(12) 按照专利合作条约所公布的国际申请

(19) 世界知识产权组织
国 际 局

(43) 国际公布日
2021 年 4 月 1 日 (01.04.2021)

WIPO | PCT

(10) 国际公布号
WO 2021/057166 A1

(51) 国际专利分类号:
G06F 15/78 (2006.01) *G06F 21/62* (2013.01)
G06F 9/455 (2006.01) *G06F 21/64* (2013.01)
G06F 21/60 (2013.01)

(21) 国际申请号: PCT/CN2020/100491

(22) 国际申请日: 2020 年 7 月 6 日 (06.07.2020)

(25) 申请语言: 中文

(26) 公布语言: 中文

(30) 优先权:
201910913488.X 2019年9月25日 (25.09.2019) CN

(71) 申请人: 支付宝（杭州）信息技术有限公司 (**ALIPAY (HANGZHOU) INFORMATION TECHNOLOGY CO., LTD.**) [CN/CN]; 中国浙江省杭州市西湖区西溪路 556 号 8 层 B 段 801-11, Zhejiang 310000 (CN)。

(72) 发明人: 潘国振(**PAN, Guozhen**); 中国浙江省杭州市西湖区西溪路556号8层B段801-11, Zhejiang 310000 (CN)。 魏长征(**WEI, Changzheng**); 中国浙江省杭州市西湖区西溪路556号8层B段801-11, Zhejiang 310000 (CN)。 闫莺(**YAN, Ying**); 中国浙江省杭州市西湖区西溪路 556 号 8 层 B 段801-11, Zhejiang 310000 (CN)。

(74) 代理人: 北京博思佳知识产权代理有限公司 (**BEIJING BESTIPR INTELLECTUAL PROPERTY LAW CORPORATION**); 中国北京市海淀区上地三街 9 号嘉华大厦 B 座 409, Beijing 100085 (CN)。

(81) 指定国(除另有指明，要求每一种可提供的国家保护): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, IT, JO, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL,

(54) **Title:** METHOD AND APPARATUS FOR IMPLEMENTING EXTERNAL CALL IN FPGA

(54) **发明名称**: 在FPGA中实现外部调用的方法及装置

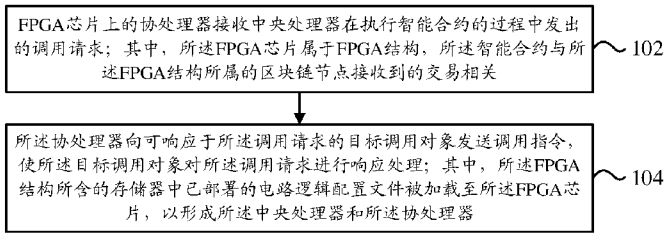

图 1

102 A coprocessor on an FPGA chip receives a call request sent by a central processing unit during the process of executing a smart contract, wherein the FPGA chip is an FPGA structure, and the smart contract is relevant to a transaction received by a blockchain node to which the FPGA structure belongs

104 The coprocessor sends a call instruction to a target call object which can respond to the call request, so that the target call object responds to the call request, wherein a circuit logic configuration file deployed in a memory comprised in the FPGA structure is loaded into the FPGA chip, so as to form the central processing unit and the coprocessor

(57) **Abstract:** One or more embodiments of the present description provide a method and apparatus for implementing an external call in an FPGA. The method may comprise: a coprocessor on an FPGA chip receives a call request sent by a central processing unit during the process of executing a smart contract, wherein the FPGA chip is an FPGA structure, and the smart contract is relevant to a transaction received by a blockchain node to which the FPGA structure belongs; and the coprocessor sends a call instruction to a target call object which can respond to the call request, so that the target call object responds to the call request, wherein a circuit logic configuration file deployed in a memory comprised in the FPGA structure is loaded into the FPGA chip, so as to form the central processing unit and the coprocessor.

(57) **摘要**: 本说明书一个或多个实施例提供一种在FPGA中实现外部调用的方法及装置，该方法可以包括：FPGA芯片上的协处理器接收中央处理器在执行智能合约的过程中发出的调用请求；其中，所述FPGA芯片属于FPGA结构，所述智能合约与所述FPGA结构所属的区块链节点接收到的交易相关；所述协处理器向可响应于所述调用请求的目标调用对象发送调用指令，使所述目标调用对象对所述调用请求进行响应处理；其中，所述FPGA结构所含的存储器中已部署的电路逻辑配置文件被加载至所述FPGA芯片，以形成所述中央处理器和所述协处理器。

WO 2021/057166 A1

[见续页]

ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, WS, ZA, ZM, ZW。

(84) **指定国**(除另有指明，要求每一种可提供的地区保护)：ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), 欧亚 (AM, AZ, BY, KG, KZ, RU, TJ, TM), 欧洲 (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG)。

本国际公布:

— 包括国际检索报告(条约第21条(3))。

在 FPGA 中实现外部调用的方法及装置

技术领域

[01] 本说明书一个或多个实施例涉及区块链技术领域，尤其涉及一种在 FPGA 中实现外部调用的方法及装置。

背景技术

[02] 区块链技术构建在传输网络（例如点对点网络）之上。传输网络中的网络节点利用链式数据结构来验证与存储数据，并采用分布式节点共识算法来生成和更新数据。

[03] 目前企业级的区块链平台技术上最大的两个挑战就是隐私和性能，往往这两个挑战很难同时解决。大多解决方案都是通过损失性能换取隐私，或者不大考虑隐私去追求性能。常见的解决隐私问题的加密技术，如同态加密（Homomorphic encryption）和零知识证明（Zero-knowledge proof）等复杂度高，通用性差，而且还可能带来严重的性能损失。

[04] 可信执行环境（Trusted Execution Environment, TEE）是另一种解决隐私问题的方式。TEE 可以起到硬件中的黑箱作用，在 TEE 中执行的代码和数据操作系统层都无法偷窥，只有代码中预先定义的接口才能对其进行操作。在效率方面，由于 TEE 的黑箱性质，在 TEE 中进行运算的是明文数据，而不是同态加密中的复杂密码学运算，计算过程效率没有损失，因此与 TEE 相结合可以在性能损失较小的前提下很大程度上提升区块链的安全性和隐私性。目前工业界十分关注 TEE 的方案，几乎所有主流的芯片和软件联盟都有自己的 TEE 解决方案，包括软件方面的 TPM（Trusted Platform Module，可信赖平台模块）以及硬件方面的 Intel SGX（Software Guard Extensions, 软件保护扩展）、ARM Trustzone（信任区）和 AMD PSP（Platform Security Processor，平台安全处理器）。

发明内容

[05] 有鉴于此，本说明书一个或多个实施例提供一种在 FPGA 中实现外部调用的方法及装置。

[06] 为实现上述目的，本说明书一个或多个实施例提供技术方案如下。

[07] 根据本说明书一个或多个实施例的第一方面，提出了一种在 FPGA 中实现外部调

用的方法，包括：FPGA 芯片上的协处理器接收中央处理器在执行智能合约的过程中发出的调用请求；其中，所述 FPGA 芯片属于 FPGA 结构，所述智能合约与所述 FPGA 结构所属的区块链节点接收到的交易相关；所述协处理器向可响应于所述调用请求的目标调用对象发送调用指令，使所述目标调用对象对所述调用请求进行响应处理；其中，所述 FPGA 结构所含的存储器中已部署的电路逻辑配置文件被加载至所述 FPGA 芯片，以形成所述中央处理器和所述协处理器。

[08] 根据本说明书一个或多个实施例的第二方面，提出了一种在 FPGA 中实现外部调用的装置，包括：接收单元，使 FPGA 芯片上的协处理器接收中央处理器在执行智能合约的过程中发出的调用请求；其中，所述 FPGA 芯片属于 FPGA 结构，所述智能合约与所述 FPGA 结构所属的区块链节点接收到的交易相关；发送单元，使所述协处理器向可响应于所述调用请求的目标调用对象发送调用指令，使所述目标调用对象对所述调用请求进行响应处理；其中，所述 FPGA 结构所含的存储器中已部署的电路逻辑配置文件被加载至所述 FPGA 芯片，以形成所述中央处理器和所述协处理器。

[09] 根据本说明书一个或多个实施例的第三方面，提出了一种电子设备，包括：处理器；

[10] 用于存储处理器可执行指令的存储器；其中，所述处理器通过运行所述可执行指令以实现如第一方面所述的方法。

[11] 根据本说明书一个或多个实施例的第四方面，提出了一种计算机可读存储介质，其上存储有计算机指令，该指令被处理器执行时实现如第一方面所述方法的步骤。

附图说明

[12] 图 1 是一示例性实施例提供的一种在 FPGA 中实现外部调用的方法的流程图。

[13] 图 2 是一示例性实施例提供的一种区块链节点的结构示意图。

[14] 图 3 是一示例性实施例提供的一种在 FPGA 芯片上形成功能模块的示意图。

[15] 图 4 是一示例性实施例提供的一种在 FPGA 中实现外部调用的装置的框图。

具体实施方式

[16] 这里将详细地对示例性实施例进行说明，其示例表示在附图中。下面的描述涉及附图时，除非另有表示，不同附图中的相同数字表示相同或相似的要素。以下示例性实

施例中所描述的实施方式并不代表与本说明书一个或多个实施例相一致的所有实施方式。相反，它们仅是与如所附权利要求书中所详述的、本说明书一个或多个实施例的一些方面相一致的装置和方法的例子。

[17] 需要说明的是：在其他实施例中并不一定按照本说明书示出和描述的顺序来执行相应方法的步骤。在一些其他实施例中，其方法所包括的步骤可以比本说明书所描述的更多或更少。此外，本说明书中所描述的单个步骤，在其他实施例中可能被分解为多个步骤进行描述；而本说明书中所描述的多个步骤，在其他实施例中也可能被合并为单个步骤进行描述。

[18] 区块链一般被划分为三种类型：公有链（Public Blockchain）、私有链（Private Blockchain）和联盟链（Consortium Blockchain）。此外，还有多种类型的结合，比如私有链+联盟链、联盟链+公有链等不同组合形式。其中去中心化程度最高的是公有链。公有链以比特币、以太坊为代表，加入公有链的参与者可以读取链上的数据记录、参与交易以及竞争新区块的记账权等。而且，各参与者（即节点）可自由加入以及退出网络，并进行相关操作。私有链则相反，该网络的写入权限由某个组织或者机构控制，数据读取权限受组织规定。简单来说，私有链可以为一个弱中心化系统，参与节点具有严格限制且少。这种类型的区块链更适合于特定机构内部使用。联盟链则是介于公有链以及私有链之间的区块链，可实现“部分去中心化”。联盟链中各个节点通常有与之相对应的实体机构或者组织；参与者通过授权加入网络并组成利益相关联盟，共同维护区块链运行。

[19] 不论是公有链、私有链还是联盟链，区块链网络中的节点出于隐私保护的目的，均可能通过区块链与 TEE（Trusted Execution Environment，可信执行环境）相结合的解决方案，在 TEE 内执行收到的交易。TEE 是基于 CPU 硬件的安全扩展，且与外部完全隔离的可信执行环境。TEE 最早是由 Global Platform 提出的概念，用于解决移动设备上资源的安全隔离，平行于操作系统为应用程序提供可信安全的执行环境。ARM 的 Trust Zone 技术最早实现了真正商用的 TEE 技术。伴随着互联网的高速发展，安全的需求越来越高，不仅限于移动设备，云端设备，数据中心都对 TEE 提出了更多的需求。TEE 的概念也得到了高速的发展和扩充。现在所说的 TEE 相比与最初提出的概念已经是更加广义的 TEE。例如，服务器芯片厂商 Intel，AMD 等都先后推出了硬件辅助的 TEE 并丰富了 TEE 的概念和特性，在工业界得到了广泛的认可。现在提起的 TEE 通常更多指这类硬件辅助的 TEE 技术。

[20] 以 Intel SGX 技术为例，SGX 提供了围圈（enclave，也称为飞地），即内存中一个加密的可信执行区域，由 CPU 保护数据不被窃取。以第一区块链节点采用支持 SGX 的 CPU 为例，利用新增的处理器指令，在内存中可以分配一部分区域 EPC（Enclave Page Cache，围圈页面缓存或飞地页面缓存），通过 CPU 内的加密引擎 MEE（Memory Encryption Engine）对其中的数据进行加密。EPC 中加密的内容只有进入 CPU 后才会被解密成明文。因此，在 SGX 中，用户可以不信任操作系统、VMM（Virtual Machine Monitor，虚拟机监控器）、甚至 BIOS（Basic Input Output System，基本输入输出系统），只需要信任 CPU 便能确保隐私数据不会泄漏。因此，围圈就相当于 SGX 技术下产生的 TEE。

[21] 不同于移动端，云端访问需要远程访问，终端用户对硬件平台不可见，因此使用 TEE 的第一步就是要确认 TEE 的真实可信。例如，相关技术中提供了针对上述 SGX 技术的远程证明机制，以用于证明目标设备上的 SGX 平台与挑战方部署了相同的配置文件。但是，由于相关技术中的 TEE 技术是以软件或软硬件结合的方式实现，使得即便通过远程证明方式可以在一定程度上表明 TEE 内所部署的配置文件未经篡改，但是 TEE 本身所依托的运行环境却无法被验证。例如，在需要实现隐私功能的区块链节点上，TEE 内需要配置用于执行智能合约的虚拟机，该虚拟机所执行的指令并非直接执行，而是实际上执行了对应的若干条 X86 指令（假定目标设备采用 X86 架构），从而造成了一定程度上的安全性风险。

[22] 为此，本说明书提出了一种基于 FPGA 实现的硬件 TEE 技术，FPGA 通过加载电路逻辑配置文件而实现硬件 TEE。由于电路逻辑配置文件的内容可以被预先查看与检验，并且 FPGA 完全基于电路逻辑配置文件中记载的逻辑而配置运行，因而可以确保 FPGA 所实现的硬件 TEE 具有相对更高的安全性。其中，电路逻辑配置文件可以在 FPGA 上形成用于执行智能合约的代码程序的中央处理器，以满足对智能合约的运行。中央处理器在执行代码程序的过程中，出于提升运行效率或数据交互等方面的目的，需要实现外部调用。因此，本说明书中通过在 FPGA 上配置形成协处理器，可以提升运行效率。

[23] 以下结合实施例说明本说明书提供的一种在 FPGA 中实现外部调用的方法，以满足中央处理器的外部调用需求。

[24] 图 1 是一示例性实施例提供的一种在 FPGA 中实现外部调用的方法的流程图。如图 1 所示，该方法应用于 FPGA 结构，可以包括以下步骤 102 和步骤 104。

[25] 步骤 102，FPGA 芯片上的协处理器接收中央处理器在执行智能合约的过程中发出

的调用请求；其中，所述 FPGA 芯片属于 FPGA 结构，所述智能合约与所述 FPGA 结构所属的区块链节点接收到的交易相关。

[26] 通过对 FPGA 芯片进行配置，可以在 FPGA 芯片上形成中央处理器，该中央处理器用于实现虚拟机逻辑，相当于在 FPGA 芯片上配置形成的“硬件虚拟机”，譬如该虚拟机逻辑可以包括以太坊虚拟机的执行逻辑或者 WASM 虚拟机的执行逻辑等，本说明书并不对此进行限制。

[27] 在智能合约的合约代码为字节码（Byte-code）程序的情况下，中央处理器可以为字节码指令集 CPU。字节码由一连串的字节组成，每一字节可以标识一个操作。基于开发效率、可读性等多方面考虑，开发者可以不直接书写字节码程序，而是选择一门高级语言编写智能合约的代码程序。高级语言编写的代码程序经过编译器编译，可以生成相应的字节码程序，进而该字节码程序可以部署至区块链。以太坊支持的高级语言很多，如 Solidity、Serpent、LLL 语言等。上述的编译器可以部署在客户端上，使得该客户端可以通过该编译器将采用高级语言编写的代码程序编译为字节码程序后，通过交易提交至区块链网络；或者，上述的编译器可以部署在区块链节点处，使得区块链节点在收到客户端提交的交易后，通过编译器将采用高级语言编写的代码程序编译为字节码程序。

[28] 以 Solidity 语言为例，用其编写的合约与面向对象编程语言中的类（Class）很相似，在一个合约中可以声明多种成员，包括合约状态（或称状态变量）、函数、函数修改器、事件等。合约状态是永久存储在智能合约的账户存储中的值，用于保存合约的状态。

[29] 如下是以 Solidity 语言编写的一个简单的智能合约的代码示例：

```
Contract Example{
    int balance;
    function C( ){
        balance + = 1;
}
function getbalance( ) returns(int){
    return balance ;
}
    }
```

[30] 对于上述代码示例中的C( )函数部分，编译器的编译结果例如为如下所示（/*...*/中...的部分为注释，后面如有汉字则为对应的中文注释）：

/* compile function C( ) balance+ = 1 编译函数 C( ) balance+ =1*/

tag_2

/* pushes 1 onto stack 将 1 压入栈顶，这个 1 就是要赋值的 1 */

0x1

/* pushes 0 onto stack 将 0 压入栈顶，这个 0 是指 balance 这个数据将要存储到合约账户数据存储的 0 号位置。上面这两句执行完后，堆栈里从顶往下，就有了 0 和 1 两个数据*/

0x0

/* balance += 1 将 balance 赋值为 balance+1 后的值*/

dup2 /*复制栈中从顶往下数的第二项，所以这时堆栈从顶往上就有了 1、0、1 三个数据*/

swap1 /*交换栈顶的两项数据，这时堆栈从顶往下存储的是 0、1、1*/

/* store(0x0, 0x1) 存储(0x0, 0x1)，从栈顶往下数，将第二项数据存储到第一项标识的位置上，同时将这两项弹出堆栈。这里便是将数据 1 存储到 0 号位置，前面因为已经将 balance 与 0 号位置做了绑定，所以就完成了 balance =1 的赋值。这时堆栈里就只剩一层数据：1 */

sstore

pop /*丢弃栈顶数据，这时堆栈变成空，等待下一条指令的执行*/

[31] 可见，上述代码示例中的 Solidity 代码被编译为相应的字节码程序，该字节码程序所含的每一字节码包括一个字节长度的操作码（Opcode）及跟随在后的零至多个操作数（Operands），该操作数为相应操作码在执行时所需的参数。

[32] 在相关技术中，当上述字节码程序运行于区块链节点上的虚拟机时，譬如该区块链节点采用 X86 架构，那么该区块链节点将字节码程序在虚拟机内运行时，实际上是通过 X86 指令集来模拟实现字节码程序所含的各个字节码。而在本说明书的技术方案中，通过在 FPGA 芯片上配置形成上述的字节码指令集 CPU，使得该字节码指令集 CPU 在执行字节码程序的过程中，直接采用字节码指令集中的字节码指令执行字节码程序所含

的各个字节码，而无需通过其他指令集来模拟执行字节码程序，从而具有相对更高的处理效率。

[33] 上述的字节码指令集 CPU 维护有字节码指令集，该字节码指令集中可以包含预定义的任意类型的字节码指令。例如，add 指令用于实现加运算，sub 指令用于实现减运算，mul 指令用于实现乘运算，div 指令用于实现除运算，or 指令用于实现按位或运算，and 指令用于实现按位与运算，xor 指令用于实现按位异或运算等，本说明书并不对此进行限制。

[34] 当然，本说明书中的中央处理器可以并非字节码指令集 CPU，譬如该中央处理器可以采用其他形式的指令集，并通过该其他形式的指令集模拟执行字节码程序所含的各个字节码，同样可以实现对智能合约的执行。

[35] 上述交易可以用于部署或调用智能合约。如果该交易用于部署智能合约，那么交易内容的 data 字段会包含该智能合约的代码程序；若代码程序基于高级语言编写，则 FPGA 结构还可以通过已部署的电路逻辑配置文件在 FPGA 芯片上形成编译器，并通过该编译器将代码程序编译为字节码程序。如果该交易用于调用智能合约，那么交易内容的 to 字段会包含被调用的智能合约的合约地址，而 FPGA 结构可以基于该合约地址调用相应已部署的字节码程序。

[36] 步骤 104，所述协处理器向可响应于所述调用请求的目标调用对象发送调用指令，使所述目标调用对象对所述调用请求进行响应处理；其中，所述 FPGA 结构所含的存储器中已部署的电路逻辑配置文件被加载至所述 FPGA 芯片，以形成所述中央处理器和所述协处理器。

[37] FPGA 芯片上包含若干可编辑的硬件逻辑单元，这些硬件逻辑单元经由电路逻辑配置文件进行配置后，可以实现为相应的功能模块，以用于实现相应的逻辑功能。具体的，该电路逻辑配置文件可以基于比特流的形式被烧录至 FPGA 结构。因此，通过向 FPGA 结构部署相应的电路逻辑配置文件，可使 FPGA 芯片上形成如前所述的中央处理器和协处理器。

[38] 协处理器（coprocessor）用于向中央处理器提供协助，以完成中央处理器无法完成的处理任务，或者中央处理器虽然能够完成但执行效率低下、效果不佳的处理任务等。协处理器通常独立于中央处理器而存在。协处理器通过接收中央处理器的相关请求，以实现针对中央处理器的协助操作。例如，如果中央处理器向协处理器发送针对目标调用

对象的调用请求，那么协处理器可以向该目标调用对象发送调用指令，以使得该目标调用对象可以满足中央处理器的调用需求。

[39] 上述的调用请求可以包括 local call，即该调用请求为本地调用请求，相应的目标调用对象为 FPGA 结构上的本地对象；或，上述的调用请求可以包括 remote call，即该调用请求为远程调用请求，相应的目标调用对象为与 FPGA 结构相连的外部对象，或者相应的目标调用对象同时包含外部对象和本地对象。可见，目标调用对象的数量可能很多，通过设置协处理器可以避免由中央处理器直接与各个目标调用对象进行对接，中央处理器只需对接协处理器、而由协处理器与各个目标调用对象进行对接，不仅可以实现更高的对接效率，而且极大地节省了中央处理器在外部对接方面所需消耗的资源，以用于提升对智能合约的执行效率。

[40] 当目标调用对象包括本地对象时，该本地对象可以包括通过加载上述电路逻辑配置文件而在 FPGA 芯片上形成的预设功能模块。例如，该预设功能模块可以包括以下至少之一：加密模块，用于对所述中央处理器产生的明文数据进行加密；解密模块，用于对外部传入所述 FPGA 结构的密文数据进行解密；计算模块，用于执行所述智能合约涉及的计算操作；缓存模块，用于存储所述智能合约产生的合约状态等，本说明书并不对此进行限制。

[41] 例如，当交易发起方存在隐私保护需求时，可以向区块链提交经过加密的隐私交易，使得区块链节点向 FPGA 结构传入该隐私交易。隐私交易被传入中央处理器后，中央处理器可以通过向协处理器发起 local call，以针对解密模块进行调用，那么协处理器可以将隐私交易传输至解密模块，并将解密模块进行解密得到的明文交易内容返回至中央处理器。如前所述，当该隐私交易用于部署智能合约时，中央处理器可从收到的明文交易内容的 data 字段获得该智能合约的合约代码；当该隐私交易用于调用智能合约时，中央处理器可以从收到的明文交易内容的 to 字段获得该智能合约的合约地址，进而基于该合约地址获得相应的合约代码：如果合约代码被部署于上述的缓存模块中，则中央处理器可以通过向协处理器发起 local call，以使得协处理器协助从缓存模块中调取相应的合约代码，如果合约代码被部署于区块链节点或者与 FPGA 芯片相连的外部存储设备处，则涉及到目标调用对象为外部对象的情况，下文将对此详述。

[42] 中央处理器在运行合约代码的过程中，可能需要调用计算模块进行协助处理。例如，当合约代码被表征为 wasm 字节码程序时，由于 wasm 字节码规定了数据长度为 32b 或 64b，因而在针对一些长度较大的数据进行处理时，往往会影响其处理效率。譬如在

针对长度为 1024b 的数据进行比较时，若直接由中央处理器进行处理，需要将该 1024b 长度的数据划分为 32 个 32b 或 16 个 64b 长度的数据段，然后分别对各个数据段进行比较，需要比较 32 次或 16 次；而如果采用 FPGA 芯片上额外配置的计算模块，则可以不受 wasm 字节码的限制，譬如可以将数据最大长度设定为 1024b 或更大，那么只需一次即可完成对 1024b 长度数据的比较操作，极大地提升了处理效率。类似地，在实现诸如大数乘法等运算过程中，对于计算模块的调用均可以极大地提升对合约代码的执行效率。

[43] 中央处理器运行合约代码后，可能使得合约代码所涉及的至少一部分合约状态发生取值变化。如果合约状态被维护于 FPGA 芯片上，比如上述的缓存模块，那么中央处理器可以向协处理器发起 local call，使得协处理器将相关合约状态的更新后取值存储至上述的缓存模块；如果合约状态被维护于前述的区块链节点或外部存储设备处，则涉及到目标调用对象为外部对象的情况，下文将对此详述。类似地，中央处理器可以向协处理器发起 local call，从而对合约代码执行产生的交易收据进行处理。

[44] 由于 FPGA 芯片内部被认为属于安全范围、FPGA 芯片外部被认为属于非安全范围，因此当 FPGA 结构需要将诸如合约代码、合约状态、交易收据等从 FPGA 芯片向外传输时，譬如传输至区块链节点或外部存储设备，中央处理器需要通过向协处理器发起 local call，使得协处理器可以将上述的合约代码、合约状态、交易收据等传输至前述的加密模块进行加密处理，确保向区块链节点或外部存储设备传输相应的密文合约代码、密文合约状态、密文交易收据等。

[45] 当目标调用对象包括外部对象时，该外部对象可以包括：外部存储设备或区块链节点（实际为区块链节点包含的节点主机），而中央处理器所发起的调用请求用于与该外部存储设备或区块链节点进行数据读写。

[46] 例如，FPGA 结构在收到用于部署智能合约的交易后，针对从该交易中提取出的合约代码，可以通过向协处理器发起 remote call，以使得协处理器协助向合约代码发送至上述的外部存储设备或区块链节点进行部署。协处理器与区块链节点之间可以通过建立 DMA（Direct Memory Access，直接内存存取）连接以实现数据交互。

[47] 如前所述，FPGA 结构在收到用于调用智能合约的交易后，如果相应的合约代码被部署于外部对象处，那么中央处理器可以通过向协处理器发起 remote call，以使得协处理器协助从外部对象处获得合约代码，以由中央处理器予以执行。中央处理器在执行合约代码的过程中，需要使用所涉及的合约状态的取值，并在合约代码执行过程中或执行完毕后针对其中的至少一部分取值予以更新。其中，中央处理器可以通过向协处理器发

起 remote call，以使得协处理器从外部对象处获得合约状态的取值，或者向外部对象返回更新后的合约状态的取值。此外，如前所述：从外部对象处获得的合约代码、合约状态等可能处于密文状态，中央处理器需要结合发起 local call，以由协处理器调用解密模块进行解密，以及在向外部对象返回相关数据之前，中央处理器需要结合发起 local call，以由协处理器调用加密模块进行加密。

[48] 图 2 是一示例性实施例提供的一种区块链节点的结构示意图。基于本说明书的技术方案，可以在区块链节点上添加 FPGA 结构以实现硬件 TEE，譬如该 FPGA 结构可以为如图 2 所示的 FPGA 板卡。FPGA 板卡可以通过 PCIE 接口连接至区块链节点上，以实现 FPGA 板卡与区块链节点之间的数据交互。FPGA 板卡可以包括 FPGA 芯片、Flash（闪存）芯片和密管芯片等结构；当然，在一些实施例中除了包含 FPGA 芯片之外，可能仅包含剩余的 Flash 芯片和密管芯片等中的部分结构，或者可能包含更多结构，此处仅用于举例。

[49] 在初始阶段，FPGA 芯片上并未烧录用户定义的任何逻辑，相当于 FPGA 芯片处于空白状态。用户可以通过向 FPGA 芯片上烧录电路逻辑配置文件，以在 FPGA 芯片上形成相应的功能或逻辑。在首次烧录电路逻辑配置文件时，FPGA 板卡不具有安全防护的能力，因而通常需要外部提供安全环境，比如用户可以在离线环境下实施对电路逻辑配置文件的烧录以实现物理安全隔离，而非在线上实施远程烧录。

[50] 针对用户所需实现的功能或逻辑，可以通过 FPGA 硬件语言形成相应的逻辑代码，并进而对该逻辑代码进行镜像化处理，即可得到上述的电路逻辑配置文件。在烧录至 FPGA 板卡之前，用户可以针对上述的逻辑代码进行检查。尤其是，当同时涉及到多个用户时，多个用户可以分别对上述的逻辑代码进行检查，以确保 FPGA 板卡最终能够满足所有用户的需求，防止出现安全性风险、逻辑错误、欺诈等异常问题。

[51] 在确定代码无误后，用户可以在上述的离线环境下，将电路逻辑配置文件烧录至 FPGA 板卡上。具体的，电路逻辑配置文件被从区块链节点传入 FPGA 板卡，进而部署至如图 2 所示的 Flash 芯片中，使得即便 FPGA 板卡发生掉电，Flash 芯片仍然能够保存上述的电路逻辑配置文件。

[52] 图 3 是一示例性实施例提供的一种在 FPGA 芯片上形成功能模块的示意图。通过将 Flash 芯片中所部署的电路逻辑配置文件加载至 FPGA 芯片，可以对 FPGA 芯片所含的硬件逻辑单元进行配置，从而在 FPGA 芯片上形成相应的功能模块，譬如所形成的功能模块可以包括如图 3 所示的中央处理器、协处理器、缓存模块、加解密模块、计算模

块、密钥协商模块、解密验签模块等。同时，电路逻辑配置文件还可以用于向 FPGA 板卡传输需要存储的信息，比如可以将预置证书存储于 FPGA 芯片上、将认证根密钥存储于密管芯片中（认证根密钥也可以存储于 FPGA 芯片上）等。

[53] 基于 FPGA 芯片上所形成的密钥协商模块，以及部署于 FPGA 板卡上的认证根密钥，使得 FPGA 板卡可以与用户实现远程的密钥协商，该密钥协商过程可以采用相关技术中的任意算法或标准来实现，本说明书并不对此进行限制。举例而言，密钥协商过程可以包括：用户可以在本地的客户端生成一密钥 Ka-1、密钥协商模块可以在本地生成一密钥 Kb-1，且客户端可以基于密钥 Ka-1 计算得到密钥协商信息 Ka-2、密钥协商模块可以基于密钥 Kb-1 计算得到密钥协商信息 Kb-2，然后客户端将密钥协商信息 Ka-2 发送至密钥协商模块、密钥协商模块将密钥协商信息 Kb-2 发送至客户端，使得客户端可以基于密钥 Ka-1 与密钥协商信息 Kb-2 生成一秘密值，而密钥协商模块可以基于密钥 Kb-1 与密钥协商信息 Ka-2 生成相同的秘密值，最后由客户端、密钥协商模块分别基于密钥导出函数从该相同的秘密值导出相同的配置文件部署密钥，该配置文件部署密钥可以存在 FPGA 芯片或密管芯片。在上述过程中，虽然密钥协商信息 Ka-2、密钥协商信息 Kb-2 是经由区块链节点在客户端与密钥协商模块之间传输，但是由于密钥 Ka-1 由客户端掌握、密钥 Kb-1 由密钥协商模块掌握，因而可以确保区块链节点无法获知最终得到的秘密值和配置文件部署密钥，避免可能造成的安全性风险。

[54] 除了配置文件部署密钥之外，秘密值还用于导出业务秘密部署密钥；例如，秘密值可以导出 32 位数值，可以将前 16 位作为配置文件部署密钥、后 16 位作为业务秘密部署密钥。用户可以通过业务秘密部署密钥向 FPGA 板卡部署业务密钥，譬如该业务密钥可以包括节点私钥和业务根密钥。例如，用户可以在客户端上采用业务秘密部署密钥对节点私钥或业务根密钥进行签名、加密并发送至 FPGA 板卡，使得 FPGA 板卡通过解密验签模块进行解密、验签后，对得到的节点私钥或业务根密钥进行部署。

[55] 基于部署的节点密钥、业务根密钥和 FPGA 芯片上的加解密模块、中央处理器、协处理器、计算模块等，使得 FPGA 板卡可以实现为区块链节点上的 TEE，以满足隐私需求。例如，当区块链节点收到一笔交易时，如果该交易为明文交易，区块链节点可以直接处理该明文交易，如果该交易为隐私交易，区块链节点将该隐私交易传入 FPGA 板卡进行处理。

[56] 明文交易的交易内容为明文形式，并且交易执行后所产生的合约状态等同样采用明文形式进行存储。隐私交易的交易内容为密文形式，由交易发起方对明文交易内容进

行加密而得到，且交易执行后产生的合约状态等需要采用密文形式进行存储，从而确保交易隐私保护。例如，交易发起方可以随机或基于其他方式生成一对称密钥，同样上述的业务私钥对应的业务公钥被公开，那么交易发起方可以基于该对称密钥和业务公钥对明文交易内容进行数字信封加密：交易发起方通过对称密钥加密明文交易内容，并通过业务公钥对该对称密钥进行加密，得到的两部分内容均被包含于上述的隐私交易中；换言之，隐私交易中包含两部分内容：采用对称密钥加密的明文交易内容、采用业务公钥加密的对称密钥。

[57] 因此，FPGA 板卡在收到区块链节点传入的隐私交易后，可将该隐私交易交由中央处理器进行处理。中央处理器通过向协处理器发起 local call 的调用请求，由协处理器进一步调用加解密模块，使得加解密模块通过业务私钥对采用业务公钥加密的对称密钥进行解密、得到对称密钥，然后加解密模块进一步通过对称密钥对采用对称密钥加密的明文交易内容进行解密、得到明文交易内容。

[58] 如果隐私交易用于部署智能合约，那么明文交易内容的 data 字段可以包含待部署的智能合约的合约代码。中央处理器可以对获得的合约代码进行部署。如果部署于 FPGA 芯片上的缓存模块，则中央处理器直接以明文形式将合约代码部署至缓存模块；如果部署于区块链节点（或其他外部对象，此处以区块链节点为例），则中央处理器通过向协处理器发起 local call 的调用请求，由协处理器调用加解密模块对该合约代码进行加密，进而将加密后的密文合约代码部署至区块链节点。

[59] 如果隐私交易用于调用智能合约，那么明文交易内容的 to 字段可以包含被调用的智能合约的合约地址，而 FPGA 板卡可以基于该合约地址调取相应的合约代码。如果合约代码部署于 FPGA 芯片的缓存模块，该合约代码往往采用明文形式进行存储，那么协处理器可以直接从该缓存模块中获得合约代码，并交由中央处理器进行处理。如果合约代码部署于区块链节点，协处理器从区块链节点处获得密文合约代码，那么中央处理器需要进一步向协处理器发起 local call 的调用请求，由协处理器调用加解密模块对该密文合约代码进行解密，进而由中央处理器执行解密后的合约代码。

[60] 中央处理器用于实现相关技术中的虚拟机逻辑，即中央处理器相当于 FPGA 板卡上的“硬件虚拟机”。因此，基于上述明文交易内容确定出合约代码后，可以将该合约代码传入中央处理器中，以由该中央处理器执行该合约代码。中央处理器可以为本说明书中的字节码指令集 CPU，针对区块链节点所收到的交易需要部署或调用的智能合约，FPGA 板卡可以将该智能合约的字节码程序读入字节码指令集 CPU，使得字节码指令集

CPU 直接执行该字节码程序所含的各个字节码，而无需通过其他指令集对字节码进行模拟，极大地提升了针对字节码的执行效率，从而加快了交易处理速度。

[61] 在中央处理器执行合约代码之前或过程中，需要获取该合约代码所涉及的合约状态的最近取值。合约状态的最近取值可能以明文形式部署于 FPGA 芯片上的缓存模块，那么中央处理器可以通过向协处理器发起 local call 的调用请求，由协处理器从缓存模块中调取合约状态的最近取值。合约状态的最近取值可能以密文形式部署于区块链节点（或其他外部对象）处，那么中央处理器可以通过向协处理器发起 remote call 的调用请求，由协处理器从区块链节点远程调用密文形式的合约状态的最近取值，以及中央处理器可以通过向协处理器发起 local call 的调用请求，由协处理器调用加解密模块对该密文形式的合约状态的最近取值进行解密，进而由中央处理器获得解密后的合约状态的最近取值。

[62] 执行完毕后，合约代码所涉及的合约状态可能发生更新。如果合约状态的更新取值需要部署于 FPGA 芯片上的缓存模块，那么中央处理器可以通过向协处理器发起 local call 的调用请求，由协处理器直接将合约状态的更新取值存入缓存模块。如果合约状态的更新取值需要部署于区块链节点处，那么中央处理器可以通过向协处理器发起 local call 的调用请求，由协处理器调用加解密模块对合约状态的更新取值进行加密，以及中央处理器通过向协处理器发起 remote call 的调用请求，由协处理器向区块链节点远程存入密文形式的合约状态的更新取值。

[63] 基于一些原因，用户可能希望对 FPGA 板卡上部署的电路逻辑配置文件进行版本更新，比如该电路逻辑配置文件所含的认证根密钥可能被风险用户获知、再比如用户希望对 FPGA 板卡上部署的功能模块进行升级等，本说明书并不对此进行限制。为了便于区分，可以将上述过程中已部署的电路逻辑配置文件称之为旧版电路逻辑配置文件，而将需要部署的电路逻辑配置文件称之为新版电路逻辑配置文件。

[64] 与旧版电路逻辑配置文件相类似的，用户可以通过编写代码、镜像化等过程生成新版电路逻辑配置文件。进一步的，用户可以通过自身持有的私钥对新版电路逻辑配置文件进行签名，然后通过上文协商出的配置文件部署密钥对签名后的新版电路逻辑配置文件进行加密，得到加密后新版电路逻辑配置文件。在一些情况下，可能同时存在多名用户，那么旧版电路逻辑配置文件需要将这些用户对应的预置证书均部署至 FPGA 板卡中，且这些用户需要分别采用自身持有的私钥对新版电路逻辑配置文件进行签名。

[65] 用户可以通过客户端远程将加密后新版电路逻辑配置文件发送至区块链节点，并

由区块链节点进一步将其传入 FPGA 板卡。前述过程中在 FPGA 芯片上形成的解密验签模块位于 PCIE 接口与 Flash 芯片之间的传输通路上，使得加密后新版电路逻辑配置文件必然需要优先经过解密验签模块的成功处理后，才能够被传入 Flash 芯片以实现可信更新，无法绕过解密验签的过程而直接对 Flash 芯片进行更新。

[66] 解密验签模块在收到加密后新版电路逻辑配置文件后，首先通过 FPGA 板卡上部署的配置文件部署密钥进行解密，如果解密成功则解密验签模块进一步基于 FPGA 芯片上部署的预置证书，对解密后的新版电路逻辑配置文件进行签名验证。如果解密失败或者签名验证未通过，则说明收到的文件并非来自上述用户或者遭到篡改，解密验签模块将触发终止本次的更新操作；而在解密成功且验签通过的情况下，可以确定得到的新版电路逻辑配置文件来自上述用户且传输过程中未遭到篡改，可以将该新版电路逻辑配置文件进一步传输至 Flash 芯片，以针对 Flash 芯片中的旧版电路逻辑配置文件进行更新部署。

[67] 新版电路逻辑配置文件被加载至 FPGA 芯片后，同样可以在该 FPGA 芯片上形成诸如上述的密钥协商模块、解密验签模块，以及向 FPGA 芯片存入预置证书、向密管芯片存入认证根密钥等信息。其中，所形成的密钥协商模块、解密验签模块等，所实现的功能逻辑可以发生变化和升级，所存入部署的预置证书、认证根密钥等信息也可能区别于更新前的信息。那么，FPGA 板卡可以基于更新后的密钥协商模块、认证根密钥等，与用户进行远程协商得到新的配置文件部署密钥，该配置文件部署密钥可以被用于下一次的可新更新过程。类似地，可以据此不断实现针对 FPGA 板卡的可信更新操作。

[68] 在完成更新部署后，FPGA 板卡可以针对新版电路逻辑配置文件生成认证结果。例如，上述的密钥协商模块可以通过诸如 sm3 算法或其他算法对新版电路逻辑配置文件的哈希值、基于新版电路逻辑配置文件协商得到的配置文件部署密钥的哈希值进行计算，得到的计算结果可以被作为上述的认证结果，并由密钥协商模块将该认证结果发送至用户。相应地，用户可以在客户端上基于所维护的新版电路逻辑配置文件和据此协商的配置文件部署密钥对认证结果进行验证，如果验证成功则表明新版电路逻辑配置文件在 FPGA 板卡上成功部署，且用户与 FPGA 板卡之间据此成功协商得到了一致的配置文件部署密钥，从而确认成功完成了针对电路逻辑配置文件的更新部署。

[69] 图 4 是一示例性实施例提供的一种在 FPGA 中实现外部调用的装置的示意结构图。请参考图 4，在软件实施方式中，该在 FPGA 中实现外部调用的装置可以包括接收单元 401 和发送单元 402。

[70] 接收单元 401，使 FPGA 芯片上的协处理器接收中央处理器在执行智能合约的过程中发出的调用请求；其中，所述 FPGA 芯片属于 FPGA 结构，所述智能合约与所述 FPGA 结构所属的区块链节点接收到的交易相关。

[71] 发送单元 402，使所述协处理器向可响应于所述调用请求的目标调用对象发送调用指令，使所述目标调用对象对所述调用请求进行响应处理；其中，所述 FPGA 结构所含的存储器中已部署的电路逻辑配置文件被加载至所述 FPGA 芯片，以形成所述中央处理器和所述协处理器。

[72] 可选的，当所述调用请求为本地调用请求时，所述目标调用对象为所述 FPGA 结构上的本地对象；或，当所述调用请求为远程调用请求时，所述目标调用对象为与所述 FPGA 结构相连的外部对象，或者所述目标调用对象为所述外部对象和所述本地对象。

[73] 可选的，所述本地对象包括：通过加载所述电路逻辑配置文件而在所述 FPGA 芯片上形成的预设功能模块。

[74] 可选的，所述预设功能模块包括以下至少之一：加密模块，用于对所述中央处理器产生的明文数据进行加密；解密模块，用于对外部传入所述 FPGA 结构的密文数据进行解密；计算模块，用于执行所述智能合约涉及的计算操作；缓存模块，用于存储所述智能合约产生的合约状态。

[75] 可选的，所述外部对象包括：外部存储设备或所述区块链节点包含的节点主机。

[76] 可选的，所述调用请求用于与所述外部存储设备或所述节点主机进行数据读写。

[77] 可选的，所述交易用于部署或调用所述智能合约。

[78] 可选的，所述智能合约的合约代码为字节码程序，且所述中央处理器为字节码指令集 CPU。

[79] 可选的，所述中央处理器用于实现虚拟机逻辑。

[80] 可选的，所述虚拟机逻辑包括：以太坊虚拟机的执行逻辑或者 WASM 虚拟机的执行逻辑。

[81] 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，计算机的具体形式可以是个人计算机、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、电子邮件收发设备、游戏控制台、平板计算机、可穿戴设备或

者这些设备中的任意几种设备的组合。

[82] 在一个典型的配置中，计算机包括一个或多个处理器 (CPU)、输入/输出接口、网络接口和内存。

[83] 内存可能包括计算机可读介质中的非永久性存储器，随机存取存储器（RAM）和/或非易失性内存等形式，如只读存储器（ROM）或闪存（flash RAM）。内存是计算机可读介质的示例。

[84] 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、磁盘存储、量子存储器、基于石墨烯的存储介质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。

[85] 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。

[86] 上述对本说明书特定实施例进行了描述。其它实施例在所附权利要求书的范围内。在一些情况下，在权利要求书中记载的动作或步骤可以按照不同于实施例中的顺序来执行并且仍然可以实现期望的结果。另外，在附图中描绘的过程不一定要求示出的特定顺序或者连续顺序才能实现期望的结果。在某些实施方式中，多任务处理和并行处理也是可以的或者可能是有利的。

[87] 在本说明书一个或多个实施例使用的术语是仅仅出于描述特定实施例的目的，而非旨在限制本说明书一个或多个实施例。在本说明书一个或多个实施例和所附权利要求书中所使用的单数形式的“一种”、“所述”和“该”也旨在包括多数形式，除非上下

文清楚地表示其他含义。还应当理解，本文中使用的术语“和/或”是指并包含一个或多个相关联的列出项目的任何或所有可能组合。

[88] 应当理解，尽管在本说明书一个或多个实施例可能采用术语第一、第二、第三等来描述各种信息，但这些信息不应限于这些术语。这些术语仅用来将同一类型的信息彼此区分开。例如，在不脱离本说明书一个或多个实施例范围的情况下，第一信息也可以被称为第二信息，类似地，第二信息也可以被称为第一信息。取决于语境，如在此所使用的词语“如果”可以被解释成为“在……时”或“当……时”或“响应于确定”。

[89] 以上所述仅为本说明书一个或多个实施例的较佳实施例而已，并不用以限制本说明书一个或多个实施例，凡在本说明书一个或多个实施例的精神和原则之内，所做的任何修改、等同替换、改进等，均应包含在本说明书一个或多个实施例保护的范围之内。

权利要求书

1、一种在 FPGA 中实现外部调用的方法，包括:

FPGA 芯片上的协处理器接收中央处理器在执行智能合约的过程中发出的调用请求;其中，所述 FPGA 芯片属于 FPGA 结构，所述智能合约与所述 FPGA 结构所属的区块链节点接收到的交易相关;

所述协处理器向可响应于所述调用请求的目标调用对象发送调用指令，使所述目标调用对象对所述调用请求进行响应处理;其中，所述 FPGA 结构所含的存储器中已部署的电路逻辑配置文件被加载至所述 FPGA 芯片，以形成所述中央处理器和所述协处理器。

2、根据权利要求 1 所述的方法，

当所述调用请求为本地调用请求时，所述目标调用对象为所述 FPGA 结构上的本地对象;或，

当所述调用请求为远程调用请求时，所述目标调用对象为与所述 FPGA 结构相连的外部对象，或者所述目标调用对象为所述外部对象和所述本地对象。

3、根据权利要求 2 所述的方法，所述本地对象包括:通过加载所述电路逻辑配置文件而在所述 FPGA 芯片上形成的预设功能模块。

4、根据权利要求 3 所述的方法，所述预设功能模块包括以下至少之一:

加密模块，用于对所述中央处理器产生的明文数据进行加密;

解密模块，用于对外部传入所述 FPGA 结构的密文数据进行解密;

计算模块，用于执行所述智能合约涉及的计算操作;

缓存模块，用于存储所述智能合约产生的合约状态。

5、根据权利要求 2 所述的方法，所述外部对象包括:外部存储设备或所述区块链节点包含的节点主机。

6、根据权利要求 5 所述的方法，所述调用请求用于与所述外部存储设备或所述节点主机进行数据读写。

7、根据权利要求 1 所述的方法，所述交易用于部署或调用所述智能合约。

8、根据权利要求 1 所述的方法，所述智能合约的合约代码为字节码程序，且所述中央处理器为字节码指令集 CPU。

9、根据权利要求 1 所述的方法，所述中央处理器用于实现虚拟机逻辑。

10、根据权利要求 9 所述的方法，所述虚拟机逻辑包括:以太坊虚拟机的执行逻辑或者 WASM 虚拟机的执行逻辑。

11、一种在 FPGA 中实现外部调用的装置，包括:

接收单元，使 FPGA 芯片上的协处理器接收中央处理器在执行智能合约的过程中发出的调用请求；其中，所述 FPGA 芯片属于 FPGA 结构，所述智能合约与所述 FPGA 结构所属的区块链节点接收到的交易相关；

发送单元，使所述协处理器向可响应于所述调用请求的目标调用对象发送调用指令，使所述目标调用对象对所述调用请求进行响应处理；其中，所述 FPGA 结构所含的存储器中已部署的电路逻辑配置文件被加载至所述 FPGA 芯片，以形成所述中央处理器和所述协处理器。

12、一种电子设备，包括：

处理器；

用于存储处理器可执行指令的存储器；

其中，所述处理器通过运行所述可执行指令以实现如权利要求 1-10 中任一项所述的方法。

13、一种计算机可读存储介质，其上存储有计算机指令，该指令被处理器执行时实现如权利要求 1-10 中任一项所述的方法。

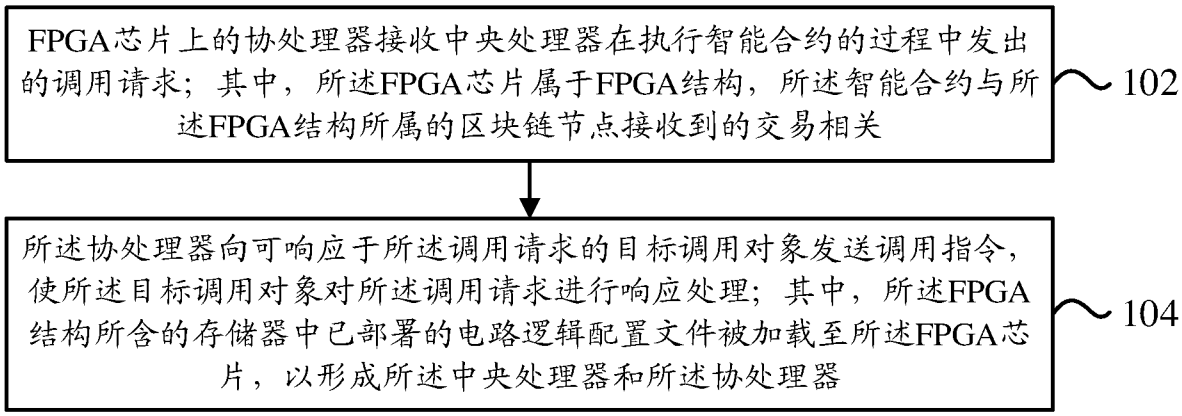
FPGA芯片上的协处理器接收中央处理器在执行智能合约的过程中发出的调用请求；其中，所述FPGA芯片属于FPGA结构，所述智能合约与所述FPGA结构所属的区块链节点接收到的交易相关
102
所述协处理器向可响应于所述调用请求的目标调用对象发送调用指令，使所述目标调用对象对所述调用请求进行响应处理；其中，所述FPGA结构所含的存储器中已部署的电路逻辑配置文件被加载至所述FPGA芯片，以形成所述中央处理器和所述协处理器
104

图 1

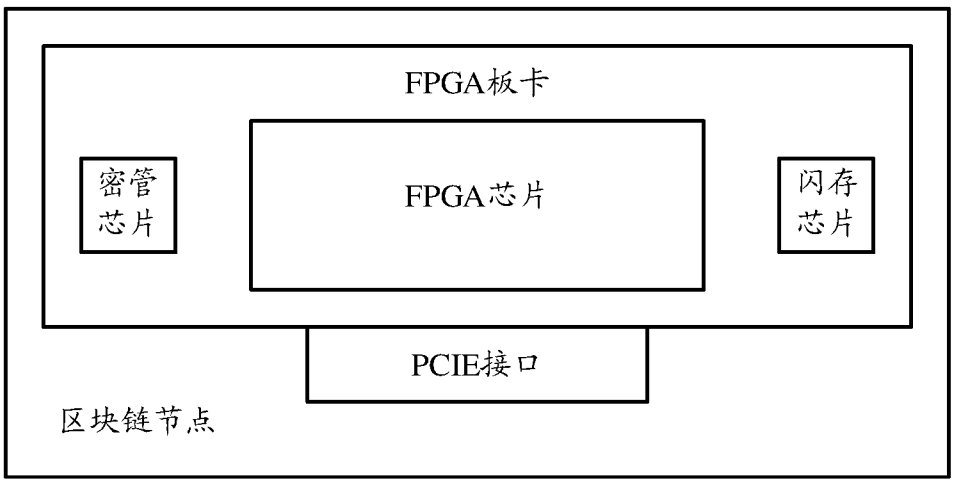
FPGA板卡
密管芯片
FPGA芯片
闪存芯片
PCIE接口
区块链节点

图 2

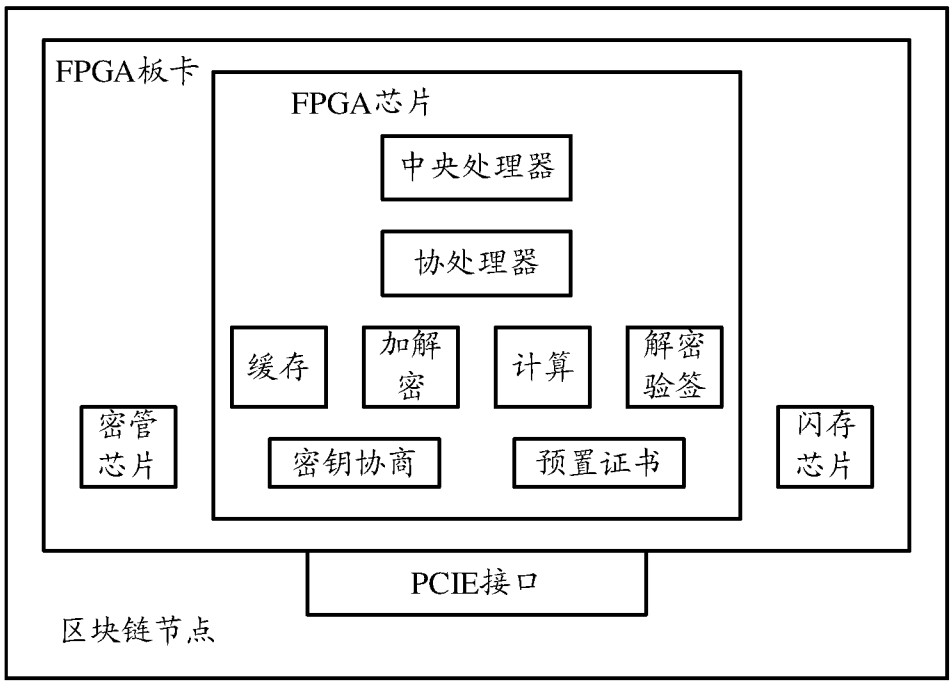
FPGA板卡
FPGA芯片
中央处理器
协处理器
缓存
加解密
计算
解密验签
密管芯片
密钥协商
预置证书
闪存芯片
PCIE接口
区块链节点

图 3

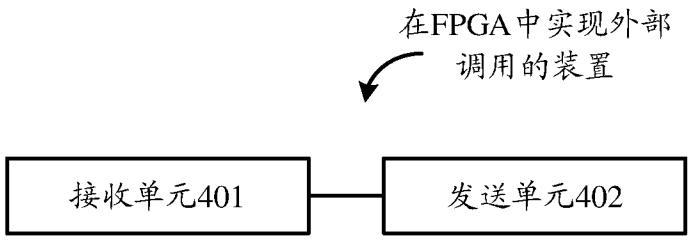
在FPGA中实现外部调用的装置
接收单元401
发送单元402

图 4

INTERNATIONAL SEARCH REPORT

International application No.
PCT/CN2020/100491

A. CLASSIFICATION OF SUBJECT MATTER

G06F 15/78(2006.01)i; G06F 9/455(2006.01)i; G06F 21/60(2013.01)i; G06F 21/62(2013.01)i; G06F 21/64(2013.01)i

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

G06F 15/; G06F 9/; G06F 21/

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

DWPI, CNABS, SIPOABS, CNTXT, USTXT, CNKI: 现场可编程门阵列, 外部调用, 芯片, 处理器, 协处理器, 调用, 请求, 调用对象, 智能合约, 区块链, 节点, 交易, 指令, 响应, 存储器, 电路逻辑, 配置文件, 加密, 解密, 缓存, 字节码, 虚拟机, 执行逻辑 FPGA, field programmable gate array, external call, chip, processor, CPU, coprocessor, call, request, calling object, intelligent contract, blockchain, node, transaction, instruction, respond, memory, storage, circuit logic, configuration file, encrypt, encipher, cipher, decrypt, decipher, buffer memory, byte code, syllable code, virtual machine, actuating logic

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
PX	CN 110750488 A (ALIPAY (HANGZHOU) INFORMATION TECHNOLOGY CO., LTD.) 04 February 2020 (2020-02-04) entire document	1-13
X	CN 109886682 A (ALIBABA GROUP HOLDING LIMITED) 14 June 2019 (2019-06-14) description paragraphs [0007]-[0036], [0160]-[0177]	1-13
A	CN 104778148 A (HARBIN INSTITUTE OF TECHNOLOGY) 15 July 2015 (2015-07-15) entire document	1-13
A	US 2003223581 A (HANOUNIK B) 04 December 2003 (2003-12-04) entire document	1-13
A	WO 2016209717 (MICROSOFT TECHNOLOGY LICENSING LLC) 29 December 2016 (2016-12-29) entire document	1-13

☐ Further documents are listed in the continuation of Box C. ☑ See patent family annex.

* Special categories of cited documents:
"A" document defining the general state of the art which is not considered to be of particular relevance
"E" earlier application or patent but published on or after the international filing date
"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
"O" document referring to an oral disclosure, use, exhibition or other means
"P" document published prior to the international filing date but later than the priority date claimed
"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
27 August 2020	**10 September 2020**
Name and mailing address of the ISA/CN **China National Intellectual Property Administration (ISA/CN)** **No. 6, Xitucheng Road, Jimenqiao Haidian District, Beijing 100088** **China** Facsimile No. **(86-10)62019451**	Authorized officer Telephone No.

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.
PCT/CN2020/100491

Patent document cited in search report	Publication date (day/month/year)	Patent family member(s)	Publication date (day/month/year)
CN 110750488 A	04 February 2020	None	
CN 109886682 A	14 June 2019	None	
CN 104778148 A	15 July 2015	None	
US 2003223581 A	04 December 2003	None	
WO 2016209717	29 December 2016	US 2016378574 A1	29 December 2016

国际检索报告

国际申请号
PCT/CN2020/100491

A. 主题的分类

G06F 15/78(2006.01)i; G06F 9/455(2006.01)i; G06F 21/60(2013.01)i; G06F 21/62(2013.01)i; G06F 21/64(2013.01)i

按照国际专利分类(IPC)或者同时按照国家分类和IPC两种分类

B. 检索领域

检索的最低限度文献(标明分类系统和分类号)

G06F 15/; G06F 9/; G06F 21/

包含在检索领域中的除最低限度文献以外的检索文献

在国际检索时查阅的电子数据库(数据库的名称，和使用的检索词（如使用）)

DWPI, CNABS, SIPOABS, CNTXT, USTXT, CNKI: 现场可编程门阵列，外部调用，芯片，处理器，协处理器，调用，请求，调用对象，智能合约，区块链，节点，交易，指令，响应，存储器，电路逻辑，配置文件，加密，解密，缓存，字节码，虚拟机，执行逻辑 FPGA, field programmable gate array, external call, chip, processor, CPU, coprocessor, call, request, calling object, intelligent contract, blockchain, node, transaction, instruction, respond, memory, storage, circuit logic, configuration file, encrypt, encipher, cipher, decrypt, decipher, buffer memory, byte code, syllable code, virtual machine, actuating logic

C. 相关文件

类 型*	引用文件，必要时，指明相关段落	相关的权利要求
PX	CN 110750488 A (支付宝杭州信息技术有限公司) 2020年 2月 4日 (2020-02-04) 全文	1-13
X	CN 109886682 A (阿里巴巴集团控股有限公司) 2019年 6月 14日 (2019-06-14) 说明书第[0007]-[0036]、[0160]-[0177]段	1-13
A	CN 104778148 A (哈尔滨工业大学) 2015年 7月 15日 (2015-07-15) 全文	1-13
A	US 2003223581 A (HANOUNIK B) 2003年 12月 4日 (2003-12-04) 全文	1-13
A	WO 2016209717 (MICROSOFT TECHNOLOGY LICENSING LLC) 2016年 12月 29日 (2016-12-29) 全文	1-13

☐ 其余文件在C栏的续页中列出。

☑ 见同族专利附件。

* 引用文件的具体类型：

“A” 认为不特别相关的表示了现有技术一般状态的文件

“E” 在国际申请日的当天或之后公布的在先申请或专利

“L” 可能对优先权要求构成怀疑的文件，或为确定另一篇引用文件的公布日而引用的或者因其他特殊理由而引用的文件（如具体说明的）

“O” 涉及口头公开、使用、展览或其他方式公开的文件

“P” 公布日先于国际申请日但迟于所要求的优先权日的文件

“T” 在申请日或优先权日之后公布，与申请不相抵触，但为了理解发明之理论或原理的在后文件

“X” 特别相关的文件，单独考虑该文件，认定要求保护的发明不是新颖的或不具有创造性

“Y” 特别相关的文件，当该文件与另一篇或者多篇该类文件结合并且这种结合对于本领域技术人员为显而易见时，要求保护的发明不具有创造性

“&” 同族专利的文件

国际检索实际完成的日期	国际检索报告邮寄日期
2020年 8月 27日	2020年 9月 10日
ISA/CN的名称和邮寄地址 中国国家知识产权局(ISA/CN) 中国北京市海淀区蓟门桥西土城路6号 100088 传真号 (86-10)62019451	受权官员 谭毅 电话号码 62412070

国际检索报告
关于同族专利的信息

国际申请号
PCT/CN2020/100491

检索报告引用的专利文件	公布日 (年/月/日)	同族专利	公布日 (年/月/日)
CN 110750488 A	2020年 2月 4日	无	
CN 109886682 A	2019年 6月 14日	无	
CN 104778148 A	2015年 7月 15日	无	
US 2003223581 A	2003年 12月 4日	无	
WO 2016209717	2016年 12月 29日	US 2016378574 A1	2016年 12月 29日