



(12) 发明专利

(10) 授权公告号 CN 110020859 B

(45) 授权公告日 2022.03.11

(21) 申请号 201910242254.7

(22) 申请日 2019.03.28

(65) 同一申请的已公布的文献号  
申请公布号 CN 110020859 A

(43) 申请公布日 2019.07.16

(73) 专利权人 杭州秘猿科技有限公司  
地址 310013 浙江省杭州市西湖区文三路  
478号华星时代广场A座1301

(72) 发明人 谢晗剑 张亚宁

(74) 专利代理机构 北京德崇智捷知识产权代理  
有限公司 11467

代理人 董柏雷

(51) Int.Cl.

G06Q 20/38 (2012.01)

H04L 12/18 (2006.01)

(56) 对比文件

CN 108648078 A, 2018.10.12

CN 108399572 A, 2018.08.14

审查员 罗玲莉

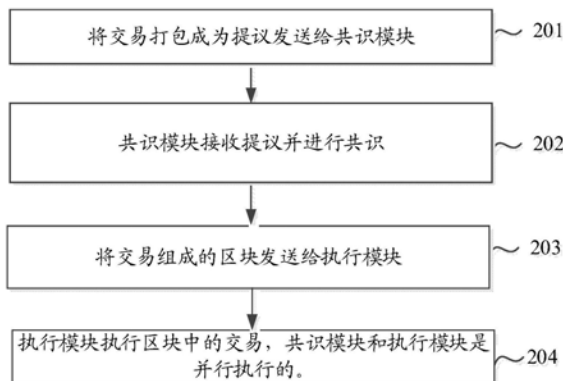
权利要求书2页 说明书7页 附图2页

(54) 发明名称

一种并行执行的区块链共识方法、装置及电子设备

(57) 摘要

本申请实施例公开一种并行执行的区块链共识方法、装置及电子设备,包括:将交易打包成为提议发送给共识模块;共识模块接收所述提议并进行共识;将所述交易组成的区块发送给执行模块;所述执行模块执行所述区块中的交易;其中,所述共识模块和所述执行模块是并行执行的。本申请实施例提供的方法,减少了执行模块闲置的时间,提高了系统的工作效率。



1. 一种并行执行的区块链共识方法,其特征在于,所述方法包括:
  - 将交易打包成为提议发送给共识模块;
  - 共识模块接收所述提议并进行共识;
  - 将所述交易组成的区块发送给执行模块;
  - 所述执行模块执行所述区块中的交易;
  - 其中,所述共识模块和所述执行模块是并行执行的;
  - 所述共识模块完成所述共识之后,获得共识结果;
  - 所述共识模块将所述共识结果发送给所述执行模块;
  - 在所述共识模块将所述共识结果发送给所述执行模块之后,所述方法还包括:
    - 所述执行模块执行完毕所述区块中的交易,获得第一执行结果;
    - 当所述共识结果和所述交易组成的区块一致,所述执行模块将所述第一执行结果上链;
  - 当所述共识结果和所述交易组成的区块不一致,所述执行模块执行所述共识结果中的交易,获得第二执行结果,并将所述第二执行结果上链;
  - 在所述共识模块将所述共识结果发送给所述执行模块之后,所述方法还包括:
    - 所述执行模块未执行完毕所述区块中的交易;
    - 当所述共识结果和所述交易组成的区块一致,所述执行模块执行完毕所述区块中的交易,获得第一执行结果,并将所述第一执行结果上链;
    - 当所述共识结果和所述交易组成的区块不一致,所述执行模块终止执行所述区块中的交易,以及执行所述共识结果中的交易,获得第二执行结果,并将所述第二执行结果上链。
2. 一种并行执行的区块链共识装置,其特征在于,所述装置包括:
  - 打包模块,用于将交易打包成为提议发送给共识模块;
  - 所述共识模块,用于接收所述提议进行共识;
  - 执行模块,用于接收所述交易组成的区块,并执行所述区块中的交易;
  - 其中,所述共识模块和所述执行模块是并行执行的;
  - 其中,所述共识模块,还用于完成所述共识之后,获得共识结果;将所述共识结果发送给所述执行模块;
  - 所述执行模块,还用于:
    - 执行完毕所述区块中的交易,获得第一执行结果;
    - 当所述共识结果和所述交易组成的区块一致,将所述第一执行结果上链;
    - 当所述共识结果和所述交易组成的区块不一致,执行所述共识结果中的交易,获得第二执行结果,并将所述第二执行结果上链;
  - 所述执行模块,还用于:
    - 未执行完毕所述区块中的交易;
    - 当所述共识结果和所述交易组成的区块一致,执行完毕所述区块中的交易,获得第一执行结果,并将所述第一执行结果上链;
    - 当所述共识结果和所述交易组成的区块不一致,终止执行所述区块中的交易,以及执行所述共识结果中的交易,获得第二执行结果,并将所述第二执行结果上链。
3. 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,所述计算机程序被所述处理器执行如权利要求1所述的方法。

4.一种计算机存储介质,包括指令,用于使计算机执行如权利要求1所述的方法。

## 一种并行执行的区块链共识方法、装置及电子设备

### 技术领域

[0001] 本申请涉及网络技术领域,尤其涉及一种并行执行的区块链共识方法、装置及电子设备。

### 背景技术

[0002] 区块链是一种新的分布式技术,是分布式数据存储、点对点传输、共识机制、加密算法等计算机技术的新型应用模式。采用区块链技术架构的网络可视为区块链网络,该区块链网络中包含多个区块链节点,任一区块链节点可对应至少一个区块链,任一区块链可包含至少一个区块。

[0003] 在区块链技术中,共识算法是区块链网络中实现不同区块链节点之间建立信任、获取权益的重要方法。目前采用拜占庭容错(Byzantine Fault Tolerance,BFT)这类共识算法的区块链中,区块链底层平台先对交易完成共识,共识完成后才会去完成交易的执行计算。当一些交易被打包进行提议之后,这些提议有很大的可能会被成功共识,然而在共识的过程中,执行模块是处于闲置状态的,这降低了整个系统的工作效率。

[0004] 由此,亟需找到一种区块链中提高系统工作效率的方案,以克服上述问题。

### 发明内容

[0005] 本申请实施例提供一种并行执行的区块链共识方法、装置及电子设备,使共识模块与执行模块并行工作,从而提高系统的工作效率。

[0006] 为了解决上述技术问题,本申请实施例采用下述技术方案:

[0007] 第一方面,提供了一种并行执行的区块链共识方法,包括:

[0008] 将交易打包成为提议发送给共识模块;

[0009] 共识模块接收所述提议并进行共识;

[0010] 将所述交易组成的区块发送给执行模块;

[0011] 所述执行模块执行所述区块中的交易;

[0012] 其中,所述共识模块和所述执行模块是并行执行的。

[0013] 可选的,所述共识模块完成所述共识之后,获得共识结果;所述共识模块将所述共识结果发送给所述执行模块。

[0014] 在一种可能的实现方式中,在所述共识模块将所述共识结果发送给所述执行模块之后,所述方法还包括:

[0015] 所述执行模块执行完毕所述区块中的交易,获得第一执行结果;

[0016] 当所述共识结果和所述交易组成的区块一致,所述执行模块将所述第一执行结果上链;

[0017] 当所述共识结果和所述交易组成的区块不一致,所述执行模块执行所述共识结果中的交易,获得第二执行结果,并将所述第二执行结果上链。

[0018] 在一种可能的实现方式中,在所述共识模块将所述共识结果发送给所述执行模块

之后,所述方法还包括:

[0019] 所述执行模块未执行完毕所述区块中的交易;

[0020] 当所述共识结果和所述交易组成的区块一致,所述执行模块执行完毕所述区块中的交易,获得第一执行结果,并将所述第一执行结果上链;

[0021] 当所述共识结果和所述交易组成的区块不一致,所述执行模块终止执行所述区块中的交易,以及执行所述共识结果中的交易,获得第二执行结果,并将所述第二执行结果上链。

[0022] 第二方面,提供了一种并行执行的区块链共识装置,包括:

[0023] 打包模块,用于将交易打包成为提议发送给共识模块;

[0024] 所述共识模块,用于接收所述提议进行共识;

[0025] 执行模块,用于接收所述交易组成的区块,并执行所述区块中的交易;

[0026] 其中,所述共识模块和所述执行模块是并行执行的。

[0027] 可选的,所述共识模块,还用于完成所述共识之后,获得共识结果;将所述共识结果发送给所述执行模块。

[0028] 在一种可能的实现方式中,所述执行模块,还用于:

[0029] 执行完毕所述区块中的交易,获得第一执行结果;

[0030] 当所述共识结果和所述交易组成的区块一致,将所述第一执行结果上链;

[0031] 当所述共识结果和所述交易组成的区块不一致,执行所述共识结果中的交易,获得第二执行结果,并将所述第二执行结果上链。

[0032] 在一种可能的实现方式中,所述执行模块,还用于:

[0033] 未执行完毕所述区块中的交易;

[0034] 当所述共识结果和所述交易组成的区块一致,执行完毕所述区块中的交易,获得第一执行结果,并将所述第一执行结果上链;

[0035] 当所述共识结果和所述交易组成的区块不一致,终止执行所述区块中的交易,以及执行所述共识结果中的交易,获得第二执行结果,并将所述第二执行结果上链。

[0036] 第三方面,本申请提供一种电子设备,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,所述计算机程序被所述处理器执行第一方面所述的方法。

[0037] 第四方面,本申请提供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,使得计算机执行第一方面所述的方法。

[0038] 第五方面,本申请提供一种芯片,所述芯片与存储器相连,用于读取并执行所述存储器中存储的软件程序,以实现第一方面所述的方法。

[0039] 本申请实施例采用的上述至少一个技术方案能够达到以下有益效果:

[0040] 本申请实施例中提供的技术方案,使共识模块和执行模块并行执行各自的操作,减少了执行模块闲置等待的时间,极大地提高了系统的工作效率。

## 附图说明

[0041] 为了更清楚地说明本说明书实施例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简单地介绍,显而易见地,下面描述中的附图仅仅是

本说明书实施例中记载的一些实施例,对于本领域普通技术人员来讲,在不付出创造性劳动性的前提下,还可以根据这些附图获得其他的附图。

[0042] 图1为本申请实施例提供的一种区块链共识算法的示意图;

[0043] 图2为本申请实施例提供的一种并行执行的区块链共识方法流程图;

[0044] 图3为本申请实施例提供的一种电子设备的结构示意图;

[0045] 图4为本申请实施例提供的一种并行执行的区块链共识装置结构示意图。

### 具体实施方式

[0046] 为使本说明书实施例的目的、技术方案和优点更加清楚,下面将结合本说明书具体实施例及相应的附图对本说明书实施例的技术方案进行清楚、完整地描述。显然,所描述的实施例仅是本说明书一部分实施例,而不是全部的实施例。基于本说明书中的实施例,本领域普通技术人员在没有做出创造性劳动前提下所获得的所有其他实施例,都属于本说明书实施例保护的范围。

[0047] 以下结合附图,详细说明本说明书各实施例提供的技术方案。

[0048] 区块链本质上是一个去中心化的数据库,是一连串使用密码学方法产生相关联的数据块,每一个数据块中包含了一段时间内全网交易的信息。不同的区块链,有着不同的内涵和功能。区块链中最重要的就是共识算法,共识算法中包括分布式一致性算法,分布式一致性算法一般可以分为BFT算法和非BFT算法。目前主流的拜占庭容错算法包括传统的实用拜占庭容错(Practical Byzantine Fault Tolerance, PBFT)算法、tendermint算法等。

[0049] 基于拜占庭将军的问题,一致性的确保主要分为三个阶段:预准备(pre-prepare)、准备(prepare)和确认(commit)。其示意图如图1所示。

[0050] 在图1中,C为发送请求端节点,0、1、2、3为服务端节点,3为宕机的服务端节点,具体步骤如下:

[0051] 请求(Request):请求端节点C发送请求到任意一节点,在本申请的实施例中为0。

[0052] 预准备(Pre-Prepare):服务端节点0收到节点C的请求后进行广播,扩散至节点1、2、3。

[0053] 准备(Prepare):节点1、2、3收到节点0的广播后记录并再次广播。节点1向节点0、2、3广播,节点2向节点0、1、3广播,节点3因为宕机无法广播。

[0054] 确认(Commit):节点0、1、2、3在准备阶段,若收到超过一定数量的相同请求,则进入确认阶段,广播确认请求。

[0055] 答复(Reply):节点0、1、2、3在确认阶段,若收到超过一定数量的相同请求,则对节点C进行反馈。

[0056] 由此可以看出,PBFT共识算法其实是一种状态机副本复制算法,即服务作为状态机进行建模,状态机在分布式系统的不同节点进行副本复制。每个状态机的副本都保存了服务的状态,同时也实现了服务的操作。

[0057] 目前采用BFT这类共识算法的区块链中,区块链底层平台先对交易完成共识,共识完成后才会去完成交易的执行计算。当一些交易被打包进行提议之后,这些提议有很大的可能会被成功共识,然而在共识的过程中,执行模块是处于闲置状态的,这降低了整个系统的工作效率。

[0058] 鉴于上述问题,本申请实施例提供了一种并行执行的区块链共识方法、装置及电子设备,解决了现有技术中在共识的过程中由于执行模块的闲置造成的系统工作效率降低的问题。

[0059] 实施例一

[0060] 图2为本申请实施例提供的一种并行执行的区块链共识方法的流程图,包括如下步骤:

[0061] S201、将交易打包成为提议发送给共识模块。

[0062] S202、共识模块接收所述提议并进行共识。

[0063] 在BFT共识算法的实际应用中,当一些交易被打包进行提议之后,这些提议有很大的可能会被成功共识,因此,在提议节点中,交易被打包后发送给共识模块,共识模块在广播交易区接收该提议并进行共识。

[0064] S203、将所述交易组成的区块发送给执行模块。

[0065] 在共识模块进行共识的同时,步骤201中的交易组成的区块也被发送给执行模块。

[0066] S204、所述执行模块执行所述区块中的交易。

[0067] 在共识模块进行共识的同时,执行模块并行地执行区块中的交易。

[0068] 共识模块完成共识之后,获得共识结果,并将该共识结果发送给执行模块。此时,执行模块要根据不同的情况执行不同的操作:

[0069] 1) 如果执行模块已经执行完毕所述区块中的交易,获得一个执行结果,则

[0070] 当所述共识结果和步骤203中交易组成的区块一致时,执行模块就将该执行结果上链;

[0071] 当所述共识结果和步骤203中交易组成区块不一致时,执行模块执行所述共识结果中的交易,获得另一个执行结果,并将该执行结果上链。

[0072] 2) 如果执行模块还没有执行完毕所述区块中的交易,则

[0073] 当所述共识结果和步骤203中交易组成的区块一致时,所述执行模块就继续执行步骤203中收到的区块中的交易,执行完毕后,获得一个执行结果,并将该执行结果上链;

[0074] 当所述共识结果和步骤203中交易组成的区块不一致时,所述执行模块就终止执行步骤203中区块中的交易,然后执行所述共识结果中的交易,获得另一个执行结果,并将该执行结果上链。

[0075] 在BFT共识算法的实际应用中,当一些交易被打包进行提议之后,这些提议有很大的可能会被成功共识,因此本申请实施例中提供的方法中,并行执行提议的执行模块有很大的可能执行的就是最终的共识结果。比起共识模块完成共识,获得共识结果之后,执行模块再接收共识结果,然后去执行共识结果的顺序执行方法,本申请实施例提供的方法,使共识模块和执行模块并行执行各自的操作,减少了执行模块闲置等待的时间,极大地提高了系统的工作效率。

[0076] 实施例二

[0077] 下面参照图3详细介绍本说明书实施例的电子设备的。请参考图3,在硬件层面,该电子设备包括处理器,可选地还包括内部总线、网络接口、存储器。其中,存储器可能包含内存,例如高速随机存取存储器(Random-Access Memory, RAM),也可能还包括非易失性存储器(Non-Volatile Memory),例如至少1个磁盘存储器等。当然,该电子设备还可能包括其他

业务所需要的硬件。

[0078] 处理器、网络接口和存储器可以通过内部总线相互连接,该内部总线可以是工业标准体系结构(Industry Standard Architecture,ISA)总线、外设部件互连标准(Peripheral Component Interconnect,PCI)总线或扩展工业标准结构(Extended Industry Standard Architecture,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3中仅用一个双向箭头表示,但并不表示仅有一根总线或一种类型的总线。

[0079] 存储器,用于存放程序。具体地,程序可以包括程序代码,所述程序代码包括计算机操作指令。存储器可以包括内存和非易失性存储器,并向处理器提供指令和数据。

[0080] 处理器从非易失性存储器中读取对应的计算机程序到内存中然后运行,在逻辑层面上形成内容推荐装置。处理器,执行存储器所存放的程序,并具体用于执行前文所述服务器作为执行主体时所执行的方法操作。

[0081] 上述如本说明书实施例图2所示实施例揭示的方法可以应用于处理器中,或者由处理器实现。处理器可能是一种集成电路芯片,具有信号的处理能力。在实现过程中,上述方法的各步骤可以通过处理器中的硬件的集成逻辑电路或者软件形式的指令完成。上述的处理器可以是通用处理器,包括中央处理器(Central Processing Unit,CPU)、网络处理器(Network Processor,NP)等;还可以是数字信号处理器(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、分立门或者晶体管逻辑器件、分立硬件组件。可以实现或者执行本说明书实施例中的公开的各方法、步骤及逻辑框图。通用处理器可以是微处理器或者该处理器也可以是任何常规的处理器等。结合本说明书实施例所公开的方法的步骤可以直接体现为硬件译码处理器执行完成,或者用译码处理器中的硬件及软件模块组合执行完成。软件模块可以位于随机存储器,闪存、只读存储器,可编程只读存储器或者电可擦写可编程存储器、寄存器等本领域成熟的存储介质中。该存储介质位于存储器,处理器读取存储器中的信息,结合其硬件完成上述方法的步骤。

[0082] 该电子设备还可执行图2的方法,并实现区块链并行交易装置在图2所示实施例的功能,本说明书实施例在此不再赘述。

[0083] 当然,除了软件实现方式之外,本说明书实施例的电子设备并不排除其他实现方式,比如逻辑器件抑或软硬件结合的方式等等,也就是说以下处理流程的执行主体并不限定于各个逻辑单元,也可以是硬件或逻辑器件。

[0084] 实施例三

[0085] 本说明书实施例还提供一种计算机可读存储介质,所述计算机可读存储介质存储一个或多个程序,所述一个或多个程序当被包括多个应用程序的电子设备执行时,使得所述电子设备执行实施例一所述的方法。在此不再赘述。

[0086] 其中,所述的计算机可读存储介质,如只读存储器(Read-Only Memory,ROM)、随机存取存储器(Random Access Memory,RAM)、磁碟或者光盘等。

[0087] 实施例四

[0088] 参照图4所示,为本说明书实施例提供的区块链并行交易的装置的结构示意图,该



装置主要包括打包模块401、共识模块402和执行模块403。其中，

[0089] 打包模块401,用于将交易打包成为提议发送给共识模块402。在BFT共识算法的实际应用中,当一些交易被打包进行提议之后,这些提议有很大的可能会被成功共识。

[0090] 共识模块402,用于在广播交易区接收上述提议并进行共识。

[0091] 执行模块403,用于接收所述交易组成的区块,并执行所述区块中的交易。

[0092] 在共识模块402进行共识的同时,打包模块打包的交易组成的区块也被发送给执行模块。

[0093] 在共识模块进行共识的同时,执行模块并行地执行区块中的交易。

[0094] 共识模块完成共识之后,获得共识结果,并将该共识结果发送给执行模块。此时,执行模块要根据不同的情况执行不同的操作:

[0095] 1) 如果执行模块已经执行完毕所述区块中的交易,获得一个执行结果,则

[0096] 当所述共识结果和交易组成的区块一致时,执行模块就将该执行结果上链;

[0097] 当所述共识结果和交易组成区块不一致时,执行模块执行所述共识结果中的交易,获得另一个执行结果,并将该执行结果上链。

[0098] 2) 如果执行模块还没有执行完毕所述区块中的交易,则

[0099] 当所述共识结果和交易组成的区块一致时,所述执行模块就继续执行收到的区块中的交易,执行完毕后,获得一个执行结果,并将该执行结果上链;

[0100] 当所述共识结果和交易组成的区块不一致时,所述执行模块就终止执行区块中的交易,然后执行所述共识结果中的交易,获得另一个执行结果,并将该执行结果上链。

[0101] 在BFT共识算法的实际应用中,当一些交易被打包进行提议之后,这些提议有很大的可能会被成功共识,因此本申请实施例中提供的装置中,并行执行提议的执行模块有很大的可能执行的就是最终的共识结果。比起共识模块完成共识,获得共识结果之后,执行模块再接收共识结果,然后去执行共识结果的顺序执行装置,本申请实施例提供的装置,使共识模块和执行模块并行执行各自的操作,减少了执行模块闲置等待的时间,极大地提高了系统的工作效率。

[0102] 总之,以上所述仅为本说明书实施例的较佳实施例而已,并非用于限定本说明书实施例的保护范围。凡在本说明书实施例的精神和原则之内,所作的任何修改、等同替换、改进等,均应包含在本说明书实施例的保护范围之内。

[0103] 上述实施例阐明的系统、装置、模块或单元,具体可以由计算机芯片或实体实现,或者由具有某种功能的产品来实现。一种典型的实现设备为计算机。具体的,计算机例如可以为个人计算机、膝上型计算机、蜂窝电话、相机电话、智能电话、个人数字助理、媒体播放器、导航设备、电子邮件设备、游戏控制台、平板计算机、可穿戴设备或者这些设备中的任何设备的组合。

[0104] 计算机可读介质包括永久性和非永久性、可移动和非可移动媒体可以由任何方法或技术来实现信息存储。信息可以是计算机可读指令、数据结构、程序的模块或其他数据。计算机的存储介质的例子包括,但不限于相变内存 (PRAM)、静态随机存取存储器 (SRAM)、动态随机存取存储器 (DRAM)、其他类型的随机存取存储器 (RAM)、只读存储器 (ROM)、电可擦除可编程只读存储器 (EEPROM)、快闪记忆体或其他内存技术、只读光盘只读存储器 (CD-ROM)、数字多功能光盘 (DVD) 或其他光学存储、磁盒式磁带,磁带磁磁盘存储或其他磁性存储设备

或任何其他非传输介质,可用于存储可以被计算设备访问的信息。按照本文中的界定,计算机可读介质不包括暂存电脑可读媒体(transitory media),如调制的数据信号和载波。

[0105] 还需要说明的是,术语“包括”、“包含”或者其任何其他变体意在涵盖非排他性的包含,从而使得包括一系列要素的过程、方法、商品或者设备不仅包括那些要素,而且还包括没有明确列出的其他要素,或者是还包括为这种过程、方法、商品或者设备所固有的要素。在没有更多限制的情况下,由语句“包括一个……”限定的要素,并不排除在包括所述要素的过程、方法、商品或者设备中还存在另外的相同要素。

[0106] 本说明书实施例中的各个实施例均采用递进的方式描述,各个实施例之间相同相似的部分互相参见即可,每个实施例重点说明的都是与其他实施例的不同之处。尤其,对于系统实施例而言,由于其基本相似于方法实施例,所以描述的比较简单,相关之处参见方法实施例的部分说明即可。

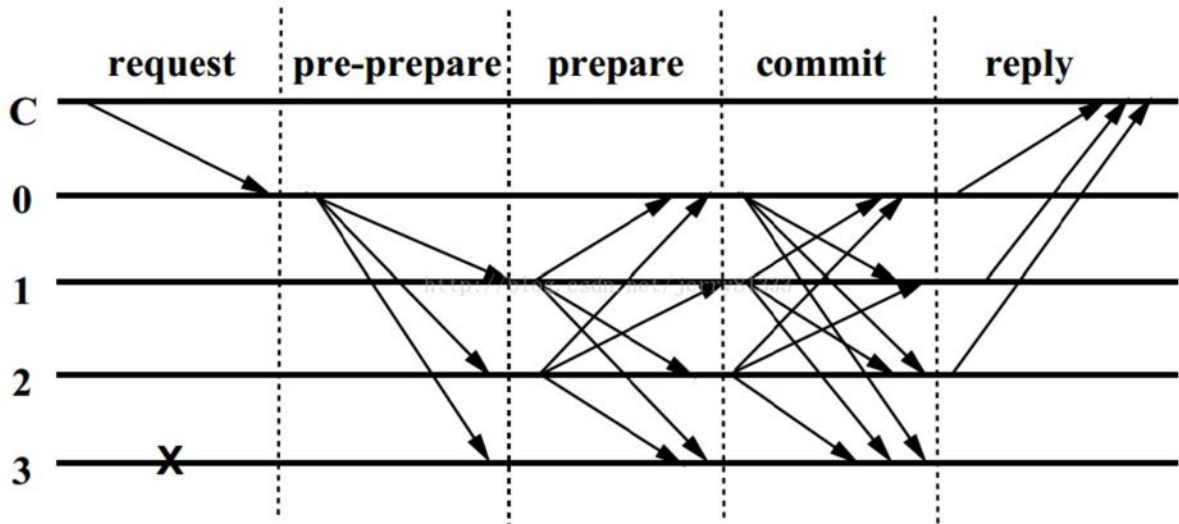


图1

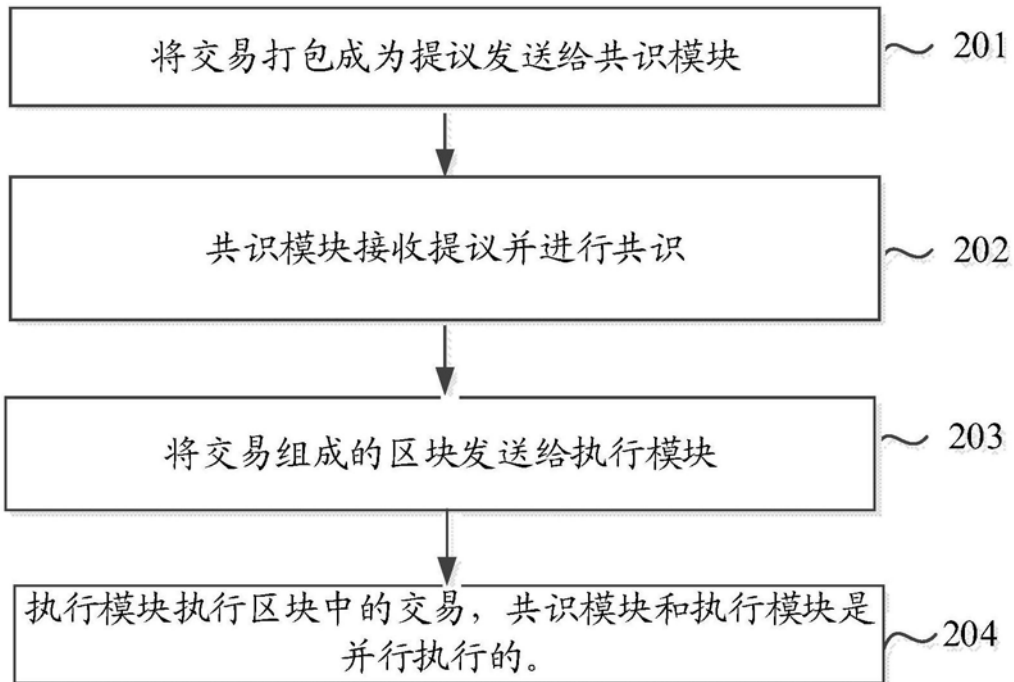


图2

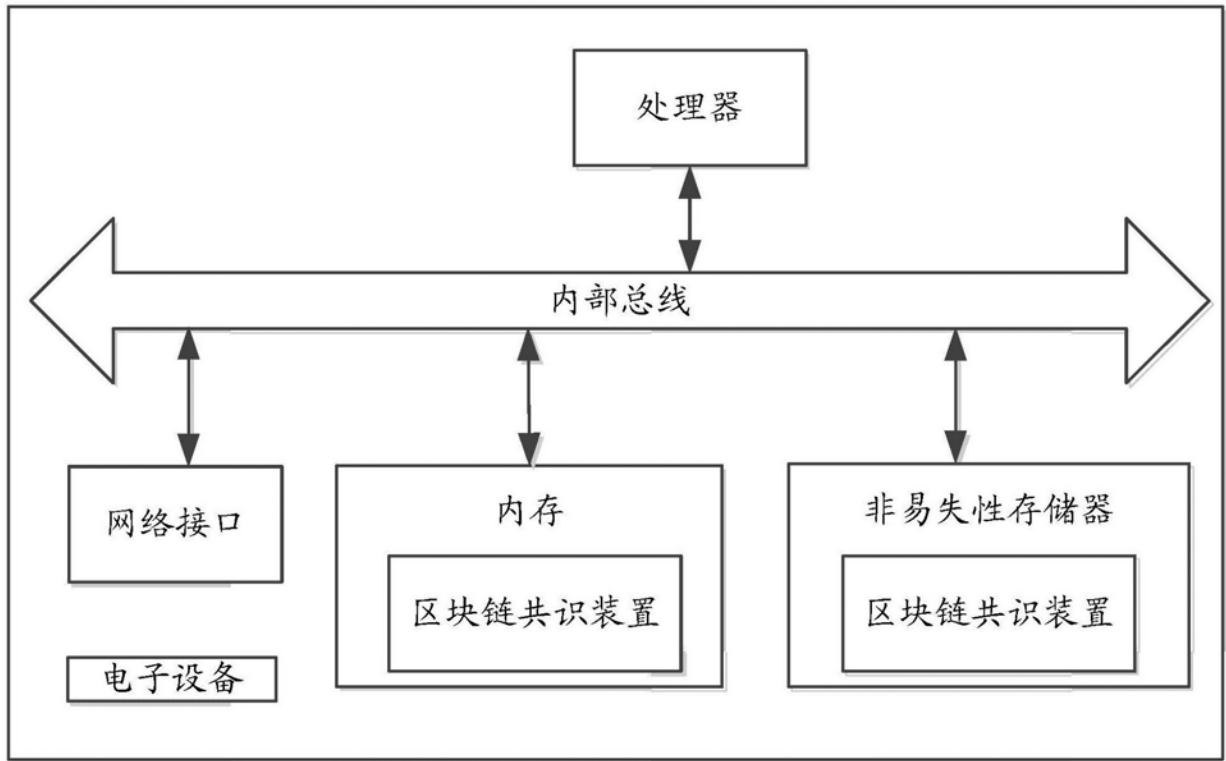


图3

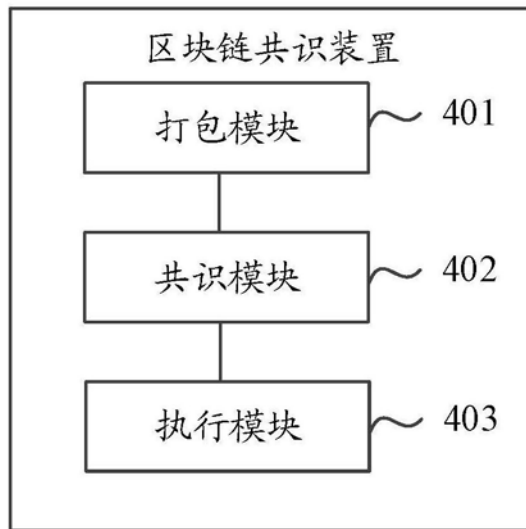


图4