示意图。

具体实施方式

[0041] 为使本发明的目的、技术方案和优点更加清楚,下面将结合本发明中的附图,对本发明中的技术方案进行清楚、完整地描述,显然,所描述的实施例是本发明一部分实施例,而不是全部的实施例。基于本发明中的实施例,本领域普通技术人员在没有作出创造性劳动前提下所获得的所有其他实施例,都属于本发明保护的范围。

[0042] 下面对本发明实施例中出现的专利术语的含义进行解释。

[0043] 其中,RB(全称为Resource Block,译为资源块)的含义为:频率上连续12个子载波,时域上一个slot,称为1个RB。根据一个子载波带宽是15k可以得出一个RB的带宽为180khz。

[0044] RE(全称为Resource Element,译为资源元素)的含义为:频率上一个子载波及时域上一个symbol(符号),称为一个RE。

[0045] CP(循环前缀):是指一个符号前缀,无线系统中这个前缀在OFDM(全称为Orthogonal Frequency Division Multiplexing,译为正交频分复用)末端是重复的,接收端通常配置是丢弃循环前缀样本。CP循环前缀可用于抵消多重路径传输影响。

[0046] 下面结合图1-图5描述本发明的恢复时域数据的方法。

[0047] 请参照图1,一种恢复时域数据的方法,包括:

[0048] 步骤100、接收频域数据;

[0049] 具体的,基站可通过FPGA(全称为Field Programmable Gate Array,译为现场可编程逻辑门阵列)获取频域数据。

[0050] 步骤300、基于频域数据进行傅里叶逆变换转换成时域数据。

[0051] 具体的,本发明实施例的目的在于物理层要实现PRACH format 0格式(序列长度

839,子载波间隔1.25KHz)。其中,PRACH format 0格式适用于普通半径的小区覆盖,是一种很经典的格式类型,但是目前产品中FPGA侧为了降低处理复杂度,没有把PRACH信道与PUSCH信道分开单独处理(PRACH与PUSCH的子载波间隔可能不一致)。即FPGA侧将PRACH信道对应的数据按照PUSCH数据处理方式删除循环前缀后转成频域数据发给物理层。

[0052] 由于目前PUSCH帧格式只支持30KHz的子载波间隔,所以PRACH帧格式也只能按照30KHz的子载波间隔来配置(PRACH只支持format B4格式)。如果物理层要实现PRACH format 0格式(序列长度839,子载波间隔1.25KHz),需要先将FPGA获取的频域数据转成时域数据。因此本发明实施例基于频域数据进行傅里叶逆变换转换成时域数据。

[0053] 步骤400、基于PUSCH帧结构,将所述时域数据在时域删除的第一循环前缀长度通过补零的方式进行补充,得到补充时域数据。

[0054] 由于FPGA侧将PRACH信道对应的时域数据按照PUSCH数据处理方式去循环前缀后转成频域数据发给物理层。因此本发明实施例的基站基于PUSCH帧结构,将所述时域数据在时域删除的第一循环前缀长度通过补零的方式进行补充,得到补充时域数据。

[0055] 需要说明的是,本发明实施例的第一循环前缀长度与第二循环前缀长度指的都是循环前缀长度。“第一”和“第二”是为了区分PUSCH帧结构和PRACH帧格式的循环前缀长度。

[0056] 步骤500、基于PRACH format 0帧结构,删除所述补充时域数据中的第二循环前缀长度和保护时间长度,得到PRACH序列对应的时域样本点。

[0057] 在获取到补充时域数据得基础上,基站基于PRACH format 0帧结构,删除所述补充时域数据中的第二循环前缀长度和保护时间长度,得到PRACH序列对应的时域样本点。

[0058] 本发明实施例按照PUSCH帧结构,用补零的方式将PUSCH帧结构在时域扣掉的第一循环前缀长度反补回去后,再按照PRACH format 0的帧结构扣掉第二循环前缀长度和保护时间长度,从而避免引入多余的干扰数据,同时降低处理复杂度。

[0059] 由于物理层收到的频域数据是RRU(全称:Remote Radio Unit,译为射频拉远单元)统一按照122.88M采样率处理的,没有对PRACH时域数据单独做处理,但PRACH format 0和PUSCH帧格式又不同,为了恢复PRACH原始时域数据,需要进行补循环前缀及去循环前缀处理,从而为后端流程处理做铺垫。

[0060] 需要说明的是,本发明实施例还可以基于其他PRACH帧结构,例如PRACH format 1帧结构、PRACH format 2帧结构和PRACH format 3帧结构,删除所述补充时域数据中的与PRACH帧结构对应长度的第二循环前缀长度和保护时间长度,得到PRACH序列对应的时域样本点。

[0061] 本发明实施例通过基于PUSCH帧结构,将所述时域数据在时域删除的第一循环前缀长度通过补零的方式进行补充,得到补充时域数据;以及基于PRACH format 0帧结构,删除所述补充时域数据中的第二循环前缀长度和保护时间长度,得到PRACH序列对应的时域样本点。从而本发明实施例通过对时域数据补零再去循环前缀的方法来恢复PRACH format 0格式数据,实现不同信道在帧格式不同的情况下的时域样本点恢复,既避免引入多余的干扰数据,又降低处理复杂度。

[0062] 在本发明实施例的其他方面,请参照图2,步骤300、所述基于频域数据进行傅里叶逆变换转换成时域数据之前,还包括:

[0063] 步骤200、将多个资源元素数据经IFFT_shift操作放置所述频域数据两端。

[0064] 基站先将从符号中获取到的3276个RE (273个RB) 数据经IFFT_shift操作放置4096点两端 (FPGA侧获取到的频域数据), 经傅里叶逆变换IFFT后, 可以得到每个符号对应的时域采样点。

[0065] 在本发明实施例的其他方面, 步骤400、所述基于PUSCH帧结构, 将所述时域数据在时域删除的第一循环前缀长度通过补零的方式进行补充, 得到补充时域数据, 包括:

[0066] 步骤410、确定PRACH format 0帧结构的时间长度;

[0067] 步骤420、基于PUSCH帧结构, 将所述时域数据在时域删除的第一循环前缀长度通过补零的方式进行补充, 得到所述时间长度的补充时域数据。

[0068] 因为PRACH format 0格式长度为1ms (其中, 循环前缀长度、随机接入前导、保护时间长度分别为0.103ms、0.8ms、0.097ms), 所以需要按照PUSCH帧结构删除第一循环前缀长度的方式, 通过补零的方式进行补充, 反补拼接成1ms的补充时域数据。

[0069] 在本发明实施例的其他方面, 步骤420、所述基于PUSCH帧结构, 将所述时域数据在时域删除的第一循环前缀长度通过补零的方式进行补充, 得到所述时间长度的补充时域数据, 包括:

[0070] 根据OFDM符号与第一循环前缀长度计算公式, 保持每个OFDM符号对应的数值不动, 在每个OFDM符号之前把第一循环前缀长度对应的采样点通过补零的方式进行补充, 直至得到所述时间长度的补充时域数据。

[0071] 具体的, OFDM符号的计算公式如公式 (1) 所示:

$$[0072] \quad N_u^\mu = 2048k \cdot 2^{-\mu}; \quad \text{公式 (1)}$$

[0073] 第一循环前缀长度计算公式如公式 (2) 所示:

$$[0074] \quad N_{CP,l}^\mu = \begin{cases} 512k \cdot 2^{-\mu} & \text{extended cyclic prefix} \\ 144k \cdot 2^{-\mu} + 16k & \text{normal cyclic prefix, } l = 0 \text{ or } l = 7 \cdot 2^\mu; \\ 144k \cdot 2^{-\mu} & \text{normal cyclic prefix, } l \neq 0 \text{ and } l \neq 7 \cdot 2^\mu \end{cases}$$

公式 (2)

[0075] 其中 $k = T_s / T_c$, 本发明实施例的基站使用的采样率为122.88MHz, 故 $k = 4$, N_u^μ 表示OFDM符号的采样点数, $N_{CP,l}^\mu$ 表示CP的采样点数, PUSCH配置的子载波间隔为30KHz, 根据表格索引可知 $\mu = 1$, PUSCH帧结构如图3所示。

[0076] 本发明实施例根据OFDM符号与第一循环前缀长度计算公式, 保持OFDM符号对应的数值不动, 在每个符号前把第一循环前缀长度对应的采样点反补回来且每个采样点的数值填0, 一直补到一个子帧的数据 (2个slot数据), 这样PRACH原始的时域样本点采集完成, 1ms时间内共包含122880个采样点。

[0077] 本发明实施例进行原始时域数据拼接的时候, 并不是按照OFDM符号加循环前缀的方式反补循环前缀, 而是通过补零的方式还原PRACH format 0格式的时域数据, 得到通过补零的补充时域数据, 实现不同信道在帧格式不同的情况下的时域样本点恢复, 既避免引入多余的干扰数据, 又降低处理复杂度。

[0078] 在本发明实施例的其他方面,步骤500、所述基于PRACH format0帧结构,删除所述补充时域数据中的第二循环前缀长度和保护时间长度,得到PRACH序列对应的时域样本点,包括:

[0079] 步骤510、基于基站采样率和PRACH format 0帧结构,确定第二循环前缀长度和保护时间长度。

[0080] 基于上述步骤可得基站采样率为122.88MHz,并且 $k=4$ 。PRACH帧结构如下表1所示:

[0081]

| 格式 | L_{RA} | Δf^{RA} | N_u | N_{CP}^{RA} | 支持的 限制 |
|----|----------|-----------------|-------|---------------|-----------|
| | | | | | |

| | | | | | |
|---|-----|---------|----------|--------|-----------|
| | | | | | 集 |
| 0 | 839 | 1.25kHz | 24576k | 3168k | 类型 A、类型 B |
| 1 | 839 | 1.25kHz | 2*24576k | 21024k | 类型 A、类型 B |
| 2 | 839 | 1.25kHz | 4*24576k | 4688k | 类型 A、类型 B |
| 3 | 839 | 5kHz | 4*6144k | 3168k | 类型 A、类型 B |

[0082]

[0083] 表1

[0084] 由表1可知,PRACH format 0帧结构中第二循环前缀长度为 $N_{CP}^{RA}=3168k$;随机接入前导长度为24576k。并且PRACH帧结构如图4可知,那么第二循环前缀长度和保护时间长度分别为3168k和2976K。

[0085] 步骤520、删除所述补充时域数据中的所述第二循环前缀长度和所述保护时间长度,得到PRACH序列对应的时域样本点。

[0086] 具体的,本发明实施例中由于 $k=4$,那么删除所述补充时域数据中的第二循环前缀长度和保护时间长度,得到PRACH序列对应的时域样本点的过程为:将122880时域样本点前端删除12672点,后端删除11904点,就得到了PRACH序列对应的98304点的时域样本点。即PRACH序列对应的时域样本点。

[0087] 本发明实施例通过对时域数据补零再去循环前缀的方法来恢复PRACH format 0格式数据,实现不同信道在帧格式不同的情况下的时域样本点恢复,既避免引入多余的干扰数据,又降低处理复杂度。

[0088] 在本发明实施例的其他方面,请参照图5,步骤500、所述基于PRACH format 0帧结构,删除所述补充时域数据中的第二循环前缀长度和保护时间长度,得到PRACH序列对应的时域样本点之后,还包括:

[0089] 步骤600、基于所述PRACH序列对应的时域样本点进行降采样处理。

[0090] 由于物理层收到的频域数据是RRU统一按照122.88M采样率处理的,没有对PRACH时域数据单独做处理。但PRACH format0和PUSCH帧格式又不同,为了恢复PRACH原始时域数据,本发明实施例通过对时域数据补零再去循环前缀的方法来恢复PRACH format0格式数据,便于基站基于所述PRACH序列对应的时域样本点进行降采样处理。为后续频谱搬移、降采样等提供了前提条件。

[0091] 下面对本发明提供的恢复时域数据的装置进行描述,下文描述的恢复时域数据的装置与上文描述的恢复时域数据的方法可相互对应参照。

[0092] 请参照图6,本发明还提供一种恢复时域数据的装置,包括:

[0093] 接收模块201,用于接收频域数据;

[0094] 傅里叶逆变换模块202,用于基于频域数据进行傅里叶逆变换转换成时域数据;

[0095] 补零模块203,用于基于PUSCH帧结构,将所述时域数据在时域删除的第一循环前缀长度通过补零的方式进行补充,得到补充时域数据;

[0096] 删除模块204,用于基于PRACH format 0帧结构,删除所述补充时域数据中的第二循环前缀长度和保护时间长度,得到PRACH序列对应的时域样本点。

[0097] 本发明实施例的恢复时域数据的装置通过基于PUSCH帧结构,将所述时域数据在时域删除的第一循环前缀长度通过补零的方式进行补充,得到补充时域数据;以及基于PRACH format 0帧结构,删除所述补充时域数据中的第二循环前缀长度和保护时间长度,得到PRACH序列对应的时域样本点。从而本发明实施例通过对时域数据补零再去循环前缀的方法来恢复PRACH format 0格式数据,实现不同信道在帧格式不同的情况下的时域样本点恢复,既避免引入多余的干扰数据,又降低处理复杂度。

[0098] 根据本发明提供的一种恢复时域数据的装置,所述补零模块包括:

[0099] 时间长度确定模块,用于确定PRACH format 0帧结构的时间长度;

[0100] 最终补零模块,用于基于PUSCH帧结构,将所述时域数据在时域删除的第一循环前

缀长度通过补零的方式进行补充,得到所述时间长度的补充时域数据。

[0101] 根据本发明提供一种恢复时域数据的装置,所述最终补零模块具体用于:根据OFDM符号与第一循环前缀长度计算公式,保持每个OFDM符号对应的数值不动,在每个OFDM符号之前把第一循环前缀长度对应的采样点通过补零的方式进行补充,直至得到所述时间长度的补充时域数据。

[0102] 根据本发明提供一种恢复时域数据的装置,所述删除模块包括:

[0103] 第二循环前缀长度和保护时间长度确定模块,用于基于基站采样率和PRACH format 0帧结构,确定第二循环前缀长度和保护时间长度;

[0104] 最终删除模块删除,用于删除所述补充时域数据中的所述第二循环前缀长度和所述保护时间长度,得到PRACH序列对应的时域样本点。

[0105] 根据本发明提供一种恢复时域数据的装置,所述恢复时域数据的装置还包括:

[0106] 预处理模块,用于将多个资源元素数据经IFFT_shift操作放置所述频域数据两端。

[0107] 根据本发明提供一种恢复时域数据的方法,所述恢复时域数据的装置还包括:

[0108] 降采样处理模块,用于基于所述PRACH序列对应的时域样本点进行降采样处理。

[0109] 图7示例了一种电子设备的实体结构示意图,如图7所示,该电子设备可以包括:处理器(processor)710、通信接口(Communications Interface)720、存储器(memory)730和通信总线740,其中,处理器710,通信接口720,存储器730通过通信总线740完成相互间的通信。处理器710可以调用存储器730中的逻辑指令,以执行恢复时域数据的方法,该方法包括:接收频域数据;基于频域数据进行傅里叶逆变换转换成时域数据;基于PUSCH帧结构,将所述时域数据在时域删除的第一循环前缀长度通过补零的方式进行补充,得到补充时域数据;基于PRACH format 0帧结构,删除所述补充时域数据中的第二循环前缀长度和保护时间长度,得到PRACH序列对应的时域样本点。

[0110] 此外,上述的存储器730中的逻辑指令可以通过软件功能单元的形式实现并作为独立的产品销售或使用,可以存储在一个计算机可读取存储介质中。基于这样的理解,本发明的技术方案本质上或者说对现有技术做出贡献的部分或者该技术方案的部分可以以软件产品的形式体现出来,该计算机软件产品存储在一个存储介质中,包括若干指令用以使得一台计算机设备(可以是个人计算机,服务器,或者网络设备等)执行本发明各个实施例所述方法的全部或部分步骤。而前述的存储介质包括:U盘、移动硬盘、只读存储器(ROM, Read-Only Memory)、随机存取存储器(RAM, Random Access Memory)、磁碟或者光盘等各种可以存储程序代码的介质。

[0111] 另一方面,本发明还提供一种计算机程序产品,所述计算机程序产品包括计算机程序,计算机程序可存储在非暂态计算机可读存储介质上,所述计算机程序被处理器执行时,计算机能够执行上述各方法所提供的恢复时域数据的方法,该方法包括:接收频域数据;基于频域数据进行傅里叶逆变换转换成时域数据;基于PUSCH帧结构,将所述时域数据在时域删除的第一循环前缀长度通过补零的方式进行补充,得到补充时域数据;基于PRACH format 0帧结构,删除所述补充时域数据中的第二循环前缀长度和保护时间长度,得到PRACH序列对应的时域样本点。

[0112] 又一方面,本发明还提供一种非暂态计算机可读存储介质,其上存储有计算机程

序,该计算机程序被处理器执行时实现以执行上述各方法提供的恢复时域数据的方法,该方法包括:接收频域数据;基于频域数据进行傅里叶逆变换转换成时域数据;基于PUSCH帧结构,将所述时域数据在时域删除的第一循环前缀长度通过补零的方式进行补充,得到补充时域数据;基于PRACH format 0帧结构,删除所述补充时域数据中的第二循环前缀长度和保护时间长度,得到PRACH序列对应的时域样本点。

[0113] 以上所描述的装置实施例仅仅是示意性的,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。本领域普通技术人员在不付出创造性的劳动的情况下,即可以理解并实施。

[0114] 通过以上的实施方式的描述,本领域的技术人员可以清楚地了解到各实施方式可借助软件加必需的通用硬件平台的方式来实现,当然也可以通过硬件。基于这样的理解,上述技术方案本质上或者说对现有技术做出贡献的部分可以以软件产品的形式体现出来,该计算机软件产品可以存储在计算机可读存储介质中,如ROM/RAM、磁碟、光盘等,包括若干指令用以使得一台计算机设备(可以是个人计算机,服务器,或者网络设备等)执行各个实施例或者实施例的某些部分所述的方法。

[0115] 最后应说明的是:以上实施例仅用以说明本发明的技术方案,而非对其限制;尽管参照前述实施例对本发明进行了详细的说明,本领域的普通技术人员应当理解:其依然可以对前述各实施例所记载的技术方案进行修改,或者对其中部分技术特征进行等同替换;而这些修改或者替换,并不使相应技术方案的本质脱离本发明各实施例技术方案的精神和范围。

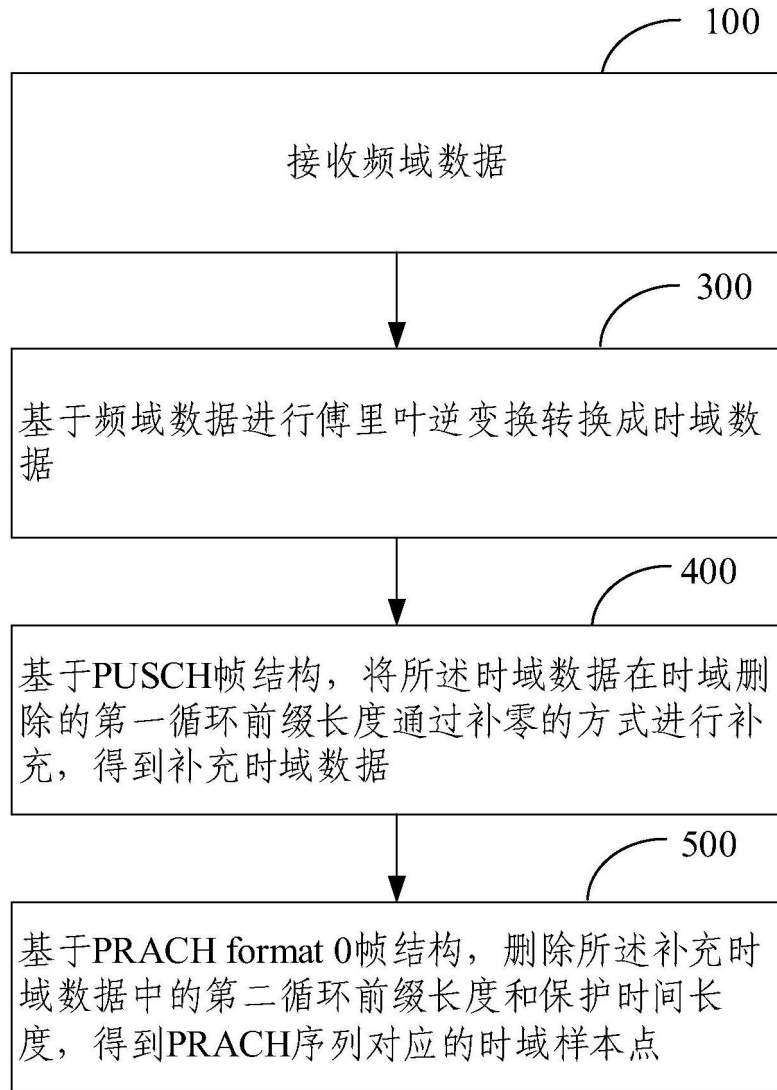


图1

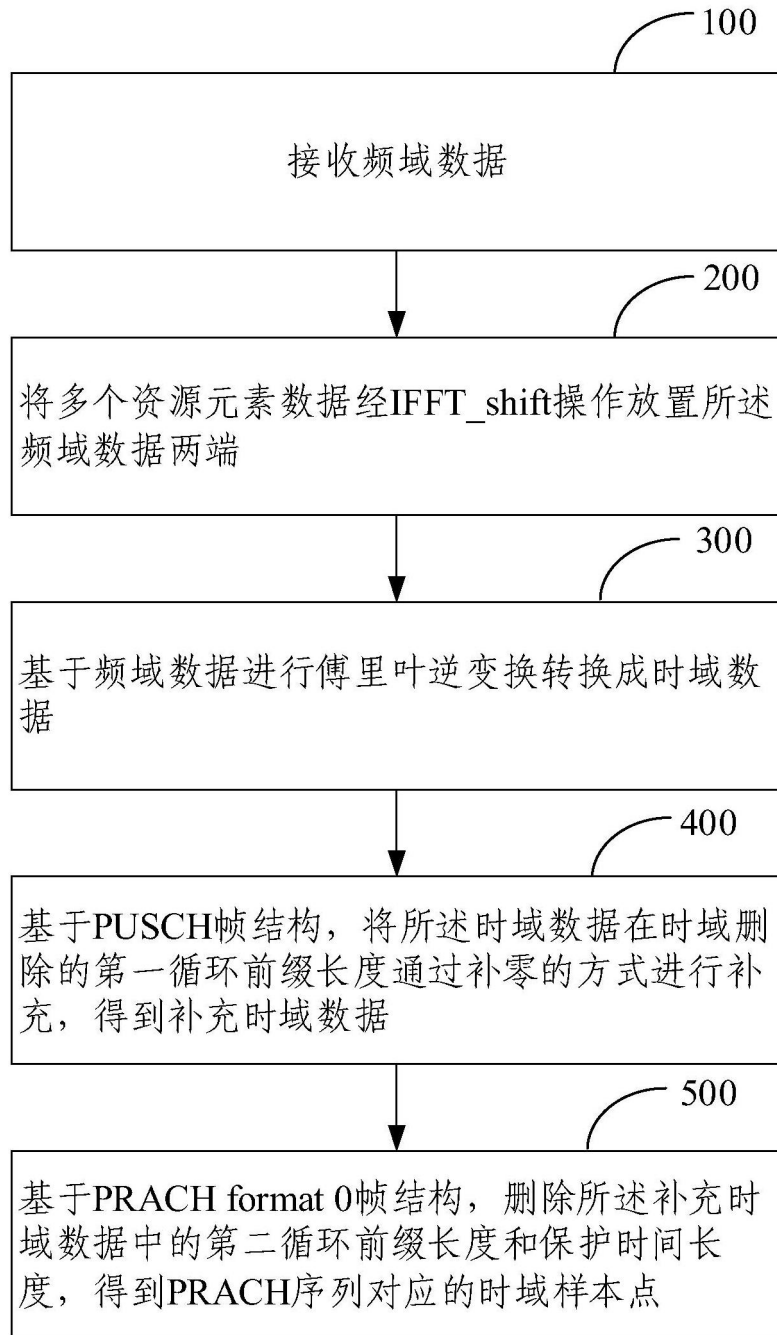


图2

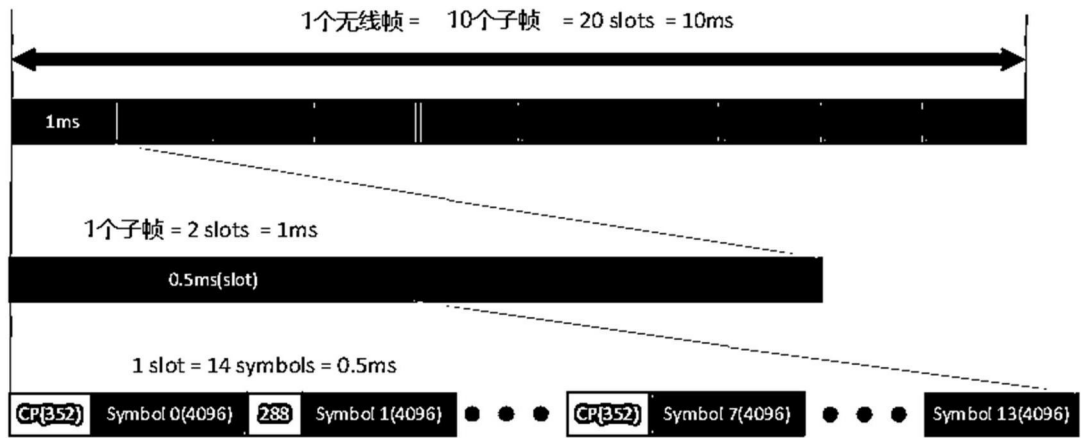


图3



图4

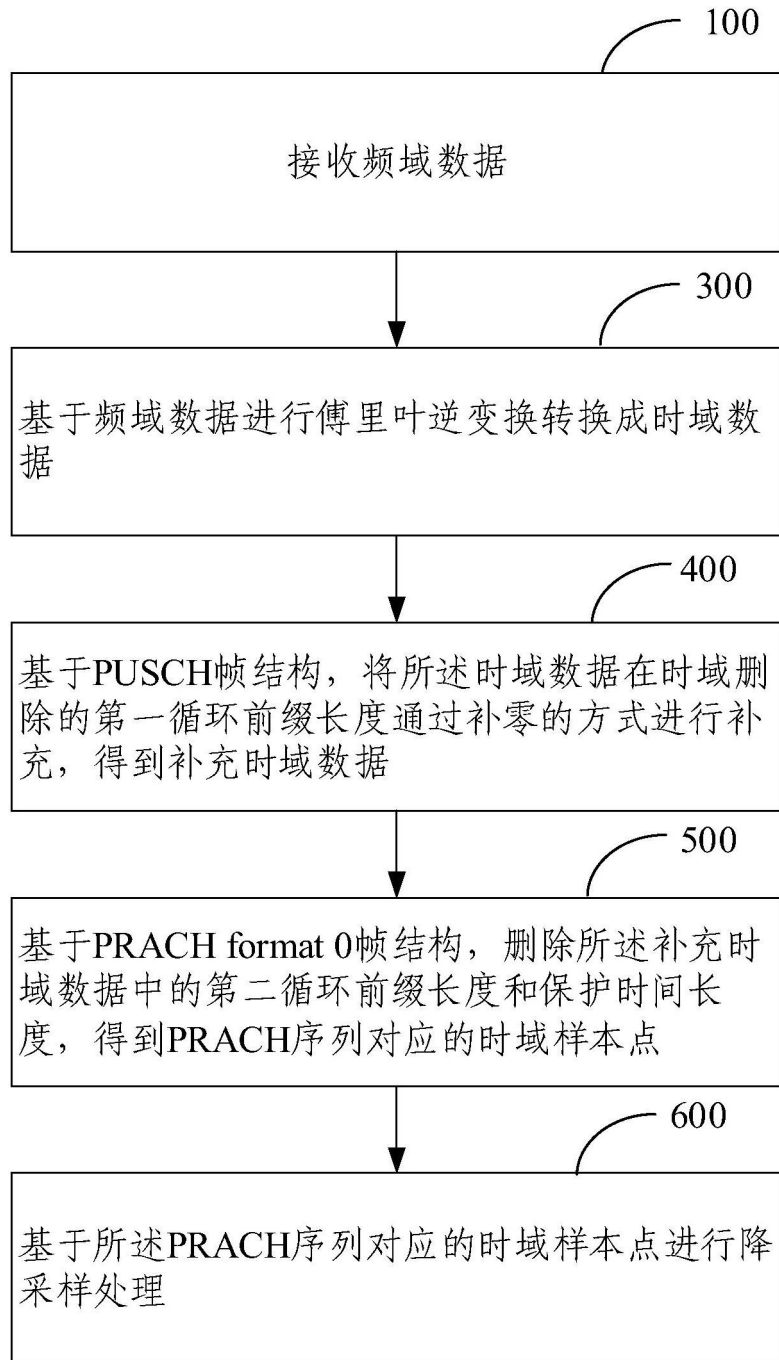


图5

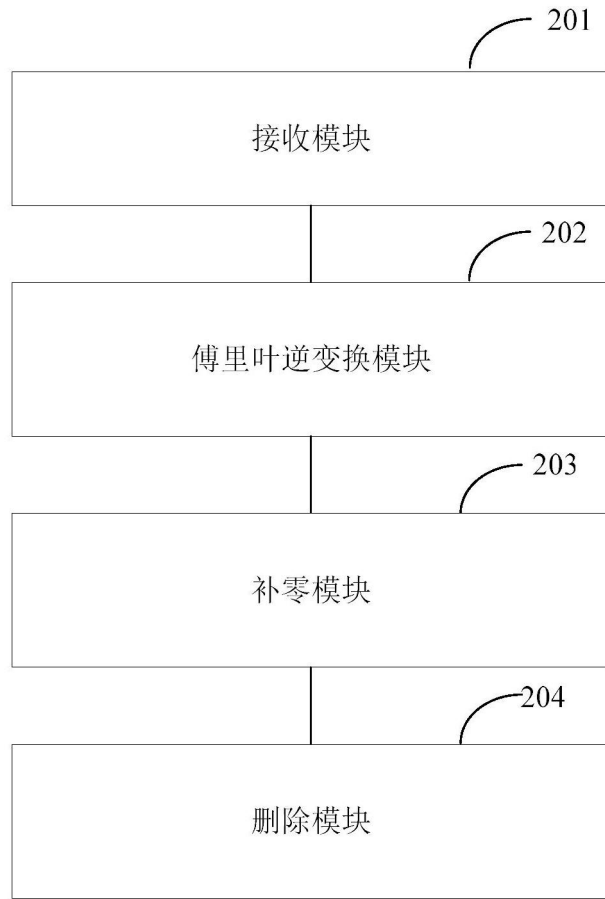


图6

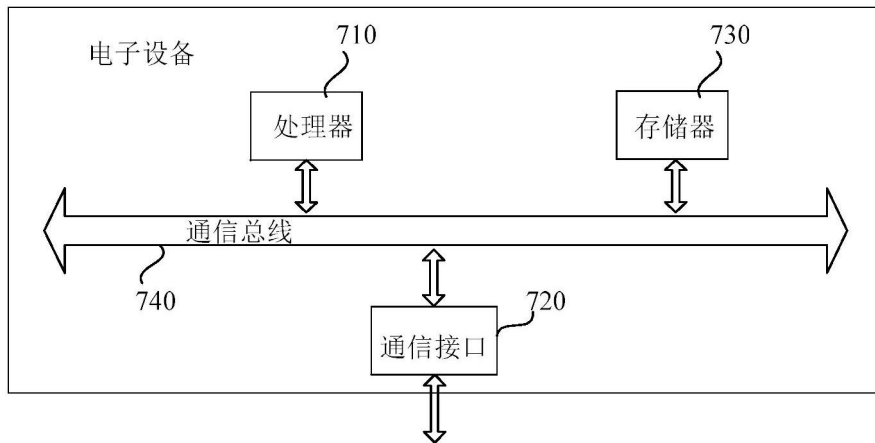


图7