

基于区块链的交易方法、装置和汇出方设备

技术领域

[01] 本说明书实施例涉及计算机技术领域，特别涉及一种基于区块链的交易方法、装置和汇出方设备。

5 背景技术

[02] 区块链技术是一种分布式数据库技术，利用密码学和共识机制保证了数据的不可篡改和不可伪造。随着计算机和互联网技术的发展，区块链技术以其去中心化、公开透明、不可篡改、可信任等优点，备受青睐，在智能合约、证券交易、电子商务、物联网、社交通讯、文件存储、存在性证明、身份验证、股权众筹等众多领域得到广泛应用。

10 [03] 目前，当将区块链技术应用用于交易场景时，由于交易信息需要发送到区块链进行验证、实施和上链，因而交易信息面临泄露给与本次交易无关的第三方的风险。

[04] 业务亟需一种在交易过程中能够实现隐私保护的技术方案。

发明内容

[05] 本说明书实施例的目的是提供一种基于区块链的交易方法、装置和汇出方设备，以
15 在无需与汇入方设备进行交互的条件下实现隐私保护。

[06] 为实现上述目的，本说明书实施例提供一种基于区块链的交易方法，包括：

[07] 根据承诺随机数和交易额，计算交易额承诺；

[08] 根据汇出方账户的同态加密公钥和所述交易额，使用同态加密算法对所述承诺随机数进行加密，得到承诺随机数第一密文；

20 [09] 根据汇入方账户的同态加密公钥和所述交易额，使用同态加密算法对所述承诺随机数进行加密，得到承诺随机数第二密文；

[10] 向所述区块链提交交易数据；所述交易数据包括所述交易额承诺、所述承诺随机数第一密文和所述承诺随机数第二密文；以便于将所述交易额承诺和所述承诺随机数第一密文记入所述汇出方账户，以及将所述交易额承诺和所述承诺随机数第二密文记入所述
25 汇入方账户。

- [11]为实现上述目的，本说明书实施例提供一种基于区块链的交易装置，包括：
- [12]计算单元，用于根据承诺随机数和交易额，计算交易额承诺；
- [13]第一加密单元，用于根据汇出方账户的同态加密公钥和所述交易额，使用同态加密算法对所述承诺随机数进行加密，得到承诺随机数第一密文；
- 5 [14]第二加密单元，用于根据汇入方账户的同态加密公钥和所述交易额，使用同态加密算法对所述承诺随机数进行加密，得到承诺随机数第二密文；
- [15]提交单元，用于向所述区块链提交交易数据；所述交易数据包括所述交易额承诺、所述承诺随机数第一密文和所述承诺随机数第二密文；以便于将所述交易额承诺和所述承诺随机数第一密文记入所述汇出方账户，以及将所述交易额承诺和所述承诺随机数第二密文记入所述汇入方账户。
- 10 [16]为实现上述目的，本说明书实施例提供一种汇出方设备，包括：
- [17]存储器，用于存储计算机指令；
- [18]处理器，用于执行计算机指令以实现本说明书任一实施例所述的方法。
- [19]由以上本说明书实施例提供的技术方案可见，本说明书实施例中，通过承诺机制和同态加密机制，可以得到交易额承诺、承诺随机数第一密文和承诺随机数第二密文；可以将所述交易额承诺、所述承诺随机数第一密文和所述承诺随机数第二密文提交上链；以便于将交易额承诺和承诺随机数第一密文记入汇出方账户，以及将交易额承诺和承诺随机数第二密文记入汇入方账户。这样一方面，通过承诺机制和同态加密机制，可以对交易额等交易信息实现隐私保护；另一方面，通过将交易额承诺、承诺随机数第一密文和承诺随机数第二密文提交上链，可以实现无需汇入方设备参与便可以在区块链上顺利完成交易。因此本实施例可以在非交互条件下实现隐私保护。
- 15
20

附图说明

- [20]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的前提下，还可以根据这些附图获得其他的附图。
- 25
- [21]图 1 为本说明书实施例一种基于区块链实施交易的示意图；

[22]图 2 为本说明书实施例一种基于区块链的交易方法的流程图;

[23]图 3 为本说明书实施例一种基于区块链实施交易的流程图;

[24]图 4 为本说明书实施例一种基于区块链的交易装置的功能结构图;

[25]图 5 为本说明书实施例一种汇出方设备的功能结构图。

5 具体实施方式

[26]下面将结合本说明书实施例中的附图,对本说明书实施例中的技术方案进行清楚、完整地描述,显然,所描述的实施例仅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作出创造性劳动前提下所获得的所有其他实施例,都应当属于本说明书保护的范围。

10 [27]请参阅图 1 和图 2。本说明书实施例提供一种基于区块链的交易方法。

[28]在本实施例中,所述区块链(Block Chain),可以是一种按照时间先后顺序,将多个区块数据以链式结构进行组织,并以密码学算法保证安全、可追溯、且不可篡改的分布式账本。所述区块链可以包括公有区块链、联合区块链(也称为联盟区块链)、和私有区块链等。所述区块链可以基于区块链网络来实现。所述区块链网络可以包括 P2P 网络(peer-to-peer network)等。所述区块链网络可以包括多个区块链节点。各个区块链节点之间共同维护统一的区块链账本。

[29]在本实施例中,所述交易方法可以以汇出方设备为执行主体。所述汇出方设备可以作为一个区块链节点加入所述区块链网络。在所述汇出方设备中可以登录有汇出方账户。所述汇出方账户可以为汇出方在所述区块链中的账户。所述汇出方账户可以具有签名公私钥对、和加密公私钥对。所述签名公私钥对可以包括相关联的签名公钥和签名私钥。所述签名私钥可以用于对待提交给所述区块链的交易数据进行签名。所述签名公钥可以向区块链网络中的其它区块链节点公开,以供其它区块链节点对签名后的交易数据进行验签。所述加密公私钥对可以包括相关联的同态加密公钥和同态加密私钥。所述同态加密公钥可以向区块链网络中的其它区块链节点公开,以供自身和其它区块链节点使用同态加密算法对数据(例如汇出方余额、汇出方随机数和承诺随机数等)进行加密。所述同态加密私钥可以用于对加密后的密文进行解密。所述汇出方账户在所述区块链中还可以登记有汇出方余额承诺和汇出方随机数密文。

25 [30]所述汇出方余额承诺可以由所述汇出方设备基于汇出方随机数和汇出方余额计算得

到。所述汇出方随机数可以为所述汇出方账户对应的用于构建汇出方余额承诺的随机数。所述汇出方余额可以为任意类型的资源的余额。所述资源例如可以包括证券、优惠券、房产、虚拟货币、和资金等。具体地，所述汇出方余额承诺可以基于任意类型的具有同态性的承诺机制 (Commitment Scheme) 来实现，例如 Pedersen 承诺机制等。以 Pedersen 承诺机制为例，所述汇出方余额承诺可以基于公式 $PC(r_A, s_A) = g^{r_A} h^{s_A}$ 计算得到。其中， $PC(r_A, s_A)$ 为汇出方余额承诺； r_A 为汇出方随机数； s_A 为汇出方余额； g 和 h 分别为已知参数。当然，所述汇出方余额承诺也可以由同态加密算法来实现。即将基于同态加密算法对所述汇出方余额进行加密得到的密文作为汇出方余额承诺。关于所述同态加密算法在后续过程中会有详细介绍。这样一方面，通过在区块链内登记汇出方余额承诺，可以避免直接登记汇出方余额，使得汇出方余额得以隐藏和保密。另一方面，鉴于承诺机制可以支持高效的零知识证明 (Zero-Knowledge Proof)，汇出方余额承诺可以使得后续过程中涉及汇出方余额的零知识证明 (例如后续过程中的第三零知识证明) 过程较为高效。

[31] 所述汇出方随机数密文可以由同态加密算法基于所述汇出方余额对所述汇出方随机数进行加密得到。具体地，所述汇出方设备可以根据汇出方账户的同态加密公钥和所述汇出方余额，使用同态加密算法对所述汇出方随机数进行加密，得到汇出方随机数密文。所述同态加密算法例如可以包括 Okamoto-Uchiyama 算法、和 Boneh-Goh-Nissim 算法等。以 Okamoto-Uchiyama 算法为例，所述汇出方随机数密文可以基于公式 $OU_A(r_A) = u1^{r_A} v1^{s_A}$ 计算得到。其中， $OU_A(r_A)$ 为汇出方随机数密文； r_A 为汇出方随机数； $u1$ 和 $v1$ 分别为所述汇出方账户的同态加密公钥的一部分； s_A 为汇出方余额。本实施例中可以将汇出方随机数作为被加密的明文，可以将汇出方余额作为用于同态加密运算的随机数，从而实现对所述汇出方随机数进行加密。这样一方面，通过在区块链内登记汇出方随机数，可以避免在链下本地保管汇出方随机数造成的负担。另一方面，通过在区块链内登记汇出方随机数密文，可以避免直接登记汇出方随机数，使得汇出方随机数得以隐藏和保密。

[32] 需要说明的是，在需要获得汇出方余额时，所述汇出方设备可以使用汇出方账户的同态加密私钥对所述汇出方随机数密文进行解密，得到汇出方随机数；可以根据所述汇出方随机数和所述汇出方余额承诺，计算得到汇出方余额。所述汇出方设备例如可以采用 Pollard 算法中的 Kangaroo 方法，计算得到汇出方余额。当然，此处的 Kangaroo 方法仅为示例，实际上所述汇出方设备还可以采用其它方法计算得到汇出方余额。例如，所

所述汇出方随机数密文可以为 $OU_A(r_A) = u1^{r-A} v1^{s-A}$ ，所述汇出方余额承诺可以为 $PC(r_A, s_A) = g^{r-A} h^{s-A}$ 。那么，所述汇出方设备可以使用汇出方账户的同态加密私钥对所述汇出方随机数密文 $OU_A(r_A)$ 进行解密，得到汇出方随机数 r_A ；可以根据汇出方随机数 r_A 和汇出方余额承诺 $PC(r_A, s_A)$ ，计算得到 h^{s-A} ；可以采用 Pollard 算法中的 Kangaroo 方法，根据 h^{s-A} 计算得到汇出方余额 s_A 。此外，在所述汇出方余额承诺由同态加密算法来实现时，所述汇出方设备还可以使用汇出方账户的同态加密私钥对所述汇出方余额承诺进行解密，得到汇出方余额。具体地，例如，由 Okamoto-Uchiyama 算法实现的汇出方余额承诺可以为 $PC(r_A, s_A) = u1^{s-A} v1^{r-A}$ 。

[33] 在本实施例中，所述交易方法可以实现交易额从汇出方账户转移至汇入方账户。所述汇入方账户可以为汇入方在所述区块链中的账户。所述汇入方账户可以在汇入方设备中登录。所述汇入方设备可以作为一个区块链节点加入所述区块链网络。所述交易额可以由汇出方与汇入方协商约定。与所述汇出方账户相类似，所述汇入方账户可以具有签名公私钥对、和加密公私钥对；所述汇入方账户在所述区块链中还可以登记有汇入方余额承诺和汇入方随机数密文。所述汇入方余额承诺可以由所述汇入方设备基于汇入方随机数和汇入方余额计算得到。所述汇入方随机数密文可以由同态加密算法基于所述汇入方余额对所述汇入方随机数进行加密得到。

[34] 所述交易方法可以包括以下步骤。需要说明的是，虽然本说明书提供了如实施例或流程图所述的方法操作步骤，但基于常规或者无创造性的劳动可以包括更多或者更少的操作步骤。此外实施例中列举的步骤顺序仅仅为众多步骤执行顺序中的一种方式，不代表唯一的执行顺序。在实际中的装置或产品执行时，可以按照实施例或者附图所示的方法顺序执行或者并行执行（例如并行处理器或者多线程处理的环境）。

[35] 步骤 S10：根据承诺随机数和校验金额，计算交易额承诺。

[36] 在本实施例中，所述承诺随机数可以由所述汇出方设备生成。所述交易额可以由汇出方与汇入方协商约定。所述交易额承诺可以基于任意类型的具有同态性的承诺机制（Commitment Scheme）来实现，例如 Pedersen 承诺机制等。以 Pedersen 承诺机制为例，所述交易额承诺可以基于公式 $PC(r, t) = g^r h^t$ 计算得到。其中， $PC(r, t)$ 为交易额承诺； r 为承诺随机数； t 为交易额； g 和 h 分别为已知参数。这样通过交易额承诺可以实现交易额的隐藏和保密。

[37] 步骤 S12：根据汇出方账户的同态加密公钥和所述交易额，使用同态加密算法对所

述承诺随机数进行加密, 得到承诺随机数第一密文。

[38]在本实施例中, 所述同态加密算法例如可以包括 Okamoto-Uchiyama 算法、和 Boneh-Goh-Nissim 算法等。以 Okamoto-Uchiyama 算法为例, 所述承诺随机数第一密文可以基于公式 $OU_A(r) = u1^r \cdot v1^t$ 计算得到。其中, $OU_A(r)$ 为承诺随机数第一密文; r 为承诺随机数; $u1$ 和 $v1$ 分别为所述汇出方账户的同态加密公钥的一部分; t 为交易额。本实施例中可以将承诺随机数作为被加密的明文, 可以将交易额作为用于同态加密运算的随机数, 从而实现对所述承诺随机数进行加密。

[39]步骤 S14: 根据汇入方账户的同态加密公钥和所述交易额, 使用同态加密算法对所述承诺随机数进行加密, 得到承诺随机数第二密文。

[40]在本实施例中, 所述同态加密算法例如可以包括 Okamoto-Uchiyama 算法、和 Boneh-Goh-Nissim 算法等。以 Okamoto-Uchiyama 算法为例, 所述承诺随机数第二密文可以基于公式 $OU_B(r) = u2^r \cdot v2^t$ 计算得到。其中, $OU_B(r)$ 为承诺随机数第二密文; r 为承诺随机数; $u2$ 和 $v2$ 分别为所述汇入方账户的同态加密公钥的一部分; t 为交易额。本实施例中可以将承诺随机数作为被加密的明文, 可以将交易额作为用于同态加密运算的随机数, 从而实现对所述承诺随机数进行加密。

[41]步骤 S16: 向所述区块链提交交易数据。

[42]在本实施例中, 所述交易数据可以包括所述交易额承诺、所述承诺随机数第一密文和所述承诺随机数第二密文。所述汇出方设备可以向所述区块链提交交易数据; 以便于将所述交易额承诺和所述承诺随机数第一密文记入所述汇出方账户, 以及将所述交易额承诺和所述承诺随机数第二密文记入所述汇入方账户。

[43]具体地, 在向所述区块链提交交易数据以后, 所述区块链中的共识区块链节点可以基于所述交易额承诺更新所述汇出方余额承诺; 可以基于所述承诺随机数第一密文更新所述汇出方随机数密文; 可以基于所述交易额承诺更新所述汇入方余额承诺; 可以基于所述承诺随机数第二密文更新所述汇入方随机数密文。其中, 所述共识区块链节点可以为基于所述区块链的共识机制确定出的区块链节点。这样可以实现将交易数据记录至区块链网络中各个区块链节点分别维护的区块链账本中, 避免发生篡改。

[44]更新后的汇出方余额承诺可以为所述汇出方余额承诺和所述交易额承诺的商。鉴于所述汇出方余额承诺和所述交易额承诺由基于具有同态性的承诺机制来实现, 这样可以实现从所述汇出方余额中扣除所述交易额、以及从所述汇出方随机数中扣除所述承诺随

机数。例如,所述汇出方余额承诺可以为 $PC(r_A, s_A) = g^{r_A} h^{s_A}$; 所述交易额承诺可以为 $PC(r_t, t) = g^r h^t$; 更新后的汇出方余额承诺可以为 $PC(r_A - r, s_A - t) = PC(r_A, s_A) / PC(r, t) = g^{(r_A - r)} h^{(s_A - t)}$ 。

5 [45]更新后的汇出方随机数密文可以为所述汇出方随机数密文和所述承诺随机数第一密文的商。鉴于所述汇出方随机数密文和所述承诺随机数第一密文由同态加密算法计算得到, 这样可以实现从所述汇出方余额中扣除所述交易额、以及从所述汇出方随机数中扣除所述承诺随机数。例如, 所述汇出方随机数密文可以为 $OU_A(r_A) = u1^{r_A} v1^{s_A}$; 所述承诺随机数第一密文可以为 $OU_A(r) = u1^r v1^t$; 更新后的汇出方随机数密文可以为 $OU_A(r_A - r) = OU_A(r_A) / OU_A(r) = u1^{r_A - r} v1^{s_A - t}$ 。

10 [46]更新后的汇入方余额承诺可以为所述汇入方余额承诺和所述交易额承诺的积。鉴于所述汇入方余额承诺和所述交易额承诺由基于具有同态性的承诺机制来实现, 这样可以实现在所述汇入方余额中增加所述交易额、以及在所述汇入方随机数中增加所述承诺随机数。例如, 所述汇入方余额承诺可以为 $PC(r_B, s_B) = g^{r_B} h^{s_B}$; 所述交易额承诺可以为 $PC(r_t, t) = g^r h^t$; 更新后的汇入方余额承诺可以为 $PC(r_B + r, s_B + t) = PC(r_B, s_B) PC(r, t) = g^{(r_B + r)} h^{(s_B + t)}$ 。其中, r_B 为汇入方随机数; s_B 为汇入方余额; g 和 h 分别为已知参数。

15 [47]更新后的汇入方随机数密文可以为所述汇入方随机数密文和所述承诺随机数第二密文的积。鉴于所述汇入方随机数密文和所述承诺随机数第一密文由同态加密算法计算得到, 这样可以实现在所述汇入方余额中增加所述交易额、以及在所述汇入方随机数中增加所述承诺随机数。例如, 所述汇入方随机数密文可以为 $OU_B(r_B) = u2^{r_B} v2^{s_B}$; 所述承诺随机数第二密文可以为 $OU_B(r) = u2^r v2^t$; 更新后的汇入方随机数密文可以为 $OU_B(r_B + r) = OU_B(r_B) OU_B(r) = u2^{r_B + r} v2^{s_B + t}$ 。

20 [48]在本实施例的一个实施方式中, 所述汇出方设备还可以基于零知识证明技术, 生成第一零知识证明; 可以在所述交易数据中添加所述第一零知识证明; 以供共识区块链节点验证: 所述交易额不小于 0。所述零知识证明技术例如可以基于 zkSNARK 方案实现。
25 进一步地, 所述零知识证明技术可以包括区间证明技术 (Range Proof)。如此所述汇出方设备可以基于所述区间证明技术, 生成第一区间证明; 可以在所述交易数据中添加所述第一区间证明; 以供共识区块链节点验证: 所述交易额不小于 0。所述区间证明技术例如可以基于 Bulletproofs 方案或 Borromean 环签名方案实现。

[49]在本实施例的一个实施方式中,所述汇出方设备还可以基于零知识证明技术,生成第二零知识证明;可以在所述交易数据中添加所述第二零知识证明;以供共识区块链节点验证:所述交易额不大于汇出方余额。进一步地,所述零知识证明技术可以包括区间证明技术。如此所述汇出方设备可以基于所述区间证明技术,生成第二区间证明;可以在所述交易数据中添加所述第二区间证明;以供共识区块链节点验证:所述交易额不大于汇出方余额。

[50]在本实施例的一个实施方式中,所述汇出方设备还可以基于零知识证明技术,生成第三零知识证明;可以在所述交易数据中添加所述第三零知识证明;以供共识区块链节点验证:用于计算交易额承诺的承诺随机数、用于计算承诺随机数第一密文的承诺随机数、和用于计算承诺随机数第二密文的承诺随机数一致,且用于计算交易额承诺的交易额、用于计算承诺随机数第一密文的交易额、和用于计算承诺随机数第二密文的交易额一致。

[51]例如,所述第三零知识证明可以为 (C, D, E, a, b) 。其中, $C = g^{r^*} h^{t^*}$; $D = u l^{r^*} v l^{t^*}$; $E = u 2^{r^*} v 2^{t^*}$; $x = \text{Hash}(C, D, E)$; $a = r^* + xr$; $b = t^* + xt$ 。其中, r^* 、 t^* 为所述汇出方设备生成的随机数,分别与承诺随机数 r 、交易额 t 相对应;Hash表示哈希运算。

[52]共识区块链节点可以在验证 $g^a h^b = CT^x$ 、 $u l^a v l^b = DM^x$ 、 $u 2^a v 2^b = EN^x$ 同时成立时认为:用于计算交易额承诺的承诺随机数、用于计算承诺随机数第一密文的承诺随机数、和用于计算承诺随机数第二密文的承诺随机数一致,且用于计算交易额承诺的交易额、用于计算承诺随机数第一密文的交易额、和用于计算承诺随机数第二密文的交易额一致。其中, T 为交易额承诺 $PC(r, t)$; M 为承诺随机数第一密文 $OU_A(r)$; N 为承诺随机数第二密文 $OU_B(r)$ 。

[53]在本实施例的一个实施方式中,所述汇出方设备可以根据第一承诺随机数和第一交易额,计算第一交易额承诺;可以根据第二承诺随机数和第二交易额,计算第二交易额承诺。所述第一承诺随机数和所述第二承诺随机数可以由所述汇出方设备生成。所述第一交易额可以为待从汇出方账户转移出的交易额;所述第二交易额可以为待转移至汇入方账户的交易额。这样可以对待从汇出方账户转移出的交易额和待转移至汇入方账户的交易额进行区分。

[54]所述汇出方设备可以根据汇出方账户的同态加密公钥和所述第一交易额,使用同态加密算法对所述第一承诺随机数进行加密,得到承诺随机数第一密文;可以根据汇入方

- 账户的同态加密公钥和所述第二交易额,使用同态加密算法对所述第二承诺随机数进行加密,得到承诺随机数第二密文;可以向所述区块链提交交易数据。所述交易数据可以包括所述第一交易额承诺、所述第二交易额承诺、所述承诺随机数第一密文和所述承诺随机数第二密文;以便于将所述第一交易额承诺和所述承诺随机数第一密文记入所述汇出方账户,以及将所述第二交易额承诺和所述承诺随机数第二密文记入所述汇入方账户。
- [55]具体地,在向所述区块链提交交易数据以后,所述共识区块链节点可以基于所述第一交易额承诺更新所述汇出方余额承诺;可以基于所述承诺随机数第一密文更新所述汇出方随机数密文;可以基于所述第二交易额承诺更新所述汇入方余额承诺;可以基于所述承诺随机数第二密文更新所述汇入方随机数密文。
- [56]更新后的汇出方余额承诺可以为所述汇出方余额承诺和所述第一交易额承诺的商;更新后的汇出方随机数密文可以为所述汇出方随机数密文和所述承诺随机数第一密文的商;更新后的汇入方余额承诺可以为所述汇入方余额承诺和所述第二交易额承诺的积;更新后的汇入方随机数密文可以为所述汇入方随机数密文和所述承诺随机数第二密文的积。
- [57]进一步地,所述汇出方设备还可以基于零知识证明技术,生成第四零知识证明;可以在所述交易数据中添加所述第四零知识证明;以供共识区块链节点验证:用于计算第一交易额承诺的第一交易额与用于计算第二交易额承诺的第二交易额一致;从而避免从汇出方账户转移出的交易额和转移至汇入方账户的交易额不一致。
- [58]在本实施例的一个实施方式中,所述汇出方设备还可以使用所述汇出方账户的签名私钥对所述交易数据进行签名,得到签名数据;可以在所述交易数据中添加所述签名数据;以供共识区块链节点进行签名验证。
- [59]在本实施例中,通过承诺机制和同态加密机制,可以得到交易额承诺、承诺随机数第一密文和承诺随机数第二密文;可以将所述交易额承诺、所述承诺随机数第一密文和所述承诺随机数第二密文提交上链;以便于将交易额承诺和承诺随机数第一密文记入汇出方账户,以及将交易额承诺和承诺随机数第二密文记入汇入方账户。这样一方面,通过承诺机制和同态加密机制,可以对交易额等交易信息实现隐私保护;另一方面,通过将交易额承诺、承诺随机数第一密文和承诺随机数第二密文提交上链,可以实现无需汇入方设备参与便可以在区块链上顺利完成交易。因此本实施例可以在非交互条件下实现隐私保护。

[60]进一步地, 所述汇出方账户可以在所述区块链中登记有汇出方余额承诺和汇出方随机数密文; 所述汇入方账户可以在所述区块链中登记有汇入方余额承诺和汇入方随机数密文。这样一方面, 可以对账户余额、交易额均实现隐私保护; 另一方面, 通过在链上登记汇出方随机数密文和汇入方随机数密文, 还可以避免在链下本地保管汇出方随机数和汇入方随机数造成的负担, 免去丢失风险。

[61]请参阅图 1、图 2 和图 3。为了便于理解, 以下介绍本说明书实施例的一个场景示例。

[62]在本实施例中, 假定汇出方账户可以为 Account_A。所述汇出方账户 Account_A 可以具有签名公私钥对和加密公私钥对。所述汇出方账户 Account_A 在所述区块链中还可以登记有汇出方余额承诺 $PC(r_A, s_A) = g^{r_A} h^{s_A}$ 、汇出方随机数密文 $OU_A(r_A) = u1^{r_A} v1^{s_A}$ 。其中, r_A 为汇出方随机数; s_A 为汇出方余额; g 和 h 分别为已知参数; $u1$ 和 $v1$ 分别为汇出方账户的同态加密公钥的一部分。

[63]假定汇入方账户可以为 Account_B。所述汇出方账户 Account_B 可以具有签名公私钥对和加密公私钥对。所述汇出方账户 Account_B 在所述区块链中还可以登记有汇入方余额承诺 $PC(r_B, s_B) = g^{r_B} h^{s_B}$ 、汇入方随机数密文 $OU_B(r_B) = u2^{r_B} v2^{s_B}$ 。其中, r_B 为汇入方随机数; s_B 为汇入方余额; g 和 h 分别为已知参数; $u2$ 和 $v2$ 分别为汇入方账户的同态加密公钥的一部分。

[64]在本实施例中, 本场景示例的交易可以实现从汇出方账户 Account_A 中转移交易额 t 至汇入方账户 Account_B。具体地, 汇出方设备可以生成承诺随机数 r ; 可以根据承诺随机数 r 和交易额 t , 计算交易额承诺 $PC(r, t) = g^r h^t$; 可以根据汇出方账户的同态加密公钥和所述交易额 t , 使用同态加密算法对所述承诺随机数 r 进行加密, 得到承诺随机数第一密文 $OU_A(r) = u1^r v1^t$; 可以根据汇入方账户的同态加密公钥和所述交易额 t , 使用同态加密算法对所述承诺随机数 r 进行加密, 得到承诺随机数第二密文 $OU_B(r) = u2^r v2^t$ 。

[65]在本实施例中, 所述汇出方设备可以生成第一零知识证明 $RP1$ 、第二零知识证明 $RP2$ 和第三零知识证明 $RP3$ 。所述第一零知识证明 $RP1$ 可以用于验证: $t \geq 0$ 。所述第二零知识证明 $RP2$ 可以用于验证: $s_A - t \geq 0$ 。所述第三零知识证明 $RP3$ 可以用于验证: $PC(r, t) = g^r h^t$ 中的承诺随机数 r 、 $OU_A(r) = u1^r v1^t$ 中的承诺随机数 r 、 $OU_B(r) = u2^r v2^t$ 中的承诺随机数 r 一致、且 $PC(r, t) = g^r h^t$ 中的交易额 t 、 $OU_A(r) = u1^r v1^t$ 中的交易额 t 、 $OU_B(r) = u2^r v2^t$ 中的交易额 t 一致。

[66]在本实施例中,所述汇出方设备可以使用汇出方账户的签名私钥对 $[PC(r,t)$ 、 $OU_A(r)$ 、 $OU_B(r)$ 、 $RP1$ 、 $RP2$ 、 $RP3]$ 进行签名,得到签名数据 SIG;可以向区块链提交交易数据 $[PC(r,t)$ 、 $OU_A(r)$ 、 $OU_B(r)$ 、 $RP1$ 、 $RP2$ 、 $RP3$ 、SIG]。

5 [67]在本实施例中,所述区块链可以根据共识机制确定出共识区块链节点。所述共识区块链节点可以利用相关技术中的防双花或防重放机制,验证该交易是否已经执行过。如果已经执行过,所述共识区块链节点可以拒绝该交易。

[68]如果未执行过,所述共识区块链节点可以验证交易数据中的签名数据 SIG 是否正确。如果不正确,所述共识区块链节点可以拒绝该交易。

10 [69]如果正确,所述共识区块链节点可以对交易数据中的第一零知识证明 $RP1$ 进行检查,以验证是否满足 $t \geq 0$ 。如果不满足,所述共识区块链节点可以拒绝该交易。

[70]如果满足,所述共识区块链节点可以对交易数据中的第二零知识证明 $RP2$ 进行检查,以验证是否满足 $s_A - t \geq 0$ 。如果不满足,所述共识区块链节点可以拒绝该交易。

15 [71]如果满足,所述共识区块链节点可以对交易数据中的第三零知识证明 $RP3$ 进行检查,以验证是否满足: $PC(r,t) = g^r h^t$ 中的承诺随机数 r 、 $OU_A(r) = u1^r v1^t$ 中的承诺随机数 r 、 $OU_B(r) = u2^r v2^t$ 中的承诺随机数 r 一致、且 $PC(r,t) = g^r h^t$ 中的交易额 t 、 $OU_A(r) = u1^r v1^t$ 中的交易额 t 、 $OU_B(r) = u2^r v2^t$ 中的交易额 t 一致。如果不满足,所述共识区块链节点可以拒绝该交易。

20 [72]如果满足,所述共识区块链节点可以更新汇出方余额承诺 $PC(r_A, s_A)$ 、汇出方随机数密文 $OU_A(r_A)$ 、汇入方余额承诺 $PC(r_B, s_B)$ 、汇入方随机数密文 $OU_B(r_B)$ 。具体地,更新后的汇出方余额承诺可以为 $PC(r_A - r, s_A - t) = PC(r_A, s_A) / PC(r, t) = g^{(r_A - r)} h^{(s_A - t)}$;更新后的汇出方随机数密文可以为 $OU_A(r_A - r) = OU_A(r_A) / OU_A(r) = u1^{r_A - r} v1^{s_A - t}$;更新后的汇入方余额承诺可以为 $PC(r_B + r, s_B + t) = PC(r_B, s_B) PC(r, t) = g^{(r_B + r)} h^{(s_B + t)}$;更新后的汇入方随机数密文可以为 $OU_B(r_B + r) = OU_B(r_B) OU_B(r) = u2^{r_B + r} v2^{s_B + t}$ 。

25 [73]请参阅图 4。本说明书实施例还提供一种基于区块链的交易装置。所述交易装置可以包括以下单元。

[74]计算单元 20,用于根据承诺随机数和交易额,计算交易额承诺;

[75]第一加密单元 22,用于根据汇出方账户的同态加密公钥和所述交易额,使用同态加

密算法对所述承诺随机数进行加密, 得到承诺随机数第一密文;

[76]第二加密单元 24, 用于根据汇入方账户的同态加密公钥和所述交易额, 使用同态加密算法对所述承诺随机数进行加密, 得到承诺随机数第二密文;

5 [77]提交单元 26, 用于向所述区块链提交交易数据; 所述交易数据包括所述交易额承诺、所述承诺随机数第一密文和所述承诺随机数第二密文; 以便于将所述交易额承诺和所述承诺随机数第一密文记入所述汇出方账户, 以及将所述交易额承诺和所述承诺随机数第二密文记入所述汇入方账户。

[78]请参阅图 5。本说明书实施例还提供一种汇出方设备。所述汇出方设备可以包括存储器和处理器。

10 [79]在本实施例中, 所述存储器可以按任何适当的方式实现。例如, 所述存储器可以为只读存储器、机械硬盘、固态硬盘、或 U 盘等。所述存储器可以用于存储计算机指令。

[80]在本实施例中, 所述处理器可以按任何适当的方式实现。例如, 处理器可以采取例如微处理器或处理器以及存储可由该(微)处理器执行的计算机可读程序代码(例如软件或固件)的计算机可读介质、逻辑门、开关、专用集成电路(Application Specific Integrated Circuit, ASIC)、可编程逻辑控制器和嵌入微控制器的形式等等。所述处理器
15 可以执行所述计算机指令实现以下步骤:

[81]根据承诺随机数和交易额, 计算交易额承诺;

[82]根据汇出方账户的同态加密公钥和所述交易额, 使用同态加密算法对所述承诺随机数进行加密, 得到承诺随机数第一密文;

20 [83]根据汇入方账户的同态加密公钥和所述交易额, 使用同态加密算法对所述承诺随机数进行加密, 得到承诺随机数第二密文;

[84]向所述区块链提交交易数据; 所述交易数据包括所述交易额承诺、所述承诺随机数第一密文和所述承诺随机数第二密文; 以便于将所述交易额承诺和所述承诺随机数第一密文记入所述汇出方账户, 以及将所述交易额承诺和所述承诺随机数第二密文记入所述
25 汇入方账户。

[85]需要说明的是, 本说明书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或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它实施例的不同之处。尤其, 对于装置实施例和设备实施例而言, 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。

[86]本领域技术人员在阅读本说明书文件之后，可以无需创造性劳动想到将本说明书列举的部分或全部实施例进行任意组合，这些组合也在本说明书公开和保护的范围。

5 [87]在 20 世纪 90 年代，对于一个技术的改进可以很明显地区分是硬件上的改进(例如，对二极管、晶体管、开关等电路结构的改进)还是软件上的改进(对于方法流程的改进)。然而，随着技术的发展，当今的很多方法流程的改进已经可以视为硬件电路结构的直接改进。设计人员几乎都通过将改进的方法流程编程到硬件电路中来得到相应的硬件电路结构。因此，不能说一个方法流程的改进就不能用硬件实体模块来实现。例如，可编程逻辑器件 (Programmable Logic Device, PLD) (例如现场可编程门阵列 (Field Programmable Gate Array, FPGA)) 就是这样一种集成电路，其逻辑功能由用户对器件编程来确定。由设计人员自行编程来把一个数字系统“集成”在一片 PLD 上，而不需要请芯片制造厂商来设计和制作专用的集成电路芯片 2。而且，如今，取代手工地制作集成电路芯片，这种编程也多半改用“逻辑编译器(logic compiler)”软件来实现，它与程序开发撰写时所用的软件编译器相类似，而要编译之前的原始代码也得用特定的
10 编程语言来撰写，此称之为硬件描述语言(Hardware Description Language, HDL)，而 HDL 也并非仅有一种，而是有许多种，如 ABEL (Advanced Boolean Expression Language)、AHDL (Altera Hardware Description Language)、Confluence、CUPL (Cornell University Programming Language)、HDCal、JHDL (Java Hardware Description Language)、Lava、Lola、MyHDL、PALASM、RHDH (Ruby Hardware Description Language) 等，
15 目前最普遍使用的是 VHDL (Very-High-Speed Integrated Circuit Hardware Description Language) 与 Verilog2。本领域技术人员也应该清楚，只需要将方法流程用上述几种硬件描述语言稍作逻辑编程并编程到集成电路中，就可以很容易得到实现该逻辑方法流程的硬件电路。

25 [88]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。

30 [89]通过以上的实施方式的描述可知，本领域的技术人员可以清楚地了解到本说明书可借助软件加必需的通用硬件平台的方式来实现。基于这样的理解，本说明书的技术方案

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）执行本说明书各个实施例或者实施例的某些部分所述的方法。

5 [90]本说明书可用于众多通用或专用的计算机系统环境或配置中。例如：个人计算机、服务器计算机、手持设备或便携式设备、平板型设备、多处理器系统、基于微处理器的系统、置顶盒、可编程的消费电子设备、网络 PC、小型计算机、大型计算机、包括以上任何系统或设备的分布式计算环境等等。

10 [91]本说明书可以在由计算机执行的计算机可执行指令的一般上下文中描述，例如程序模块。一般地，程序模块包括执行特定任务或实现特定抽象数据类型的例程、程序、对象、组件、数据结构等等。也可以在分布式计算环境中实践本说明书，在这些分布式计算环境中，由通过通信网络而被连接的远程处理设备来执行任务。在分布式计算环境中，程序模块可以位于包括存储设备在内的本地和远程计算机存储介质中。

15 [92]虽然通过实施例描绘了本说明书，本领域普通技术人员知道，本说明书有许多变形和变化而不脱离本说明书的精神，希望所附的权利要求包括这些变形和变化而不脱离本说明书的精神。