[19] 中华人民共和国国家知识产权局

[51] Int. Cl.
G06Q 30/00 (2006.01)
G06Q 20/00 (2006.01)

[12] 发明专利申请公布说明书

[21] 申请号 200710104873.7

[43] 公开日 2007 年 11 月 28 日

[11] 公开号 CN 101079141A

[22] 申请日 2007.5.23
[21] 申请号 200710104873.7
[30] 优先权
[32] 2006. 5.24 [33] EP [31] 06010888.3
[71] 申请人 国际商业机器公司
地址 美国纽约
[72] 发明人 J·L·卡梅尼施
S·R·霍恩贝格尔

[74] 专利代理机构 北京市中咨律师事务所
代理人 于 静 杨晓光

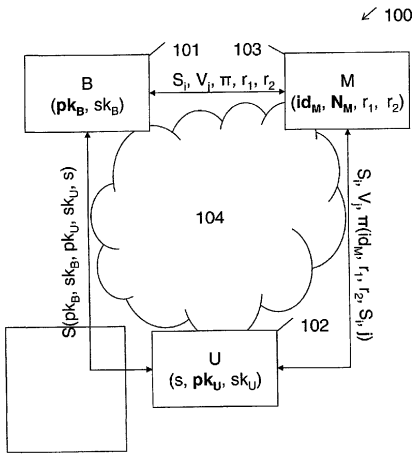
权利要求书 4 页 说明书 28 页 附图 5 页

[54] 发明名称

用于自动确认交易的方法以及电子支付系统

[57] 摘要

本发明涉及一种用于自动确认在具有签名密钥(pk_B，sk_B)的发行方、具有发放方密钥(pk_U，sk_U)的发放方、具有唯一身份(id_M)和交易限制(N_M)的接受方以及确认方之间的交易的方法。本发明进一步涉及一种电子支付系统(100)、计算机程序和计算机程序产品，所述电子支付系统(100)包括银行计算机系统(101)、用户计算机系统(102)、商人计算机系统(103)以及确认方计算机系统(101)。



1. 用于自动确认在具有签名密钥（pk_B，sk_B）的发行方、具有发放方密钥（pk_U，sk_U）的发放方、具有唯一身份（id_M）和交易限制（N_M）的接受方以及确认方之间的交易的方法，所述方法包括以下步骤：

－由所述发放方从所述发行方提取凭单（S），基于所述发行方的签名密钥（pk_B，sk_B）和所述发放方的发放方密钥（pk_U，sk_U），使用所述发行方与所述发放方之间的第一双方协议计算所述凭单（S），

－由所述发放方向所述接受方消费所述凭单（S），其中基于所述凭单（S）、所述发放方与所述接受方之间的交易数（j）、所述接受方所生成的交易询问值（r_1，r_2）以及所述接受方的唯一身份（id_M），使用所述发放方与所述接受方之间的第二双方协议计算洗钱检查值（V_j），

－由所述接受方向所述确认方存放所述凭单（S）、所述交易询问值（r_1，r_2）以及所述洗钱检查值（V_j），以及

－通过将所述洗钱检查值（V_j）与在较早的交易中已向所述确认方存放的洗钱检查值（V_j'）进行比较，由所述确认方通过检验所述发放方与所述接受方之间的交易数低于所述接受方的交易限制（N_M）来确认所述交易。

2. 根据权利要求 1 的方法，其中，

－在所述消费步骤中，对知识的证明（π）由所述发放方计算，并且被传输至所述接受方，

－在所述存放步骤中，向所述确认方存放所述证明（π），以及

－所述确认步骤进一步包括以下步骤：检验所述证明（π）关于所述接受方的唯一身份（id_M）、所述凭单（S）和所述洗钱检查值（V_j）是有效的。

3. 根据权利要求 2 的方法，其中，

－在所述确认步骤中，所述确认方检验：所述证明（π）是基于所述接受方的交易限制（N_M）计算的。

4. 根据权利要求 1 至 3 中任何一项的方法，其中，

－如果所述确认方检测到在较早的交易中向所述确认方存放的洗钱检查值（V_j'）等于当前交易的所述洗钱检查值（V_j），则所述确认步骤进一步包括以下步骤：

－拒绝所述凭单（S），以及

－检索在所述较早的交易中所使用的交易询问值（r_2'）和凭单（S），

－如果在所述较早的交易中所使用的交易询问值（r_2'）与所述交易询问值（r_2'）不同，则由所述确认方基于在所述较早的和当前交易中所使用的凭单（S，S'）计算所述发放方的身份（pk_U）。

5. 根据权利要求 1 至 4 中任何一项的方法，其中，

－在所述确认步骤中，所述确认方将所述凭单（S）与在较早的交易中存放的凭单（S'）进行比较，以及

－如果所述发行方检测到在较早的交易中向所述确认方存放的凭单（S'）等于当前交易的所述凭单（S），则所述确认步骤进一步包括以下步骤：

－检索在所述较早的交易中所使用的交易询问值（r_1'），

－如果在所述较早的交易中所使用的交易询问值（r_1'）与所述交易询问值（r_1'）相等，则所述确认方拒绝所述凭单（S），以及

－如果在所述较早的交易中所使用的交易询问值（r_1'）与所述交易询问值（r_1'）不同，则由所述确认方基于在所述较早的和当前交易中所使用的凭单（S，S'）计算所述发放方的身份（pk_U）。

6. 根据权利要求 4 或 5 的方法，其中，

－在所述提取步骤中，所述发放方向所述发行方提供对钱包秘密（s）的知识的证明，并且所述凭单（S）是基于所述钱包秘密（s）的，

－在所述消费步骤中，计算钱包检查值（T_i）并且将其传送至所述接受方，

－在所述存放步骤中，向所述确认方存放所述钱包检查值（T_i），以及

－在所述确认步骤中，如果所述发行方检测到在较早的交易中已向所

述确认方存放了所述凭单（S）或所述洗钱检查值（V_j），并且所述当前交易的交易询问值（r_1，r_2）等于所述较早的交易中所使用的交易询问值（r_1'，r_2'），则基于所述当前和较早的交易的钱包检查值（T_i，T_i'），由所述确认方计算所述发放方的钱包秘密（s）。

7. 根据权利要求 1 至 6 中任何一项的方法，其中，

－在所述消费步骤中，所述接受方检验所述交易数（j）低于所述交易限制（N_M），并且如果所述交易数（j）超出所述交易限制（N_M），则放弃所述交易。

8. 电子支付系统，所述电子支付系统包括：用于发行预定面额的钱币的银行计算机系统、用于发放所述钱币的用户计算机系统、用于接受所述钱币的商人计算机系统，以及用于确认所述钱币的确认方计算机系统，所述银行计算机系统、所述用户计算机系统、所述商人计算机系统以及所述确认方计算机系统在操作上通过数据网络连接，其中根据权利要求 1 至 7 中任何一项的方法提取、发放、存放和确认所述钱币。

9. 计算机系统，所述计算机系统包括执行权利要求 1 至 7 中任何一项的方法的装置。

10. 一种用于向发放方发行凭单（S）、特别是钱币的发行方，所述发行方被提供用于：

－基于所述发行方的签名密钥（pk_B，sk_B）和所述发放方的发放方密钥（pk_U，sk_U），使用所述发行方与所述发放方之间的第一双方协议计算所述凭单（S）。

11. 一种用于从发放方接受凭单（S）、特别是钱币的接受方，基于发行方的签名密钥（pk_B，sk_B）和所述发放方的发放方密钥（pk_U，sk_U），使用所述发行方与所述发放方之间的第一双方协议计算所述凭单（S），所述接受方被提供用于：

－基于所述凭单（S）、所述发放方与所述接受方之间的交易数（j）、由所述接受方生成的交易询问值（r_1，r_2）以及所述接受方的唯一身份（id_M），通过使用所述发放方与所述接受方之间的第二双方协议计算洗钱检查值

（V_j），

－向确认方存入所述凭单（S）、所述交易询问值（r_1，r_2）以及所述洗钱检查值（V_j）。

12. 一种用于确认在具有签名密钥（pk_B，sk_B）的发行方、具有发放方密钥（pk_U，sk_U）的发放方、具有唯一身份（id_M）和交易限制（N_M）的接受方之间的交易的确认方，所述确认方被提供用于：

－通过将洗钱检查值（V_j）与在较早的交易中已向所述确认方存放的洗钱检查值（V_j'）进行比较，检验所述发放方与所述接受方之间的交易数低于所述接受方的交易限制（N_M）。

13. 一种发放方，所述发放方被提供用于：

－从发行方提取凭单（S），基于所述发行方的签名密钥（pk_B，sk_B）和所述发放方的发放方密钥（pk_U，sk_U），使用所述发行方与所述发放方之间的第一双方协议计算所述凭单（S），

－向接受方消费所述凭单（S），其中，基于所述凭单（S）、所述发放方与所述接受方之间的交易数（j）、由所述接受方生成的交易询问值（r_1，r_2）以及所述接受方的唯一身份（id_M），使用所述发放方与所述接受方之间的第二双方协议计算洗钱检查值（V_j）。

用于自动确认交易的方法以及电子支付系统

技术领域

本发明涉及一种用于自动确认具有签名密钥的发行方、具有发放方密钥的发放方、具有唯一身份和交易限制的接受方，以及确认方之间的交易的方法。本发明还涉及一种电子支付系统、计算机程序和计算机程序产品，其中，电子支付系统包括银行计算机系统、用户计算机系统、商人计算机系统和确认方计算机系统。

背景技术

用于进行和确认交易的方法和系统面临各种竞争目的。虽然一方面它们应当可被证明是正确和安全的，以便参与交易的人都不能获得非法利益（例如通过假设的虚假身份或蓄意更改交易值），但是另一方面参与者的身份及其准确交互应当常常保持匿名。与此同时，实现交易协议的操作应当在计算上是高效率的并且遵循标准过程。

用于在线特别是通过因特网支付货物和服务的电子支付系统是电子交易系统的特别重要的例子。如果没有快速、安全、匿名且易于实现的电子支付系统，电子商务的发展可能会有风险。因此，研究人员和金融机构均已开发了若干电子支付系统。这种系统的一个例子在 J. Camenisch、S. Hohenberger 和 A. Lysyanskaya 的题为“Compact E-Cash”的论文中进行了描述，该论文出版于 EUROCRYPT，Vol. 3494 of LNCS，pages 302-321，2005，在此通过引用的方式将其并入本说明书并称之为 CHL 系统。CHL 系统可被证明是安全和匿名的，即当在该系统内不能重复使用钱币的时候，银行或钱币发行方不能恢复用户或发放方的身份，该用户或发放方在钱币被再次存入银行的时候向商人或接受方消费该钱币。

然而，这种以及类似的系统有一个缺点：它们对电子洗钱（money

laundering）是开放的。在各种情况下，应当检测、阻止或报告涉及、代表例如任意两方之间的高现金流的高交易数。举例来说，应用包括防止逃税、贪污和大规模欺诈。M. Stadler、J.-M. Piveteau 和 J. Camenisch 在题为“Fair Blind Signatures”（EUROCRYPT'95，Vol. 921 of LNCS，pages 209-219，1995）的论文中所描述的系统具有可以随时取消用户的匿名以防止洗钱的可信第三方。然而，由于系统的用户永远都不能确定他们的匿名会在何种情况下被取消，因此具有第三方是电子支付系统的主要缺点。其它的系统，例如由 T. Okamoto 和 K. Ohta 在题为“Diaposable Zero-Knowledge Authentications and their Applications to Untraceable Electronic Cash”（CRYPTO，Vol. 435 of LNCS，pages 481-496，1990）的论文中所描述的电子支付系统，就仅向用户提供了受限形式的匿名以防止误用。在该系统中，用户的钱币是匿名的但却可以相互链接。这严重限制了电子支付系统的匿名。

因此，存在对改进的安全、匿名交易系统以及方法的需要。设计一种可以帮助防止电子洗钱的电子支付系统是个特别的难题。

发明内容

根据本发明的一个方面，提供了一种用于自动确认具有签名密钥的发行方、具有发放方密钥的发放方、具有唯一身份和交易限制的接受方以及确认方之间的交易的方法。该方法包括以下步骤：

－由所述发放方从所述发行方提取凭单（voucher），基于所述发行方的签名密钥和所述发放方的发放方密钥，使用所述发行方与所述发放方之间的第一双方协议计算所述凭单，

－由所述发放方向所述接受方消费所述凭单，其中基于所述凭单、所述发放方与所述接受方之间的交易数、所述接受方所生成的交易询问值以及所述接受方的所述唯一身份，使用所述发放方与所述接受方之间的第二双方协议计算洗钱检查值，

－由所述接受方向所述确认方存放所述凭单、所述交易询问值以及所

述洗钱检查值，以及

－通过将所述洗钱检查值与在较早的交易中已向所述确认方存放的洗钱检查值进行比较，由所述确认方通过检验所述发放方与所述接受方之间的交易数低于所述接受方的交易限制来确认所述交易。

通过检验所述发放方与所述接受方之间的交易数低于所述接受方的交易限制，可以检测任何一个发放方与接受方之间反常的高交易数。

根据所述第一方面的改进实施例，在所述消费步骤中，对知识的证明由发放方计算，并且被传输至所述接受方，在所述存放步骤中，向所述确认方存放所述证明，并且所述确认步骤进一步包括以下步骤：检验所述证明关于所述接受方的唯一身份、所述凭单和所述洗钱检查值是有效的。通过计算和检验对知识的证明，可以推行所述方法的完整性。

根据所述第一方面的进一步改进的实施例，在所述确认步骤中，所述确认方检验所述证明是基于所述接受方的交易限制计算的。通过检验所述证明是基于所述接受方的交易限制计算的，所述确认方可以检测在计算所述证明期间所述接受方是否正试图使用虚假的交易限制。

根据所述第一方面的进一步改进的实施例，如果所述确认方检测到在较早的交易中向所述确认方存放的洗钱检查值等于当前交易的洗钱检查值，则所述确认步骤进一步包括以下步骤：拒绝所述凭单，以及检索在所述较早的交易中所使用的交易询问值和凭单，如果在所述较早的交易中所使用的交易询问值不同于所述交易询问值，则由所述确认方基于在所述较早的和当前交易中所使用的凭单计算所述发放方的身份。通过利用较早交易中的那些洗钱检查值和交易询问值检验所述洗钱检查值和交易询问值，可以识别和惩罚试图超过所述交易限制的各方。

根据所述第一方面的进一步改进的实施例，在所述确认步骤中，所述确认方将所述凭单与在较早的交易中存放的凭单进行比较，并且如果所述确认方检测到在较早的交易中向所述确认方存放的凭单等于当前交易的凭单，则所述确认步骤进一步包括以下步骤：检索在所述较早的交易中所使用的交易询问值，如果在所述较早的交易中所使用的交易询问值等于所述

交易询问值，则由所述确认方拒绝所述凭单，以及如果在所述较早的交易中所使用的交易询问值不同于所述交易询问值，则由所述确认方基于在所述较早的和当前交易中所使用的凭单计算所述发放方的身份。通过利用较早交易中的那些凭单和交易询问值检验所述凭单和交易询问值，可以识别和惩罚试图超支的各方。

根据所述第一方面的进一步改进的实施例，在所述提取步骤中，所述发放方向所述发行方提供对钱包秘密（wallet secret）的知识的证明，并且所述凭单是基于所述钱包秘密的，在所述消费步骤中，计算钱包检查值并且将其传送至所述接受方，在所述存放步骤中，向所述确认方存放所述钱包检查值，并且在所述确认步骤中，如果所述确认方检测到在较早的交易中已向所述确认方存放了所述凭单或所述洗钱检查值，并且当前交易的交易询问值等于在所述较早的交易中所使用的交易询问值，则基于所述当前和较早的交易的钱包检查值，由所述确认方计算所述发放方的钱包秘密。通过计算所述发放方的钱包秘密以防其超支或超过所述交易限制，可以识别其所有的凭单。

根据所述第一方面的进一步改进的实施例，在所述消费步骤中，所述接受方检验所述交易数低于所述交易限制，并且如果所述交易数超出所述交易限制，则放弃所述交易。通过检验特定交易的交易数低于所述接受方的交易限制，所述接受方本身可以防止超过其预定交易限制的交易。

根据本发明的第二方面，公开了一种电子支付系统。所述电子支付系统包括：用于发行预定面额的钱币的银行计算机系统、用于发放所述钱币的用户计算机系统、用于接受所述钱币的商人计算机系统，以及用于确认所述钱币的确认方计算机系统，所述银行计算机系统、所述用户计算机系统、所述商人计算机系统以及所述确认方计算机系统在操作上通过数据网络连接。根据依照所述第一方面的方法对所述钱币进行提取、发放、存放和确认。通过提供具有银行计算机系统、用户计算机系统、商人计算机系统以及确认方计算机系统的电子支付系统，在这些实体之间的交易可以安全、匿名且高效地在线进行，而不会引入电子洗钱的风险。

根据本发明的第三和第四方面，公开了一种计算机程序和计算机程序产品。它们包括可由计算机系统的至少一个处理器执行的程序指令，并且当由所述至少一个处理器执行所述程序指令时，实现根据本发明的第一方面所述发行方、所述发放方、所述接受方或所述确认方的所有步骤。通过提供计算机程序或计算机程序产品，可以易于提供根据所述第一方面的方法的操作并将其分发至计算机系统。

附图说明

当结合附图参照下面对依照本发明的、当前优选但仍然是说明性实施例的详细描述，将更为全面地理解本发明及其实施例。

附图说明了：

图 1 示出了依照本发明的实施例的电子支付系统，其包括组合的银行和确认方计算机系统、用户计算机系统和商人计算机系统；以及

图 2A 至图 2D 示出了依照本发明的实施例的方法的流程图。

具体实施方式

图 1 示出了电子支付系统 100，其包括组合的银行和确认方计算机系统 101、用户计算机系统 102 和商人计算机系统 103。组合的银行和确认方计算机系统 101、用户计算机系统 102 以及商人计算机系统 103 通过数据网络 104 连接。

组合的银行和确认方计算机系统 101 属于银行 B，在下面也被称为发行方或确认方。按照电子支付系统的惯例，将假设组合的银行和确认方计算机系统 101 负责发行和接受电子支付系统 100 的钱币。因此，图 1 中仅示出了单个实体并在下面对其进行了描述。然而，对本领域的技术人员将显而易见的是可以由分离的实体（即分离的发行方和确认方）来实现两个角色（即下面进一步描述的发行和确认钱币的角色）。

借助于组合的银行和确认方计算机系统 101 与用户计算系统 102 之间的第一双方协议，银行 B 可以向用户 U（下面也称为发放方）发行预定面

额的钱币（下面也称为凭单 S）。为了验证由组合的银行和确认方计算机系统 101 所发行的凭单 S，组合的银行和确认方计算机系统 101 包括签名密钥（pk_B，sk_B）。签名密钥（pk_B，sk_B）是两部分不对称的签名密钥，其中第一部分 pk_B 是公用的并且第二部分 sk_B 是保密的或专用的。

组合的银行和确认方计算机系统 101 在生成凭单 S 期间使用签名密钥（pk_B，sk_B）。这可以通过使用签名算法来对用户计算机系统 102 所提供的值签名而发生。可选地，签名密钥（pk_B，sk_B）可以用作第一双方协议中组合的银行和确认方计算机系统 101 的输入以计算凭单 S，其中不显示签名密钥（pk_B，sk_B）的保密部分 sk_B。

用户计算机系统 102 包括发放方密钥（pk_U，sk_U），其也是具有公用部分 pk_U 和保密部分 sk_U 的两部分密钥。另外，用户计算机系统 102 包括钱包秘密 s，其用于生成代表由组合的银行和确认方计算机系统 101 所发行的凭单 S 的序号 S_i。此外，发放方密钥 sk_U 的专用部分、钱包秘密 s 以及所得出的序号 S_i 并没有向组合的银行和确认方计算机系统 101 公开。作为替代，用户计算机系统 102 仅向第一双方协议中组合的银行和确认方计算机系统 101 证明这些和其它参数的知识。

通过唯一身份 id_M 标识商人计算机系统 103。另外，商人计算机系统 103 与交易限制 N_M 关联，交易限制 N_M 为组合的银行和确认方计算机系统 101 以及用户计算机系统 102 所知。交易限制 N_M 可以限制例如在给定情况下允许在任何单个的用户计算机系统 102 与商人计算机系统 103 之间进行的交易数。

取决于凭单 S 的性质，这还可以代表交易值的限制。例如，如果每个凭单 S 都具有一美元的预定值，那么，如果交易限制 N_M 被设为 100，则用户仅可以向与商人计算机系统 103 关联的商人 M 消费 100 美元。可以随电子支付系统 100 或其它凭单发行和兑换系统提供和使用对交易限制 N_M 和凭单 S 的其它定义。在电子支付系统 100 中，例如，可以存在具有不同值的凭单 S，例如，具有 1 美分、10 美分、1 美元、10 美元和 100 美元的关联值的凭单。同样，交易限制 N_M 可以对于单次购买、固定时期（例如一

周、一个月或一年）有效，或者可以对于凭单 S、签名密钥（pk_B，sk_B）、发放方密钥（pk_U，sk_U）或商人计算机系统 103 的唯一身份 id_M 的整个使用期限有效。

在第二双方协议中，用户计算机系统 102 向商人计算机系统 103 证明其拥有之前由组合的银行和确认方计算机系统 101 发行的凭单 S。为此目的，用户计算机系统 102 计算证明π。另外，用户计算机系统 102 基于由商人计算机系统 103 生成的询问值 r_1 计算洗钱检查值 V_j。洗钱检查值 V_j 强制商人 M 与用户 U 之间的交易限制 N_M 不被超过。

在用户计算机系统 102 使用与商人计算机系统 103 的第二双方协议从事交易之前，用户计算机系统 102 检查已经与商人计算机系统 103 进行的交易数。如果超过交易限制 N_M，则诚实的用户计算机系统 102 将不从事交易。同样地，如果超过交易限制 N_M，则商人计算机系统 103 也将不从事交易。为了推行这种诚实，基本数学特性（underlying mathematical property）保证商人计算机系统 103 仅可以生成与交易限制 N_M 允许的一样多的、用于各公用发放方密钥 pk_U 的不同交易询问值 r_1。因此，各方都不会允许与特定的用户计算机系统 102 发生超过限制数量的交易 N_M。然而，如果用户计算机系统 102 和商人计算机系统 103 都不诚实，则它们被迫重用先前在较早的交易中使用的交易询问值 r_1 或生成虚假的证明π。

在第三双方协议中，商人计算机系统 103 向组合的银行和确认方计算机系统 101 存放从用户计算机系统 102 获得的凭单 S。除了凭单 S 之外，商人计算机系统 103 还存放交易询问值 r_1、洗钱检查值 V_j 以及在与用户计算机系统 102 的第二双方协议中所计算的证明π。组合的银行和确认方计算机系统 101 然后检验证明π相对于凭单 S、交易询问值 r_1、身份 id_M 和试图存放凭单 S 的商人计算机系统 103 的交易限制 N_M 实际有效。如果证明π无效，则其拒绝存放。这特别意味着与凭单 S 关联的值将不会存入商人 M。

另外，组合的银行和确认方计算机系统 101 检查是否已经在较早的交易中存放了凭单 S。如果是这种情况，则其将调查交易询问值 r_1 以便识别违犯者。如果检测到在较早的交易中使用了同样的交易询问值 r_1'，则商人

M 将归咎于试图两次存放凭单 S。因此凭单 S 被拒绝，并且视情况，可以公布商人 M 的身份 id_M。如果交易询问值 r_1 和 r_1'不同，则用户 U 应受谴责。在这种情况下，基于当前和先前的交易，可以确定用户 U 的公用部分 pk_U。因此，组合的银行和确认方计算机系统 101 可以揭露欺骗性用户计算机系统 102 的身份或者可以揭露与发放方密钥（pk_U，sk_U）关联的钱包秘密。在这种情况下，组合的银行和确认方计算机系统 101 可以识别向用户计算机系统 102 发行的所有凭单 S 以及由此而涉及的交易。

在进一步的步骤中，组合的银行和确认方计算机系统 101 还通过比较洗钱检查值 V_j 和较早交易中的洗钱检查值来检验是否并未超过交易限制。如果检测到商人计算机系统 103 正试图存放之前已向组合的银行和确认方计算机系统 101 存放的洗钱检查值 V_j，则其还将调查交易询问值 r_1 以便识别违犯者。如果较早的交易基于与当前交易相同的交易询问值 r_1，则商人计算机系统 103 单独受到谴责。取决于电子支付系统 100 的安全设置，组合的银行和确认方计算机系统 101 可以拒绝存放凭单 S 和/或公布欺骗性商人 M 的身份 id_M。如果其不同，则用户计算机系统也已经有意超过了交易限制 N_M 并且可以如上所述受到惩罚。

可以以硬件或软件实现发行方和确认方 B、发放方 U 和接受方 M 的角色，例如通过专用计算机系统 101、102 和 103 或运行在通用计算机系统 101、102 和 103 上的软件模块或其组合。计算机系统 101、102 或 103 中可以包括计算机可读介质，例如，含有可由计算机系统 101、102 或 103 中所包括的处理器执行的程序的程序指令。计算机可读介质可以是例如 CD-ROM、闪存卡、硬盘或任何其它合适的计算机可读介质。

图 2A 至图 2D 示出了依照本发明的实施例的方法的流程图。

图 2A 示出了在组合的银行和确认方计算机系统 101 与用户计算机系统 102 之间执行的第一双方协议 200。第一双方协议 200 用来设置用于在电子支付系统 100 中实现交易以及用于由用户计算机系统 102 从组合的银行和确认方计算机系统 101 提取钱币的系统参数。

在步骤 201，设置电子支付系统 100 的重要的系统参数。特别地，定

义了保密参数 k、双线性映射 e、钱包尺寸 l 以及双线形映射 e 上的散列（hash）函数 H_1 和 H_2。这些参数设置了用户计算机系统 102 可以在其钱包中存储多少凭单 S 的系统边界。

在步骤 202，发行方 B 生成包括专用部分 pk_B 和保密部分 sk_B 的非对称签名密钥（pk_B，sk_B）。

在步骤 203，发放方 U 也为用户计算机 102 生成包括公用部分 pk_U 和保密部分 sk_U 的非对称发放方密钥（pk_U，sk_U）。

在步骤 204，用户计算机 102 生成随机值，特别是钱包秘密 s，其用于生成用于每个凭单 S 或者将要从组合的银行和确认方计算机系统 101 提取的钱币的唯一序号 S_i。由用户计算机系统 102 对所生成的随机值 s 保持专用。

在步骤 205，用户计算机系统 102 向组合的银行和确认方计算机系统 101 证明其保密密钥部分 sk_U 以及随机值 s 和 t 的知识。另外，用户计算机系统 102 在其保密的发放方密钥部分 sk_U 以及随机值 s 上获得签名 σ。因此，组合的银行和确认方计算机系统 101 获得钱包秘密 s 的可检验加密，且在该加密上具有用户签名。组合的银行和确认方计算机系统 101 并不学习有关发放方密钥的保密部分 sk_U 或随机值 s 的任何内容。

在步骤 206，用户计算机系统 102 存储从组合的银行和确认方计算机系统 101 提取的凭单 S。实际上可以提取并一起存储整组凭单（包括 2^l 个凭单），每个凭单 S 由序号 S_i 表示。根据 CHL 系统，可以以尺寸 O（l+k）特别紧致地存储凭单，其中 k 是系统的保密参数。

图 2B 示出了将要在用户计算机系统 102 与商人计算机系统 103 之间执行的第二双方协议 210，其用于消费向用户计算机系统 102 发行的凭单 S。

在步骤 211，用户计算机系统 102 检查其与特定的商人计算机系统 103 的消费限制 N_M。如果其已经达到交易限制 N_M，则诚实的用户计算机系统 102 不会从事与商人计算机系统 103 的交易。基于该检查，用户计算机系统 102 还计算用户计算机系统 102 与商人计算机系统 103 之间的先前的交易数 j。用户计算机系统 102 还确定将要用于当前交易的序号 S_i 的指数 i，

其通常与下一未使用的凭单 S 相关联。同样地，商人计算机系统 103 将确定其是否低于与用户计算机系统 102 的交易限制 N_M。

在步骤 212，商人计算机系统 103 将生成交易询问值，包括例如随机参数 r_1 和 r_2。然后将随机询问值 r_1 和 r_2 传送至用户计算机系统 102。

在步骤 213，用户计算机系统 102 基于具有序号 S_i 的凭单 S 以及交易询问值 r_1 和 r_2 计算钱包检查值 T_i。用户计算机系统 102 还计算在用户计算机系统 102 与商人计算机系统 103 之间第 j 个交易的洗钱检查值 V_j。

在步骤 214，通过向商人计算机系统 103 传输第 i 个序号 S_i 及其关联的检查值 T_i 和 V_j，将要使用的凭单 S 发送给商人计算机系统 103。

在步骤 215，用户计算机系统 102 为商人计算机系统 103 生成对值 i、j、sk_U、s 和 σ 的知识的证明 π。该证明 π 可以在用户计算机系统 102 与商人计算机系统 103 之间交互式实现或者使用 Fiat-Shamir 启发式方法非交互式实现。

在步骤 216，商人计算机系统 103 检验证明 π。如果证明 π 对于在交易中使用的序号 S_i 有效，则商人计算机系统 103 接受与序号 S_i 和证明 π 关联的凭单 S，例如用于支付。

图 2C 示出了由组合的银行和确认方计算机系统 101 所执行的确认过程的第一部分 220，其用于确定是否已在较早的交易中消费了凭单 S。

在步骤 221，商人计算机系统 103 提交凭单 S，其包括交易询问值 r_1 和 r_2、凭单 S 的序号 S_i、检查值 T_i 和 V_j 以及证明 π。以这样的方式，商人计算机系统 103 试图兑换与凭单 S 关联的值，例如通过请求由组合的银行和确认方计算机系统 101 存入关联的币值。

在步骤 222，组合的银行和确认方计算机系统 101 检验证明 π 对于序号 S_i 以及交易询问值 r_1 和 r_2 有效。如果证明 π 无效，则组合的银行和确认方计算机系统 101 立刻拒绝接受交易。

在步骤 223，组合的银行和确认方计算机系统 101 检验之前还未向组合的银行和确认方计算机系统 101 存放与序号 S_i 关联的凭单 S。

如果之前已向组合的银行和确认方计算机系统 101 存放了凭单 S，则

在步骤 224，执行进一步的检查，检验是否在较早的交易中使用了相同的交易询问值 r_1。

如果是这样的情况，则商人计算机系统 103 试图第二次存放凭单 S；因此在步骤 225，组合的银行和确认方计算机系统 101 拒绝该凭单。

否则，用户便已重复消费（double-spend）并且应当受到银行 B 的惩罚。在这种情况下，组合的银行和确认方计算机系统 101 在步骤 226 接受来自商人计算机系统 103 的凭单但却识别与公钥部分 pk_U 关联的用户计算机系统 102。

在任选步骤 227，取决于在图 2A 中所示的提取协议期间所使用的电子支付系统 100 的安全设置，组合的银行和确认方计算机系统 101 可以基于在其中使用了相同序号 S_i 的当前交易和较早的交易来计算用户计算机系统 102 的钱包秘密 s。

因此，在步骤 228，组合的银行和确认方计算机系统 101 可以基于钱包秘密 s 或公用用户密钥 pk_U 识别向用户计算机系统 102 发行的凭单 S 的所有其它的序号 S_i。

如果在步骤 223 测试的序号 S_i 是唯一的，则在图 2D 所示的第二部分 230 中继续确认，其用于检验当前交易并未超过交易限制 N_M。

在步骤 231，检验为交易而计算的洗钱检查值 V_j。这是通过在步骤 232 将当前的洗钱检查值 V_j 与在较早的交易中已经提交给组合的银行和确认方计算机系统 101 的先前的洗钱检查值 V_j 进行比较来完成的。

如果洗钱检查值 V_j 是唯一的，则组合的银行和确认方计算机系统 101 在步骤 233 接受所提交的凭单 S，并且该方法结束。

否则，在进一步的步骤 234，组合的银行和确认方计算机系统 101 检验在当前和较早的交易中是否使用了相同的交易询问值 r_2。

如果使用了相同的交易值 r_2，则仅是商人计算机系统 103 将受到谴责。在这种情况下，在步骤 235 拒绝凭单，从而使得与交易关联的值不被存入商人计算机系统 103。

否则，用户也要受到谴责。因此，在步骤 236、237 和 238，识别与公

钥 pk_U 关联的用户计算机系统 102 的身份，解密其钱包秘密 s，并且基于该信息，通过上述组合的银行和确认方计算机系统 101 计算该用户的凭单 S 的其它序号 S_i。另外，在步骤 235 还将通过拒绝当前的凭单来惩罚商人 M。

由于基础问题的数学复杂性，图 2A 至图 2D 中所呈现的流程图仅可以呈现电子支付系统 100 中信息流的高级概况。在以下数学描述中详述了为了实现组合的银行和确认方计算机系统 101、用户计算机系统 102 和商人计算机系统 103 之间的协议而发生的详细操作。

安全的定义:

概括 CHL 系统的定义以处理重复消费以外的违犯。我们的脱机电子货币情况由三个常见的参与者组成: 用户、银行和商人; 连同算法 *BKeygen*、*UKeygen*、*Withdraw*、*Spend*、*Deposit*、{*DetectViolation*$^{(i)}$, *IdentifyViolator*$^{(i)}$, *VerifyViolation*$^{(i)}$}、*Trace* 和 *VerifyOwnership*。非正式地，*BKeygen* 和 *UKeygen* 分别是银行和用户的密钥生成算法。用户在 *Withdraw* 期间与银行交互以获得具有 2^l 个钱币的钱包; 银行将任选的跟踪信息存储在数据库 *D* 中。在 *Spend* 中，用户从其钱包向商人消费一个钱币; 结果商人获得该钱币的序号 *S*，商人 M 记录该钱币的定位器 *V* 以及对有效性的证明 π。在 *Deposit* 中，无论诚实的商人 M 何时从用户接受钱币 $C=(S, V, \pi)$，都会保证银行将接受该钱币用于存放。银行将 $C=(S, V, \pi)$存储在数据库 *L* 中。然而，此时，银行需要确定 *C* 是否违犯任何的系统条件。

对于每种违犯 *i*，都定义了算法的元组{*DetectViolation*$^{(i)}$, *IdentifyViolator*$^{(i)}$, *VerifyViolation*$^{(i)}$}。在此，我们有两种违犯。

违犯 1（重复消费）: 在 *DetectViolation*$^{(1)}$中，银行测试 *L* 中的两个钱币 $C_1=(S_1, V_1, \pi_1)$和 $C_2=(S_2, V_2, \pi_2)$是否具有相同的序号 $S_1 = S_2$。如果如此，则银行在(C_1, C_2)上运行 *IdentifyViolator*$^{(1)}$算法，并且获得违犯者的公钥 *pk* 以及犯罪证明 Π。任何人都可以在(pk, S_1, V_1, Π)上运行 *VerifyViolation*$^{(1)}$以确信具有公钥 *pk* 的用户重复消费了具有序号 S_1 的钱币。

违犯 2（洗钱）：在 *DetectViolation*$^{(2)}$中，银行测试 L 中的两个钱币 $C_1=(S_1, V_1, \pi_1)$和 $C_2=(S_2, V_2, \pi_2)$是否具有相同的商人记录定位器 $V_1 = V_2$。如果如此，则银行在(C_1, C_2)上运行 *IdentifyViolator*$^{(2)}$算法，并且获得违犯者的公钥 *pk* 以及犯罪证明 Π。任何人都可以在(pk, S_1, V_1, Π)上运行 *VerifyViolation*$^{(2)}$以确信具有公钥 *pk* 的用户利用具有商人记录定位器 V_1 的钱币超过了有界匿名商业限制。

视情况，在任何违犯之后，银行还可以在有效的犯罪证明 Π 上运行 *Trace* 算法以获得欺骗用户利用公钥 *pk* 连同所有权的证明 Γ 曾消费的所有序号 S_i 的列表。任何人都可以在(pk, S_i, Γ)上运行 *VerifyOwnership* 以确信具有公钥 *pk* 的用户是具有序号 S_i 的钱币的所有者。

另外，将 CHL 系统的安全定义推广用于电子货币。其对正确性、平衡（balance）以及用户的匿名（anonymity of user）的形式化保持不变。概略地，平衡保证诚实的银行绝不会不得不接受超过用户提取的钱币来存放，而用户的匿名确保用户保持完全匿名，除非他们违犯了已知的系统条件之一。下面描述了三个附加的特性。这些特性是 CHL 对重复消费者的识别和跟踪，及其辩解能力（exculpability）的泛化，以便应用于任何具体的违犯，特别是上述那些。令 *params* 为全局参数，包括每钱包的钱币数以及每个商人的消费限制。

识别违犯者：假设诚实的商人（或可能的商人）与对手两次运行 *Spend* 协议，使得输出是 $C_1=(S_1, V_1, \pi_1)$和 $C_2=(S_2, V_2, \pi_2)$。该特性保证在高概率情况下，如果对于某个 i，*DetectViolation*$^{(i)}$(*params*, C_1, C_2)接受，那么 *IdentifyViolator*$^{(i)}$(*params*, C_1, C_2) 输出密钥 *pk* 和证明 Π 使得 *VerifyViolation*$^{(i)}$(*params*, *pk*, S_1, V_1, Π)接受。

跟踪违犯者：假设 *VerifyViolation*$^{(i)}$(*params*, *pk*, *S*, *V*, Π)接受从钱币 C_1、C_2 得出的某种违犯 i。该特性保证在高概率的情况下，*Trace*(*params*, *pk*, C_1, C_2, Π, *D*)输出属于具有 *pk* 的用户的所有钱币的序号 $S_1, \cdots, S_m$ 连同所有权的证明 $\Gamma_1, \cdots, \Gamma_m$，从而使得对于所有的 j，*VerifyOwnership*(*params*, *pk*, S_j, Γ_j)接受。

辨解能力：假设对手参与了任意次与具有密钥 *pk* 的诚实用户的 *Withdraw* 协议，并且随后参与了任意次与相同用户的 *non-violation Spend* 协议。对手然后输出整数 *i*、钱币序号 *S* 以及假设的证明Γ（即具有密钥 *pk* 的用户做出了违犯 *i* 并拥有钱币 *S*）。弱辨解能力特性表明，对于所有的对手来说，*VerifyO*(*params*, *pk*, *S*, Γ)接受的概率可忽略。

此外，对手可以继续使用户参加 *Spend* 协议，迫使他违犯系统条件。对手然后输出(*i*, *S*, *V*,Π)。强辨解能力特性表明，对于所有的对手来说：（1）当 *S* 是不属于具有 *pk* 的用户的钱币序号时，弱辨解能力成立，以及（2）当具有 *pk* 的用户未做出违犯 *i* 时，*VerifyViolation*$^{(i)}$(*params*, *pk*, *S*, *V*,Π)接受的概率可忽略。

技术预备：

电子货币系统使用各种已知的协议作为积木式部件，现在对其进行简要回顾。这些协议中的很多可以在几种不同的复杂度假设下、在可以扩展到我们的电子货币系统的灵活性下显示安全。注意：我们记 *G*=<*g*>表示 *g* 生成组 *G*。

双线性映射

假设 *Bilinear_Setup* 是一种在输入保密参数 1^k 时就输出双线性映射的参数为 $\gamma=(q, g_1, h_1, G_1, g_2, h_2, G_2, G_T, e)$的算法。每组 $G_1=<g_1>=<h_1>$、$G_2=<g_2>=<h_2>$以及 G_T 均具有素数阶 $q=\Theta(2^k)$。高效率可计算的映射 $e:G_1\times G_2\to G_T$ 均是：（*Bilinear*）对于所有的 $g_1\in G_1, g_2\in G_2$ 以及 $a,b\in Z_q$，$e(g_1^a,g_2^b)=e(g_1,g_2)^{ab}$；以及（*Non-degenerate*）如果 g_1 是 G_1 的生成元，并且 g_2 是 G_2 的生成元，那么 $e(g_1,g_2)$ 生成 G_T。

复杂性假设：

我们的方案的安全性依赖于与 CHL 相同的假设，即：

强 RSA 假设：给定 RSA 模 *n* 以及随机元素 $g\in Z_n^*$，难以计算 $h\in Z_n^*$ 以及整数 $e>1$ 使得 $h^e\equiv g$ 按 *n* 取模。模 *n* 具有特殊形式 *pq*，其中 $p=2p'+1$ 和

$q=2q'+1$是安全素数。

y-决策性 Diffie-Hellman 反转假设（y-DDHI）：给定随机生成元$g\in G$（其中 G 具有素数阶 q）、对于随机的$x\in Z_q$的值$\left(g,g^x,\cdots,g^{(x^y)}\right)$，以及值$R\in G$，难以判定是否$R=g^{1/x}$。

外部 Diffie-Hellman 假设（XDH）：假设 *Bilinear_Setup*（1^k）产生双线性映射$e:G_1\times G_2\to G_T$的参数。XDH 假设表明决策性 Diffie-Hellman（DDH）问题在 G_1 中是困难的。这意味着并不存在高效率可计算的同构（isomorphism）$\psi':G_1\to G_2$。

Sum-Free 决策性 Diffie-Hellman 假设（SF-DDH）：假设$g\in G$是 q 阶随机生成元。令 L 为$|q|$的任意多项式函数。令$O_{\bar{a}}(\cdot)$为谕示（oracle），在输入子集$I\subset eq\{1,\cdots,L\}$时输出值$g_1^{\beta_I}$，其中对于某$\bar{a}=(a_1,\cdots,a_L)\in Z_q^L$来说$\beta_I=\prod_{i\in I}a_i$。另外，令 R 为谓词（predicate），使得当且仅当$J\subseteq\{1,\cdots,L\}$相对于I_i是 DDH 独立的时，$R(J,I_1,\cdots,I_t)=1$；也就是说，当$v(I_i)$是当且仅当$j\in I_i$时在位置 j 是一否则是零的、L 长度的向量时，便不存在三个集合I_a,I_b,I_c使得$v(J)+v(I_a)=v(I_b)+v(I_c)$（其中在整数上逐位添加）。

然后，对于所有的概率多项式时间对手（adversary）$A^{(\cdot)}$，

$$Pr[\bar{a}=(a_1,\cdots,a_L)\leftarrow Z_q^L;(J,\alpha)\leftarrow A^{O_{\bar{a}}}(1^{|q|});y_0=g^{\prod_{i\in J}a_i};y_1\leftarrow G;$$

$$b\leftarrow\{0,1\};b'\leftarrow A^{O_{\bar{a}}}(1^{|q|},y_b,\alpha):b=b'\wedge R(J,Q)=1]<1/2+1/poly(|q|),$$

其中 Q 是 A 向$O_{\bar{a}}(\cdot)$进行的查询的集合。

密钥构建块:

1） 已知基于离散对数的、零知识证明

在共用参数模型中，使用用于证明关于离散对数的声明的几个先前已知的结果，例如（1）对以素数或合数为模的离散对数的知识的证明，（2）对表示以两个（可能不同的）素数或合数模数为模的等式的知识的证明，（3）证明承诺（commitment）对两个其它承诺值（commited value）的产生开放，（4）证明承诺值位于给定的整数间隔，以及（5）对先前任意两个的析取（disconjuntion）或合取（conjunction）的证明。这些以合数

为模的协议在强 RSA 假设之下是安全的，并且在离散对数假设之下以素数为模。Fiat-Shamir 启发式方法可以应用于将对知识的这样的证明转换成某消息 m 上对知识的签名证明（signature proofs of knowledge）。

2） CL 签名

根据 Pedersen 承诺方案，公用参数是具有素数阶 q 的组 G，以及生成元 $(g_0,\cdots,g_m)$。为了承诺于值 $(v_1,\cdots,v_m)\in_q^m$，挑选随机的 $r\in_q$ 并且设置 $C=PedCom(v_1,\cdots,v_m;r)=g_0^r\prod_{i=1}^m g_i^{v_i}$。

存在具有两个协议的安全签名方案：（1）在用户与具有密钥（pk_S，sk_S）的签名者之间的高效率协议。共用输入包括 pk_S 和 C（Pederson 承诺）。用户的秘密输入是值 $(v_1,\cdots,v_l,r)$ 的集合，使得 $C=PedCom(v_1,\cdots,v_l;r)$。作为协议的结果，用户获得其承诺值上的签名 $\sigma_{pk_S}(v_1,\cdots,v_l)$，尽管签名者并没有学习关于它们的任何内容。签名具有尺寸 $O(l\log q)$。（2）对用户与检验方之间的签名协议的知识的高效率证明。共用输入是 pk_S 和承诺 C。用户的专用输入是值 $(v_1,\cdots,v_l,r)$ 和 $\sigma_{pks}(v_1,\cdots,v_l)$，使得 $C=PedCom(v_1,\cdots,v_l;r)$。这些签名在强 RSA 假设之下是安全的。出于该阐述的目的，CL 签名实际如何工作无关紧要。

随后的电子货币系统使用与 CL 签名无关的强 RSA 假设。使用双线性映射，人们实际上可以使用其它的签名方案来产生更短的签名。

3） 可检验的加密

在可检验的加密方案中，加密方/证明方使检验方确信：在已知公钥下加密的明文等效于隐藏在 Pedersen 承诺中的值。在“Jan Camenisch and Ivan Damgard. Verifable encryption，group encryption，and their applications to group signatures and signature sharing schemes. In Tatsuaki Okamoto，editor，Advances in Cryptology | ASIACRYPT'00，volumn 1976 of LNCS，pages 331-345. Springer Verlag，2000”中描述了一种技术，其用于将任何在语义上安全的加密方案转成可检验的加密方案。

该技术可以用于将任何在语义上安全的加密方案转成可检验的加密方案。可检验的加密方案是证明方与加密方 P 以及检验方与接收方 V 之间的

双方协议。

概略地，它们的共用输入是公用加密密钥 *pk* 和承诺 *A*。作为协议的结果，*V* 要么拒绝要么获得 *A* 的开头的加密 *c*。协议确保 *V* 仅以可忽略的概率接受不正确的加密并且 *V* 没有学习关于 *A* 的开头的任何有意义的内容。连同相应的保密密钥 *sk*，抄本 *c* 含有足够的信息来高效率地恢复 *A* 的开头。在此我们避开一些细节并且涉及对 Camenisch 和 Damgard 的充分讨论。

4） 双线性 Elgamal 加密

寻找在其中 g^x 足以用于解密并且公钥对于某函数 f 是 $f(g^x)$ 的密码系统。在“Dan Boneh and Matthew Franklin. Identity-based encryption from the Weil pairing. In Joe Kilian，editor，Advances in Cryptology | CRYPTO'01，volumn 2139 of LNCS，pages 213-229. Springer Verlag，2001”中，在基于身份的情况下提供了一个例子。这里使用了 Ateniese 等人的双线性 Elgamal 加密方案，其在由 Sum-Free DDH 所暗示的假设下在语义上是安全的。

特别地，将上述可检验的加密技术应用于下面的 Elgamal 加密的双线性变量。假设在 1^k 上运行 *Bilinear_Setup* 以获得 $\gamma=(q,g_1,h_1,G_1,g_2,h_2,G_2,G_T,e)$，其中具有双线性映射 $e:G_1\times G_2\to G_T$。令 (G,E,D) 表示标准的密钥生成、加密以及解密算法。在输入 $(1^k,\gamma)$ 时，密钥生成算法 *G* 对随机的 $u\in Z_q$ 输出密钥对 $(pk,sk)=\left(ap(g_1,g_2)^u,g_1^u\right)$。主要思想在于值 g_1^u 足够解密。

为了在 *pk* 下加密消息 $m\in G_T$，选择随机的 $k\in Z_q$ 并输出密文 $c=\left(g_2^k,pk^k m\right)=\left(g_2^k,e(g_1,g_2)^{uk}m\right)$。然后，为了用值 g_1^u 解密 $c=(c_1,c_2)$，简单计算 $c_2/e(g_1^u,c_1)$。该加密方案在决策性双线性 Diffie-Hellman（DBDH）假设下已知在语义上是安全的，即对随机的 $a,b,c\in Z_q$ 以及 $X\in G_T$ 给定 $\left(g_2,g_2^a,g_2^b,g_2^c,X\right)$，难以判定是否 $X=e(g_1,g_2)^{abc}$。

5） DY 伪随机函数（PRF）

假设 $G=\langle g\rangle$ 是一组素数阶 *q*。令 *s* 为 Z_q 的随机元素。Dodis 和 Yampolskiy 最近为输入 $x\in Z_q^*$ 提出了伪随机函数 $f_{g,s}^{DY}(x)=g^{1/(s+x)}$。该构造在

y-DDHI 下是安全的。在上述构造中，可以用 Naor-Reingold PRF 替换 DY PRF，并且用更标准的 DDH 假设代替 y-DDHI 假设，代价是将我们的钱包从$O(l+k)$比特加大到$O(l\cdot k)$比特。

有界匿名模型中的紧凑电子货币：

如在 CHL 紧凑电子货币方案中一样，用户从银行提取具有 2^l 个钱币的钱包并逐个花掉。此外，如在 CHL 方案中一样，使用了伪随机函数$F_{(\cdot)}(\cdot)$，其范围是具有大的素数阶 q 的某个组 G。

在高层级，用户通过从适当的域（将在稍后解释）挑选五个值(x,s,t,v,w)来形成具有 $2^l=N$ 个钱币的钱包，并且与银行运行适当的安全协议以在这些值上获得银行的签名 σ。

下面假设用户想要通过从商人 M 购买物品而消费钱币数 i，并且只有与该商人达到 K 个交易才可以匿名。进一步假设这是用户与 M 的第 j 次交易，$j\le K$。与钱包中第 i 个钱币关联的是其序号$S=F_s(i)$。与商人 M 的第 j 次交易关联的是该商人的记录定位器$V=F_v(M,j)$。

根据本发明的实施例，提供了在 Spend 协议中，用户应当给予商人值(S,V)以及这样的（非交互式零知识）证明，即这些值是作为(s,i,v,M,j)的函数来计算的，其中$1\le i\le N,1\le j\le K$，并且(s,v)对应于银行所签名的钱包。S 和 V 是伪随机的，并且因此在计算上没有泄露任何信息；并且因为其是零知识的，所以证明也没有泄露信息。假设用户消费超过 N 个钱币。那么由于仅有 N 个可能的、$F_s(i)$形式的值 S，其中$1\le i\le N$，因此其已经超过一次地使用了某个序号。类似地，假设用户与 M 进行了超过 K 次的交易。那么由于对于固定的 M，仅有 K 个不同的值$V=F_v(M,j)$，$1\le j\le K$，因此其已经超过一次地使用了某个商人记录定位器。因此可以检测到重复消费以及对有界匿名商业模型的违犯。

下面解释如何给出超过一次使用任何的 S 或 V 导致识别。除了 s 和 v，钱包还含有 x，t 和 w。值$x\in_q$使得g^x是可以公开链接至用户的身份的值（其中 g 是组 G 的生成元）。例如，对于某些可计算的函数 f，$f(g^x)$可以是用

户的公钥。假设作为交易的一部分，商人给出随机值$r \neq 0$，并且用户显示$T = g^x F_t(i)^r$和$W = g^x F_w(M, j)^r$以及这样的证明，即 T 和 W 是作为对应于完全相同的钱包以及相同的 i 和 j 的函数(r, x, t, i, w, M, j)而适当计算的。再者，T 和 W 是伪随机的并且因此不会泄露任何信息。

如果用户两次使用相同的序号$S = F_s(i)$，并且 q 适当大，那么在两次不同的交易中其将以高概率接收不同的 r，称它们为 r_1 和 r_2，并且因此将必须用$T_1 = g^x F_t(i)^{r_1}$和$T_2 = g^x F_t(i)^{r_2}$进行响应。容易明白值g^x然后可以如下计算：$g^x = T_1/(T_1/T_2)^{r_1/(r_1-r_2)}$。

示出了也是如此的情况，即如果用户两次使用相同商人的记录定位器数 V，那么可以以相同的方式精确找到g^x。假设在两次交易中商人使用相同的 r。在这种情况下，银行可以仅仅拒绝存放该电子钱币（由于是相同的商人，因此其对自身缺乏适当的随机化负责）。如果假设商人使用了不同的 r，r_1 和 r_2，导致 W_1 和 W_2，则可以得到$g^x = W_1/(W_1/W_2)^{r_1/(r_1-r_2)}$。

因而，重复消费或违犯有界商业模型导致识别。仅剩的问题是可如何使其适于跟踪相同用户的其它交易。g^x不一定是公用值，其还可以是这样的情况，即仅$f(g^x)$是公用的，而g^x的知识给予解密密文的能力，该密文由可检验的加密 s 形成（例如，**Boneh** 和 **Franklin** 的密码系统具有g^x足以解密的特性）。当提取钱包时，用户将这样的密文给银行。而 s 的知识又允许发现来自该钱包的所有钱币的序号并了解它们是被如何消费的。

最后，应当将值(x, v, w)绑定到用户的身份而不是到特定的钱包。如此，即使用户试图从不同的钱包向特定的商人消费过多的钱，也仍然会导致检测和识别。

协议的详细描述：

由于该方案是对 **CHL** 电子货币方案的扩展，因此使用 **Dodis-Yampolskiy** 伪随机函数，即$f^{DY}_{(g,s)}(x) = g^{1/(s+x)}$，其中 g 是合适组的生成元；**CL** 签名及相关协议发行签名并证明签名的知识；并且随 **Camenisch-Damgard** 可检验加密技术使用双线性 **Elgamal** 密码系统。

注解：令$F_{(g,s)}(x)=f^{DY}_{(g,s)}(x)$，并且当$H$是范围为适当组的hash（散列）函数的时候，令$G^H_s(M,x)=f^{DY}_{(H(M),s)}(x)$。

现在描述系统的协议：*Setup*（设置）、*Withdraw*（提取）、*Spend*（消费）和*Deposit*（存放）（包括响应于违犯的协议）。

*Setup*协议：

令k为保密参数。共用系统参数是双线性映射参数$Bilinear_Setup(1^k)\rightarrow(q,g_1,G_1,g_2,h_2,G_2,G_T,e)$，钱包尺寸$l$，以及两个hash函数$H_1:\{0,1\}^*\rightarrow G_T$和$H_2:\{0,1\}^*\rightarrow G_1$。银行生成CL签名密钥（$pk_B$，$sk_B$）。

每个用户生成$sk_U=(x,v,w)$和$pk_U=(e(g_1,h_2)^x,e(g_1,h_2)^v,e(g_1,h_2)^w)$形式的密钥对，其中$x$，$v$，$w$是从$Z_q$随机选择的。每个用户还为任何的安全签名方案生成签名密钥对。每个商人还公布唯一的身份串id_M。此外，确定每个用户可以向商人id_M消费的钱币数的上界N_M。

*Withdraw*协议：

用户U如下从银行B提取2^l个钱币。用户和银行参与交互协议，并且如果都没有报告错误，那么在结束时：

1. U获得(s,t,σ)，其中s，t是Z_q中的随机值，并且σ是在(sk_U,s,t)上的银行签名，即(x,v,w,s,t)。
2. B获得在$e(g_1,h_2)^x$下s的可检验加密，即来自用户的公钥pk_U的第一元素连同该加密上的用户签名。
3. B没有学习关于sk_U、s或t的任何内容。

使用签名以及如“Jan Camenisch and Anna Lysyanskaya. A signature scheme with ecient protocols. In Stelvio Cimato, Clemente Galdi, and Giuseppe Persiano, editor, Security in Communication Networks'02, volumn 2576 of LNCS, pages 268-289. Springer Verlag, 2002”中所描述的相关协议可以实现步骤一。通过将Camenisch-Damgard可检验加密技术应用于双线性Elgamal密码系统可以实现步骤二。步骤三跟随在前两个之后。所有这些步骤本质上与CHL电子货币方案中的是相同的，除了签名的密钥现在除了x之外还包括v和w。

Spend 协议：

在 N_M 个钱币的消费限制的情况下，用户 U 如下向商人 M 消费一个钱币。如在 CHL 中的，用户为其钱包中消费的钱币数保持从 1 到 2^l 的专用计数器 i。另外，现在用户还为每个商人 M 保持计数器 j_M，表示该用户向那个商人消费的钱币数。

1. U 检查低于他与商人 M 的消费限制；也就是说，$j_M < N_M$。如果不是的话就异常中止。
2. M 向 U 发送随机的 $r_1, r_2 \in Z_q^*$。
3. U 在与 M 的第 j_M 次交易时向 M 发送其钱包中的第 i 个钱币。再调用 $sk_U = (x, v, w)$。该钱币包括序号 S 和钱包检查 T，其中

$$S = F_{(e(g_1,h_2),s)}(i) = e(g_1, h_2)^{1/(s+i)}, T = g_1^x (F_{(g_1,t)}(i))^{r_1} = g_1^{x+r_1/(t+i)}$$

以及两个洗钱检查值 V 和 W，其中

$$V = G_v^{H_1}(id_M, j_M) = H_1(id_M)^{1/(v+j_M)},$$

$$W = g_1^x (G_w^{H_2}(id_M, j_M))^{r_2} = g_1^x H_2(id_M)^{r_2/(w+j_M)}$$

以及零知识的、对 $(i, j_M, sk_U = (x, v, w), s, t, \sigma)$ 的知识的证明（ZKPOK）π，使得

（a）$1 \le i \le 2^l$；

（b）$1 \le j_M \le N_M$；

（c）$S = F_{(e(g_1,h_2),s)}(i)$，即 $S = e(g_1, h_2)^{1/(s+i)}$；

（d）$T = g_1^x (F_{(g_1,t)}(i))^{r_1}$，即 $T = g_1^{x+r_1/(t+i)}$；

（e）$V = G_v^{H_1}(id_M, j_M)$，即 $V = H_1(id_M)^{1/(v+j_M)}$；

（f）$W = g_1^x (G_w^{H_2}(id_M, j_M))^{r_2}$，即 $W = g_1^x H_2(id_M)^{r_2/(w+j_M)}$，以及

（g）$\mathrm{VerifySig}(pk_B, (sk_U = (x, v, w), s, t), \sigma) = \mathrm{true}$

可以使用 Fiat-Shamir 启发式方法使得证明 π 是非交互式的。

4. 如果 π 进行了检验并且之前 M 从未见过值 V_j，那么 M 接受并保存钱币 $(r_1, r_2, S, T, V, W, \pi)$。如果之前在钱币 $(r_1, r_2, S', T', V, W', \pi')$ 中曾见过值 V_j，那么 M 运行 $Open(W', W, r_2, r_2)$。如果 M 诚实地执行 *Spend* 协议（即在每个协议开始处选择新的随机值），那么在高

概率情况下$r_2 \neq r_2$，并且$Open(W', W, r_2, r_2) = g_1^x$。因而，商人可以通过计算$e(g_1^x, h_2)$来识别用户，$e(g_1^x, h_2)$是*U*的公钥的一部分。这使得诚实的商人能够自我保护来避免将试图超支的顾客。（如果商人不诚实，则银行将在存放时发觉超支。）

步骤3(a, c, d)和CHL方案中的相同，尽管步骤3(b, e, f)是新的但却类似于步骤3(a, c, d)。最后，步骤3(g)需要进行适当地修改。因此，使用标准技术可以完成步骤3(a)和3(b)。使用如“Jan Camenisch, Susan Hohenberger and Anna Lysyanskaya. Compact E-Cash. In Ronald Cramer，editor，Advances in Cryptology EUROCRYPT'05，volumn 3494 of LNCS，pages 302-321，2005”中所描述的Camenisch、Hohenberger和Lysyanskaya的技术可以完成步骤3(c)至步骤3(f)。使用Camenisch和Lysyanskaya签名可以完成步骤3(g)。

*Deposit*协议:

商人*M*通过提交钱币$(r_1, r_2, S, T, V, W, \pi)$向银行*B*存放钱币。银行检查证明$\pi$；其如果没有检验，则银行立即拒绝。现在，银行进行两个附加的检查。

首先，*B*检查钱币的消费者还没有超支其钱包；也就是说，银行搜索任何先前已接受的、具有相同序号*S*的钱币。假设找到了这样的钱币$(r_1, r_2, S, T', V', W', \pi')$。如果$r_1 = r_1$，则*B*拒绝接受该钱币。否则，*B*接受来自商人的该钱币，但是现在应当惩罚重复消费的用户。

1. *B*执行$Open(T', T, r_1, r_1) = g_1^x$。
2. *B*识别用户为具有含有$e(g_1^x, h_2)$的公钥的人。
3. *B*使用g_1^x在*Withdraw*协议期间对留给银行的*s*的加密进行解密。接下来，*B*使用*s*计算用户钱包中所有钱币从*j*=1到2^l的每个钱币的序号$S_j = F_{(e(g_1, h_2), s)}(j)$。实际上，银行可以使用$g_1^x$来对所有用户的钱包的秘密进行解密并以相同的方式跟踪那些交易。

其次，*B*检查钱币的消费者还没有超过其与商人*M*的消费限制。也就是说，银行搜索任何先前已接受的、具有相同的洗钱检查值V_j的钱币。假

设找到了这样的钱币$(r_1,r_2,S',T',V,W',\pi')$。银行拒绝接受存放物并惩罚商人。银行现在确定消费者是否应受到谴责。如果$r_2=r_2$，则 *B* 仅惩罚商人。否则，*B* 还可以惩罚试图洗钱的用户。

1. *B* 执行$Open(W',W,r_2,r_2)=g_1^x$。
2. *B* 识别用户为具有含有$e\left(g_1^x,h_2\right)$的公钥的人。
3. *B* 使用g_1^x在 *Withdraw* 协议期间对留给银行的 *s* 的加密进行解密。接下来，*B* 使用 *s* 计算用户钱包中所有钱币从 *j*=1 到 2^l 的每个钱币的序号$S_j=F_{(e(g_1,h_2),s)}(j)$。（实际上，银行可以使用$g_1^x$来对所有用户的钱包的秘密进行解密并以相同的方式跟踪那些交易。）

如果所有的检查都通过，则 *B* 接受存入 *M* 的帐户的钱币。

该存放协议（deposit protocol）类似于 CHL 方案中的存放协议，即并不是仅检查重复消费，银行现在还检查洗钱。因而，如果用户是诚实的，则银行进行两个数据库查找而不是之前的一个。

下面描述 *Open*()算法，其与 CHL 系统中的相同：

$$Open(W_1,W_2,r_1,r_2):=\frac{W_1}{(W_1/W_2)^{r_1/(r_1-r_2)}}。$$

因而，对于一些元素$h\in G_1$和$s,j\in Z_q^*$，如果$W_i=g^xF_{(h,s)}(j)^{r_i}$，则我们得到$Open(W_1,W_2,r_1,r_2)=g^x$。

下面描述违犯相关的协议如何工作。令$C_1=(r_1,r_2,S,T,V,W,\pi)$和$C_2=(r_1,r_2,S',T',V',W',\pi')$是一个现有的和一个新存放的钱币。检测重复消费或洗钱涉及分别检查$S_1=S_2$或$V_1=V_2$。识别算法在适当的输入上运行 *Open*，并且得到的犯罪证明是$\Pi=(C_1,C_2)$。检验违犯需要成功地检查钱币的有效性、检测所声称的违犯、运行 *Open* 以获得g_1^x，以及检查其与 *pk* 的关系。一个违犯的泄漏并不能用于消费用户的钱币或伪造另一个违犯。跟踪算法涉及在 *Withdraw* 期间从由用户签名的加密 *E* 恢复 *s*，并且计算所有的序号。所有权的证明$\Gamma=(E,\sigma,g_1^x)$，其中σ是用户在 *E* 上的签名。检验某个序号 *S* 的所有权涉及检验签名σ、检查$e\left(g_1^x,h_2\right)=pk$、解密 *E* 以恢复 *s*、计算所有的序号 S_i，以及测试是否对于任意 *i*，$S=S_i$。

收回（scaling back）对系统违犯者的惩罚：

为了使对系统违犯者的惩罚更宽大或仅仅使系统更有效率，有两个可用的其它选项：

选项（1）：检测违犯并揭示用户的身份。除了在 *Withdraw* 协议期间用户并不给予银行其钱包秘密 s 的可检验加密之外，该系统操作如上。然后稍后在 *Deposit* 协议期间，银行仍然可以检测违犯和识别用户，但是将不能够计算涉及该用户的其它交易的序号。

选项（2）：检测违犯。除了在 *Spend* 协议期间用户并不向商人提供洗钱检查值 Y_j 之外，该系统操作如选项（1）系统。然后稍后在 *Deposit* 协议期间，银行仍然可以检测违犯，但是将不能够识别用户。

效率考虑：

作为对协议效率的指示，规定了一些数。例如，可以构造 *Spend*，使得用户必须计算十四个多底取幂（multi-base exponentiation）以建立承诺以及另外二十个用于证明。在该例中，商人和银行需要实现二十个多底取幂以检查钱币是有效的。协议提供了用户与商人之间的两轮通信，以及银行与商人之间的一轮通信。如果采用上面的选项（2），那么就是十三个多底取幂来建立承诺以及另外十八个用于证明。银行和商人的检验采用了十八个多底取幂。

安全证明：

在有界匿名商业模型中，该方案实现了正确性、平衡、用户的匿名、对违犯者的识别、对违犯者的跟踪，以及在强 RSA、*y*-DDHI 和随机谕示模型中的 XDH 或 Sum-Free DDH 假设下的强辩解能力。

由于空间限制，提供了一些高级直觉以及证明概略。存在对 CHL 的三个主要的观察。可以将其扩展以防止重复消费以外的违犯，特别是通过谨慎使用 PRF 来将其输出链接至商人。

平衡：对于每个钱包来说，s 确定性地定义了可以是钱币的有效序号

的、精确的 N 个值。为了超支钱包，用户可以使用一个序号两次，在这种情况下，该用户是可识别的，或者该用户可以假造 CL 签名或伪造 ZK 证明。

用户的匿名：钱币包括四个值 (S,T,V,W)，其是伪随机的并且因而不泄漏有关用户的任何信息以及对有效性的（非交互式、零知识）证明，由于该证明是零知识的，因此也不泄漏任何内容。

在此仅有的异常是，当计算 V 和 W 的时候，用于 PRF 的底是商人的身份的散列（与用于计算 S 和 T 的固定底形成对比）。将散列 H 看作随机谕示，我们看到给定 $G_v^H(id_M,j)$，在任何其它输入上的 $G_v^H(\cdot,\cdot)$ 的输出，特别是对于 $id_M \neq id_M$ 的 $G_v^H(id_M,j)$，并不能与随机的区分开来。特别地，如果对于一些随机的、固定的 $H(id_M)$ 和 $H(id_M)$，对手给定的 $G_v^H(id_M,j)=f_{(H(id_M),v)}^{DY}(j)=H(id_M)^{1/(v+j)}$ 可以将 $H(id_M)^{1/(v+j)}$ 与随机的区分开来，那么其就解决了 DDH。

辨解能力：首先，诚实的用户不能被证明他没有犯下罪行，因为犯罪证明包括用户的秘密值 g_1^x。如果用户是诚实的，则仅有该用户知道该值。其次，甚至欺骗性用户也不能被证明他没有犯下罪行—例如，重复消费一个钱币并不能实现对洗钱二十个钱币的伪证—因为：（1）罪行是公开从钱币自身检验的，以及（2）x 的知识是要创建钱币的。违犯所泄漏的值 g_1^x 并不足以从该用户的钱包消费钱币。

下面基于以上概括的定义概述证明。

正确性：可以检验如果诚实方遵循协议，则双方都不会输出错误消息。

平衡：令 A 为执行与充当银行的提取器 E 的 n 个 *Withdraw* 协议的对手。在每个 *Withdraw* 期间，E 完全充当诚实的银行，除了当 A 给出对 (x,s,t,v,w) 的知识的证明时以外，E 提取这些值。从 s，E 可以以特定的概率计算 A 可以为该钱包计算的所有合法序号。令 L 表示来自所有钱包的所有序号的集合。现在假设 A 可以使诚实的银行接受具有序号 $S \notin L$ 的钱币。那么接着 A 就编造了对有效性的伪证，这以可忽略的概率发生。

用户的匿名：对匿名的 CHL 定义提供了仿真器 S，其可以在不访问用

户的钱包、密钥、或者甚至用户的公钥的情况下成功地执行用户端的*Spend*协议。假设*S*希望向商人*M*消费钱币，其中钱包尺寸是*N*，并且*M*的匿名消费限制是*M*。*S*选择随机值(x,s,t,v,w)以及任意的$1\le i\le N$和$1\le j_M\le M$。接下来，*S*完全按照诚实的用户那样创建钱币序号$S=F_{(e(g_1,h_2),s)}(i)$、商人记录定位器$V=G_v^{H_1}(id_M,j_M)$以及检查值$T=g_1^x F_{(g_1,t)}(i)^{r_1}$和$W=g_1^x G_w^{H_2}(id_M,j_M)^{r_2}$。因为这些值是伪随机的，所以它们不能与真实的钱币区分开来。*S*可以证明(S,T,V,W)形成良好并且*i*、j_M处于适当的范围。*S*可以伪造的仅有的证明部分是*S*在(x,s,t,v,w)上具有来自银行的签名。在此，*S*调用适当的CL仿真器用于该步骤，其请求控制随机谕示。

识别违犯者：如在平衡中所述，提取器*E*与对手*A*执行多个*Spend*协议，以高概率提取(x,s,t,v,w)。对于每个钱包来说，*E*可以计算对于$1\le i\le N$的所有有效的钱币序号$S_i=F_{(e(g_1,h_2),s)}(i)$，以及对于$1\le j_M\le M$和某商人*M*的所有有效的商人记录定位器$V_j=G_v^{H_1}(id_M,j_M)$。现在假设*A*可以使诚实的银行接受具有相同序号$S_1=S_2$的两个钱币$C_1=(S_1,T_1,V_1,W_1,\pi_1,r_1,r_1)$和$C_2=(S_2,T_2,V_2,W_2,\pi_2,r_2,r_2)$或者具有相同商人记录定位器$V_1=V_2$的两个钱币。由于这些冲突的发生，钱币来自相同的钱包(x,s,t,v,w)，具有相同的*i*或相同的（id_M，j_M），并且形成良好。否则，用户就伪造了对有效性的证明。然后以特定的概率，$g_1^x=T_1/(T_1/T_2)^{r_1/(r_1-r_2)}$或$g_1^x=W_1/(W_1/W_2)^{r_1/(r_1-r_2)}$揭示用户的身份。

对违犯*i*=1（重复消费）的犯罪证明Π是有效钱币(C_1,C_2)。任何人都可以检查$S_1=S_2$，$g_1^x=T_1/(T_1/T_2)^{r_1/(r_1-r_2)}$，并且$g_1^x$对应于*pk*（例如$f(g_1^x)=pk$）。同样，对违犯*i*=2（超过与特定商人的匿名商业边界）的犯罪证明是有效钱币(C_1,C_2)，其使得$V_1=V_2$，$g_1^x=W_1/(W_1/W_2)^{r_1/(r_1-r_2)}$，并且$g_1^x$对应于*pk*。

一个技术细节在于如果两个钱包种子*s*、*s'*是这样的，即$|s-s'|<2^l=N$，会发生什么；因而这两个钱包在至少一个序号上重叠。如果从两个钱包消费了相同的序号，则银行将它们标记为重复消费，然而识别算法不会起作用，因为实际上没有发生重复消费。由于这可能使银行混淆，因此这个问题可以通过使银行为每个钱包提供对*s*的选择的随机性来避免。当从Z_q随

机取出 s、s'时，它们重叠的可能性是 $2^{l+1}/q$。

跟踪违犯者：在 *Withdraw* 期间，每个用户都被要求留给银行其钱包秘密 s 的可检验加密 E，使得其自己的密钥 g_1^x 足以解密。用户还被要求对该加密签名。通过识别违犯者特性，可以推断系统违犯允许银行恢复 g_1^x。通过完整的可检验加密，使用 g_1^x 恢复钱包种子 s 并计算对于 $1 \le i \le N$ 的所有序号 $S_i = F_{(e(g_1,h_2),s)}(i)$，可以以高概率打开属于欺骗性用户的每个钱包。

对具有序号 S 的某钱币的所有权的证明 Γ 包括加密 E、解密密钥 g_1^x 以及 E 上的签名。检验方应当首先检查 g_1^x 对应于 *pk*，并且 E 上的签名有效。然后其可以解密 E 以恢复 s，计算对于 $1 \le i \le N$ 的所有序号 $S_i = F_{(e(g_1,h_2),s)}(i)$，并且测试对于某个 i，$S = S_i$。

强辩解能力：强辩解能力具有两部分。首先，没有与诚实用户交互的对手可以产生值 (i,S,V,Π) 使得 $VerifyViolation^{(i)}(params,S,V,\Pi)$ 接受，除非具有 *pk* 的用户有罪于违犯 i。

证明 Π 包括两个有效钱币 (C_1,C_2) 使得他们揭示 g_1^x。由于 g_1^x 是秘密信息，因此其释放用户已经犯下某种违犯的信号。用户对手边的（并且不是先前的某一个）特定违犯有过失的原因在于需要 x（不仅是 g_1^x）的知识来创建有效钱币。

第二部分是甚至在迫使用户违犯系统条件之后，也没有对手可以产生 (i,S,Γ) 使得 $VerifyOwnership(params,pk,S,\Gamma)$，除非具有序号 S 的钱币实际上属于具有 *pk* 的用户。证明 Γ 包括加密 E、用户的签名 σ 以及解密密钥 g_1^x。由用户签名的所有加密以高概率对应于某明文 s，该明文又允许计算属于该用户的所有序号。因而，对手将不得不在某新的加密上假造用户的签名以便取胜，这被认为以可忽略的概率发生。

尽管以上给出的详细描述涉及电子支付系统，然而关于有界匿名的类似问题可以通过本发明的方法得到处理与解决。特别地，基于此可以实现任何系统授权发放方对基于凭单的系统上不同接受方的、有限数目的匿名访问。这样的系统的例子可以包括但并不限于：打印机配额系统（qutoa system）、电子表决系统以及其它电子政府服务。

除了 CHL 系统，可以加强其它基础电子支付系统以便防止电子洗钱。在 Stefan Brands 的题为 “Untraceable Off-Line Cash in Wallets with Observers”，published in CRYPTO '93 Vol. 773 of LNCS，page 302，1994 的论文中描述了可以以类似的方式扩展的另一电子支付系统的例子。

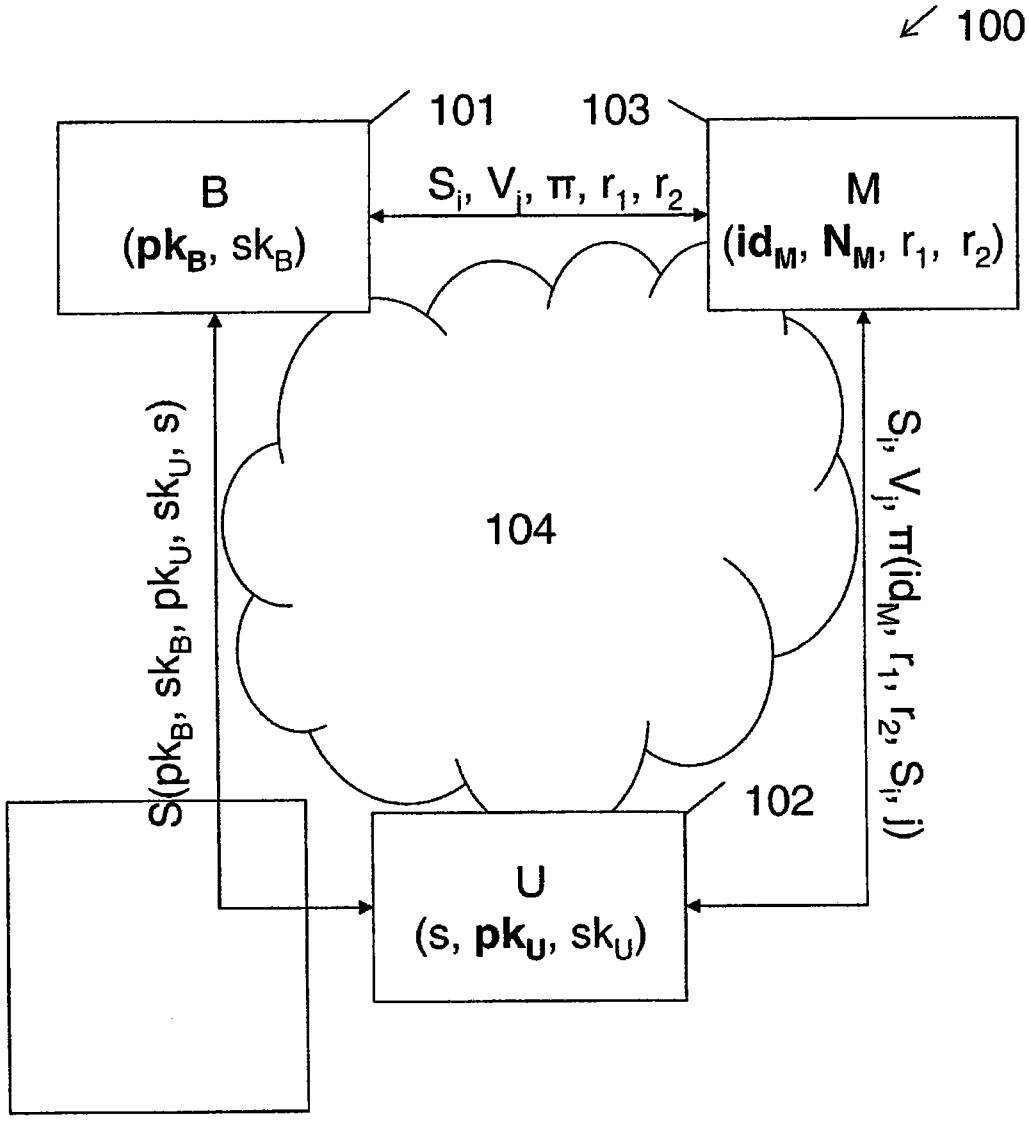

100
101
103
B
(pk_B, sk_B)
S_i, V_i, π, r_1, r_2
M
(id_M, N_M, r_1, r_2)
104
S(pk_B, sk_B, pk_U, sk_U, s)
S_i, V_j, π(id_M, r_1, r_2, S_i, j)
102
U
(s, pk_U, sk_U)

图 1

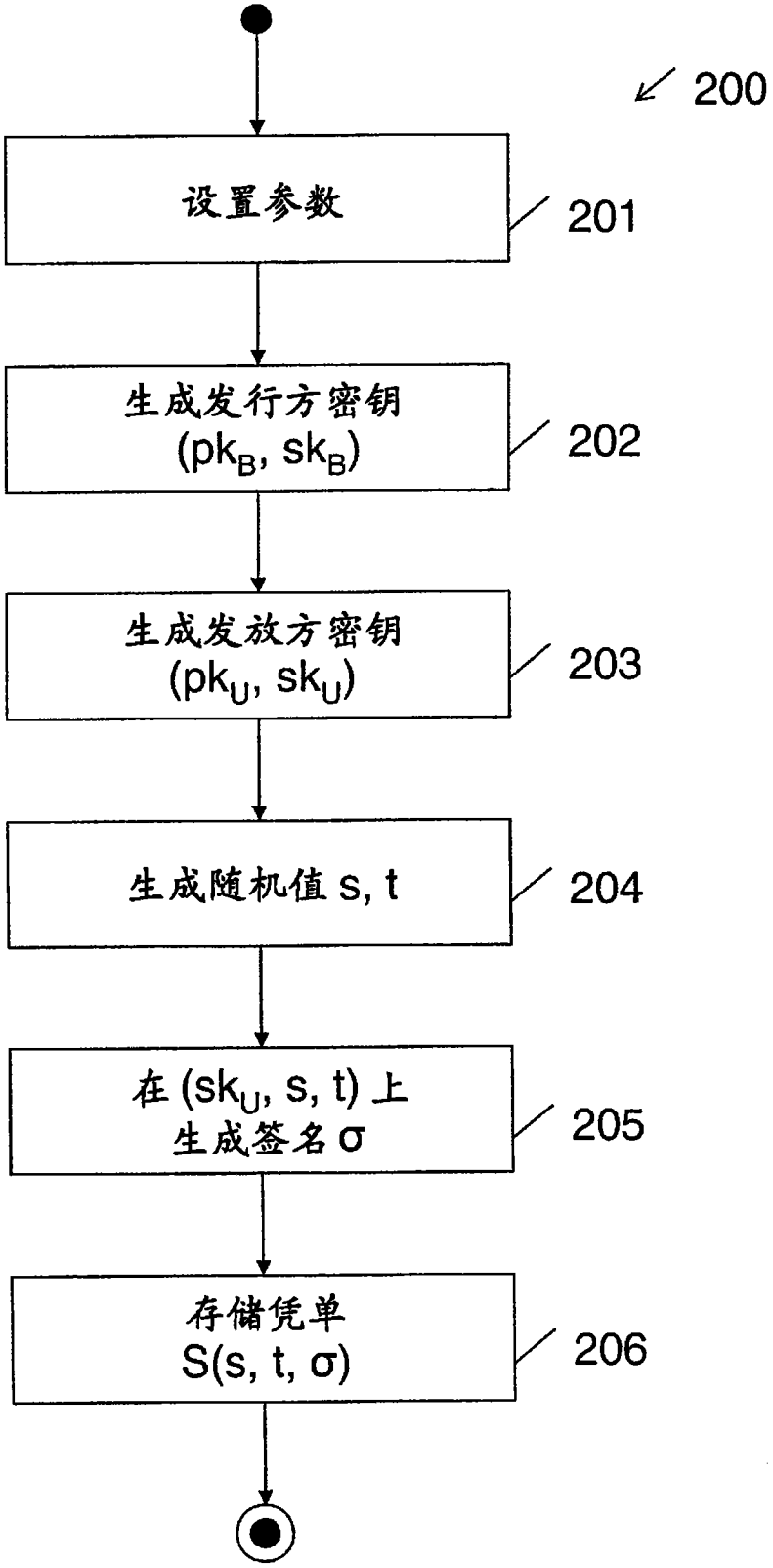

200
设置参数
201
生成发行方密钥
(pk_B, sk_B)
202
生成发放方密钥
(pk_U, sk_U)
203
生成随机值 s, t
204
在 (sk_U, s, t) 上
生成签名 σ
205
存储凭单
S(s, t, σ)
206

图 2A

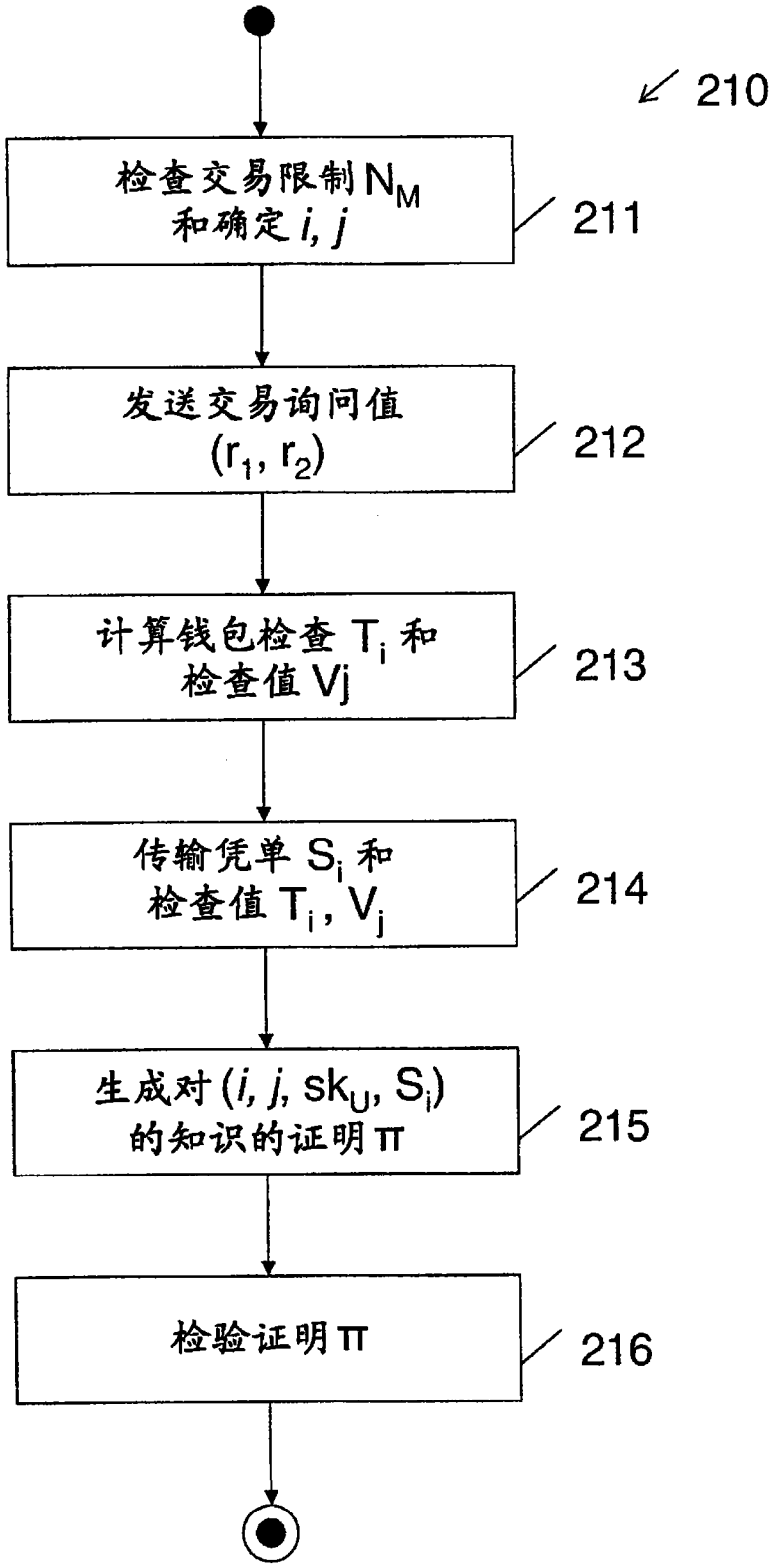

210
检查交易限制 N_M 和确定 i, j
211
发送交易询问值 (r_1, r_2)
212
计算钱包检查 T_i 和检查值 Vj
213
传输凭单 S_i 和检查值 T_i, V_j
214
生成对 (i, j, sk_U, S_i) 的知识的证明 π
215
检验证明 π
216

图 2B

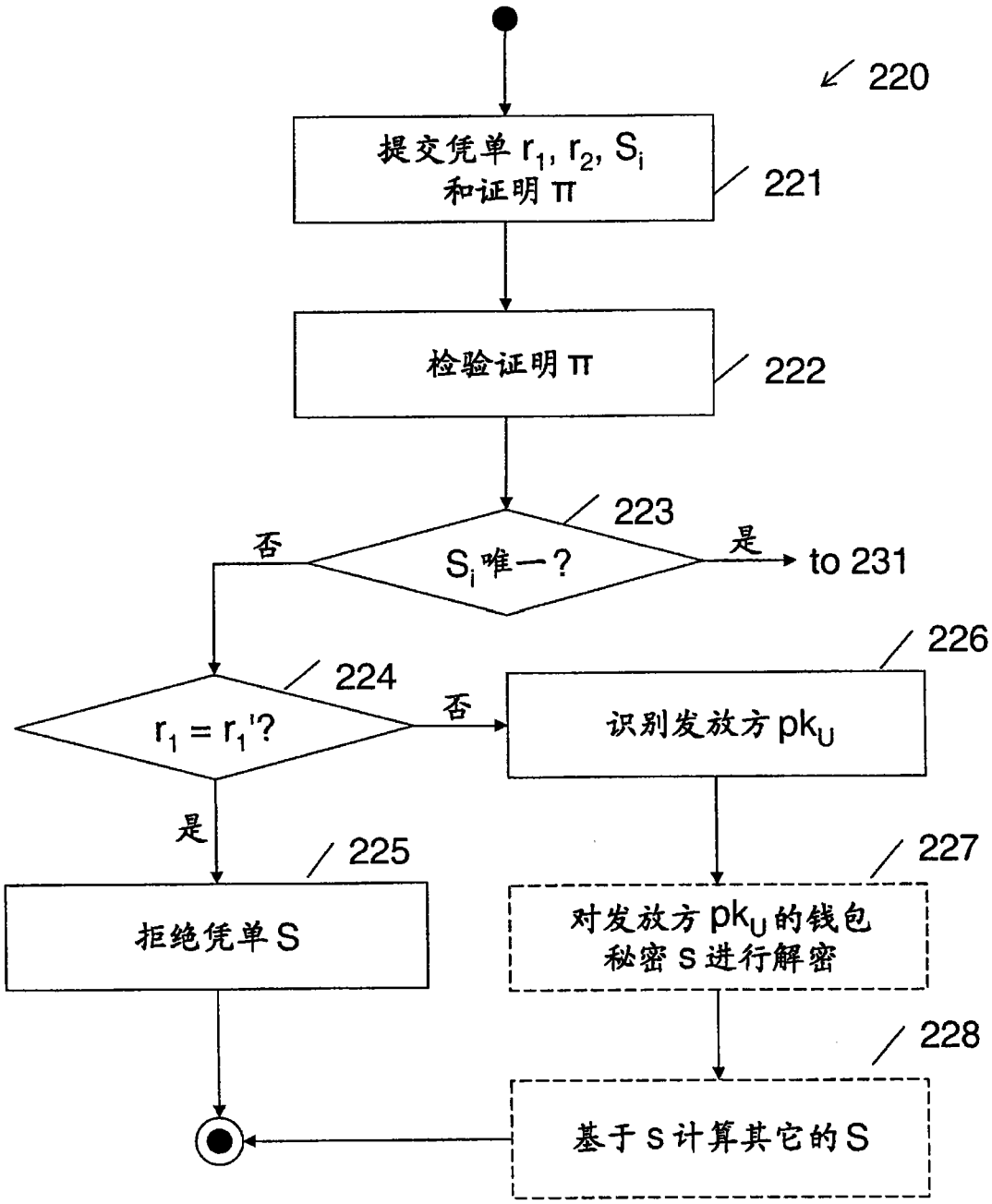

220
提交凭单 r_1, r_2, S_i 和证明 π
221
检验证明 π
222
223
S_i 唯一?
否
是
to 231
224
$r_1 = r_1'$?
否
是
226
识别发放方 pk_U
225
拒绝凭单 S
227
对发放方 pk_U 的钱包秘密 s 进行解密
228
基于 s 计算其它的 S

图 2C

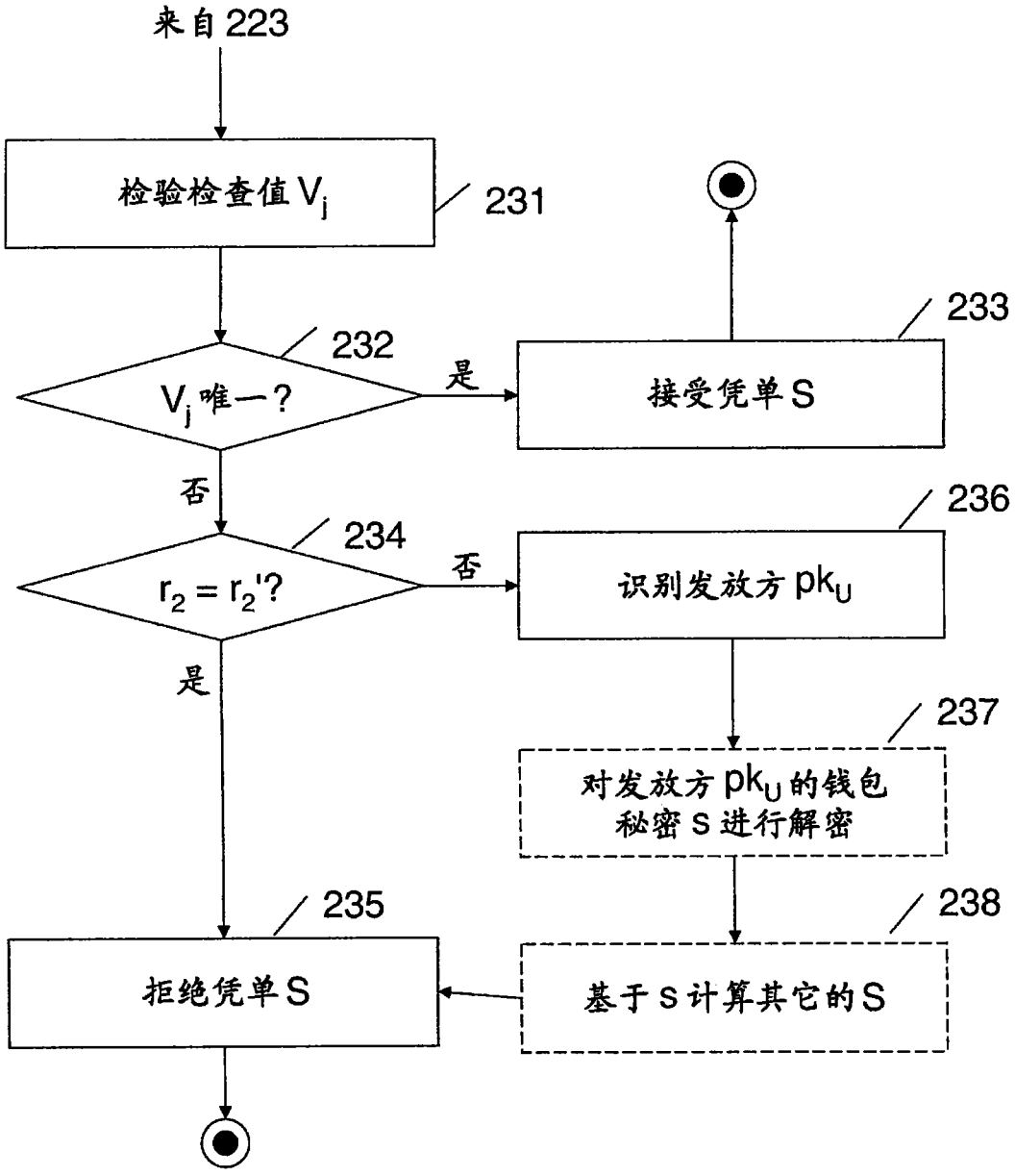

来自223
检验检查值 V_j
231
232
V_j 唯一?
是
233
接受凭单S
否
234
$r_2 = r_2'$?
否
236
识别发放方 pk_U
是
237
对发放方 pk_U 的钱包
秘密s进行解密
235
238
拒绝凭单S
基于s计算其它的S

图 2D