(19)中华人民共和国国家知识产权局

(12)发明专利申请

(10)申请公布号 CN 113268447 A
(43)申请公布日 2021.08.17

(21)申请号 202110648808.0

(22)申请日 2021.06.10

(71)申请人 海光信息技术股份有限公司
地址 300384 天津市滨海新区天津华苑产业区海泰西路18号北2-204工业孵化-3-8

(72)发明人 姜新 应志伟

(74)专利代理机构 上海知锦知识产权代理事务所(特殊普通合伙) 31327
代理人 王立娜

(51)Int.Cl.
G06F 13/40(2006.01)
G06F 13/16(2006.01)
G06F 12/14(2006.01)
G06F 12/06(2006.01)

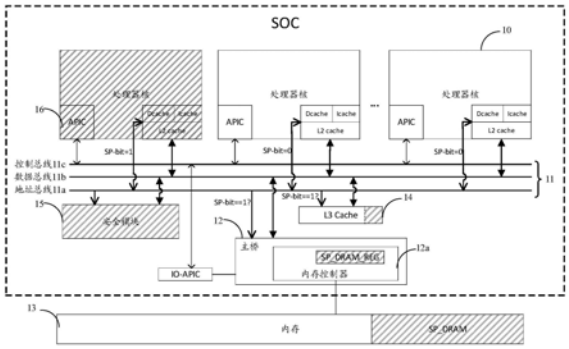
权利要求书3页 说明书15页 附图8页

(54)发明名称

计算机架构及其内的访问控制、数据交互及安全启动方法

(57)摘要

本发明实施例提供一种计算机架构及其内的访问控制、数据交互及安全启动方法，所述计算机架构包括：多个处理器核，与所述多个处理器核连接的总线，与所述总线连接的主桥，和与所述主桥连接的内存；多个处理器核中的至少一个处理器核为安全处理器核，安全处理器核产生的访问地址为安全访问地址；多个处理器核中的至少一个处理器核为普通处理器核，普通处理器核产生的访问地址为普通访问地址，所述安全访问地址与所述普通访问地址不同；所述安全处理器核和所述普通处理器核通过所述总线与所述主桥进行数据交互；所述主桥基于所述总线传输的安全访问地址和普通访问地址，执行对所述内存的数据访问，所述计算机架构提高了系统的性能。



CN 113268447 A

1.一种计算机架构,其特征在于,包括:

多个处理器核,与所述多个处理器核连接的总线,与所述总线连接的主桥,和,与所述主桥连接的内存;

所述多个处理器核中的至少一个处理器核为安全处理器核,所述安全处理器核产生的访问地址为安全访问地址;所述多个处理器核中的至少一个处理器核为普通处理器核,所述普通处理器核产生的访问地址为普通访问地址,所述安全访问地址与所述普通访问地址不同;

所述安全处理器核和所述普通处理器核通过所述总线与所述主桥进行数据交互;

所述主桥基于所述总线传输的安全访问地址和普通访问地址,执行对所述内存的数据访问。

2.根据权利要求1所述的计算机架构,其特征在于,所述多个处理器核为同构结构。

3.根据权利要求1所述的计算机架构,其特征在于,所述访问地址包括安全标记位,所述安全访问地址的安全标记位为第一值,所述普通访问地址的安全标记位为第二值。

4.根据权利要求3所述的计算机架构,其特征在于,还包括共享缓存,所述共享缓存包括相互隔离的安全缓存和普通缓存,所述安全缓存仅执行基于所述安全访问地址的数据访问,所述普通缓存仅执行基于所述普通访问地址的数据访问。

5.根据权利要求4所述的计算机架构,其特征在于,所述共享缓存中包括缓存控制器,所述缓存控制器的访问控制逻辑包括:判断访问地址中的安全标记位是否为第一值,若是,则由所述安全缓存执行对应所述访问地址的访问,若否,则由所述普通缓存执行对应所述访问地址的访问。

6.根据权利要求3所述的计算机架构,其特征在于,所述内存包括安全专有内存和普通内存,所述安全专有内存仅允许所述安全处理器核访问。

7.根据权利要求6所述的计算机架构,其特征在于,所述主桥包括内存控制器,所述内存控制器包括专有寄存器组,所述专有寄存器组内配置有所述安全专有内存的地址范围;

所述内存控制器的控制逻辑包括:所有内存地址允许基于安全访问地址的数据访问,配置在专有寄存器组内的内存地址禁止基于普通访问地址的数据访问。

8.根据权利要求7所述的计算机架构,其特征在于,所有内存地址允许基于安全访问地址的数据访问,配置在专有寄存器组内的内存地址禁止基于普通访问地址的数据访问,具体为:

根据所述访问地址的安全标记位,确定对应所述访问地址的访问权限;其中,在所述访问地址的安全标记位为第一值时,确定所述访问地址为安全访问地址,允许执行对所述访问地址的访问;在所述访问地址的安全标记位为第二值时,确定所述访问地址为普通访问地址,判断所述访问地址是否为配置在专有寄存器组内的内存地址,若是,禁止执行对所述访问地址的访问。

9.根据权利要求3所述的计算机架构,其特征在于,还包括安全模块,所述安全模块包括安全控制单元、密钥存储单元和安全启动控制单元;

其中,所述安全控制单元用于控制所述安全模块仅允许基于安全访问地址的访问;所述密钥存储单元用于存储固件密钥;所述安全启动控制单元用于执行安全启动初始化进程。

10.根据权利要求9所述的计算机架构，其特征在于，所述安全控制单元用于控制所述安全模块仅允许基于安全访问地址的访问，具体为：

根据所述访问地址的安全标记位，判断对应所述访问地址的访问权限；其中，在所述访问地址的安全标记位为第一值时，确定所述访问地址为安全访问地址，允许执行对所述安全模块的访问；在所述安全标记位为第二值时，确定所述访问地址为普通访问地址，禁止执行对所述安全模块的访问。

11.根据权利要求1所述的计算机架构，其特征在于，所述安全处理器核包括高级可编辑中断寄存器，所述高级可编辑中断寄存器用于执行与普通处理器核的交互；其中，所述安全处理器核屏蔽所述高级可编辑中断寄存器的初始化中断消息和启动中断消息。

12.根据权利要求1所述的计算机架构，其特征在于，所述内存还包括安全交互内存，所述安全交互内存用于写入所述安全处理器核与所述普通处理器核的交互数据；所述安全处理器核还包括数据交换寄存器，所述数据交换寄存器配置有所述安全交互内存的地址范围。

13.一种缓存的访问控制方法，其特征在于，所述缓存包括普通缓存和安全缓存，所述方法包括：

获取总线传输的访问请求，所述访问请求由多个处理器核中的一个处理器核发送，所述访问请求中包括访问地址；其中，所述多个处理器核中的至少一个处理器核为安全处理器核，所述安全处理器核发送的访问请求中的访问地址为安全访问地址；所述多个处理器核中的至少一个处理器核为普通处理器核，所述普通处理器核发送的访问请求的访问地址为普通访问地址，所述安全访问地址与所述普通访问地址不同；

确定所述访问请求中的访问地址是否为安全访问地址；

若是，执行对安全缓存的访问；

若否，执行对普通缓存的访问。

14.一种内存的访问控制方法，其特征在于，所述内存包括安全专有内存和普通内存，所述方法包括：

获取总线传输的访问请求，所述访问请求由多个处理器核中的一个处理器核发送，所述访问请求中包括访问地址；其中，所述多个处理器核中的至少一个处理器核为安全处理器核，所述安全处理器核发送的访问请求中的访问地址为安全访问地址；所述多个处理器核中的至少一个处理器核为普通处理器核，所述普通处理器核发送的访问请求的访问地址为普通访问地址，所述安全访问地址与所述普通访问地址不同；

确定所述访问请求中的访问地址是否为安全访问地址；

若是，执行对所述访问地址的数据访问；

若否，判断所述访问请求中的访问地址是否为安全专有内存的内存地址，若否，允许执行对所述访问地址的访问，若是，禁止执行对所述访问地址的访问。

15.根据权利要求14所述的内存访问控制方法，其特征在于，所述判断所述访问请求中的访问地址是否为安全专有内存的内存地址，具体为：

判断所述访问请求中的访问地址是否为配置在专有寄存器组内的内存地址。

16.一种安全模块的访问控制方法，其特征在于，所述方法包括：

获取总线传输的访问请求，所述访问请求由多个处理器核中的一个处理器核发送，所

述访问请求中包括访问地址;其中,所述多个处理器核中的至少一个处理器核为安全处理器核,所述安全处理器核发送的访问请求中的访问地址为安全访问地址;所述多个处理器核中的至少一个处理器核为普通处理器核,所述普通处理器核发送的访问请求的访问地址为普通访问地址,所述安全访问地址与所述普通访问地址不同;

确定所述访问请求中的访问地址是否为安全访问地址;

若是,执行对所述安全模块的访问;

若否,禁止执行对所述安全模块的访问。

17.一种数据交互方法,其特征在于,应用于权利要求1～12所述的计算机架构,所述方法包括:

普通处理器核获取安全处理器核的安全交互内存的内存地址,并将待交互数据写入所述安全交互内存;

安全处理器核根据所述安全交互内存中的待交互数据,生成响应数据,并将所述响应数据写入所述安全交互内存;

所述普通处理器核获取所述响应数据。

18.根据权利要求17所述的数据交互方法,其特征在于,所述安全处理器核根据所述安全交互内存中的待交互数据,生成响应数据的步骤中,包括:

安全处理器核基于总线传输的中断消息,读取所述安全交互内存中的待交互数据;其中,所述安全处理器核屏蔽总线传输的初始化中断消息和启动中断消息。

19.一种安全启动方法,其特征在于,应用于权利要求1～12所述的计算机架构中的安全处理器核,所述方法包括:

基于安全模块中的密钥信息,执行密钥验证;

配置所述安全处理器核的安全专有内存和安全交互内存,校验并加载基本输入输出系统;

释放并启动普通处理器核。

计算机架构及其内的访问控制、数据交互及安全启动方法

技术领域

[0001] 本发明实施例涉及软件安全技术领域，具体涉及一种计算机架构及其内的访问控制、数据交互及安全启动方法。

背景技术

[0002] 随着信息技术的发展，特别是云计算技术的发展与普及，使得越来越多的客户把业务系统部署在云端，由此使得涉及云端信息安全的业务例如电子签名、电子合同、在线支付以及数字认证等的重要性日益突出。如何构建可信数字化的安全平台，成为人们关注的技术焦点。

[0003] 虚拟化技术作为一种应用于云端的计算机技术，可通过主机虚拟化出多台虚拟机(Virtual Machine，VM)，以实现对主机的硬件资源的高效利用；同时，在主机中可以设置专门的安全处理器(Platform Secure Processor，PSP)，为系统提供固件验签、密钥生成、数据加解密等可信安全业务，以提高平台的安全性。

[0004] 然而，配置有安全处理器的计算机架构的系统性能有待提高。

发明内容

[0005] 有鉴于此，本发明实施例提供一种计算机架构及其内的访问控制、数据交互及安全启动方法，能够提高系统性能。

[0006] 为实现上述目的，本发明实施例提供如下技术方案：

[0007] 在本发明的一个实施例中，提供了一种计算机架构，包括：

[0008] 多个处理器核，与所述多个处理器核连接的总线，与所述总线连接的主桥，和，与所述主桥连接的内存；

[0009] 所述多个处理器核中的至少一个处理器核为安全处理器核，所述安全处理器核产生的访问地址为安全访问地址；所述多个处理器核中的至少一个处理器核为普通处理器核，所述普通处理器核产生的访问地址为普通访问地址，所述安全访问地址与所述普通访问地址不同；

[0010] 所述安全处理器核和所述普通处理器核通过所述总线与所述主桥进行数据交互；

[0011] 所述主桥基于所述总线传输的安全访问地址和普通访问地址，执行对所述内存的数据访问。

[0012] 可选的，所述多个处理器核为同构结构。

[0013] 可选的，所述访问地址包括安全标记位，所述安全访问地址的安全标记位为第一值，所述普通访问地址的安全标记位为第二值。

[0014] 可选的，还包括共享缓存，所述共享缓存包括相互隔离的安全缓存和普通缓存，所述安全缓存仅执行基于所述安全访问地址的数据访问，所述普通缓存仅执行基于所述普通访问地址的数据访问。

[0015] 可选的，所述共享缓存中包括缓存控制器，所述缓存控制器的访问控制逻辑包括：

判断访问地址中的安全标记位是否为第一值，若是，则由安全缓存执行对应所述访问地址的访问，若否，则由普通缓存执行对应所述访问地址的访问。

[0016] 可选的，所述内存包括安全专有内存和普通内存，所述安全专有内存仅允许所述安全处理器核访问。

[0017] 可选的，所述主桥包括内存控制器，所述内存控制器包括专有寄存器组，所述专有寄存器组内配置有所述安全专有内存的地址范围；

[0018] 所述内存控制器的控制逻辑包括：所有内存地址允许基于安全访问地址的数据访问，配置在专有寄存器组内的内存地址禁止基于普通访问地址的数据访问。

[0019] 可选的，所有内存地址允许基于安全访问地址的数据访问，配置在专有寄存器组内的内存地址禁止基于普通访问地址的数据访问，具体为：

[0020] 根据所述访问地址的安全标记位，确定对应所述访问地址的访问权限；其中，在所述访问地址的安全标记位为第一值时，确定所述访问地址为安全访问地址，允许执行对所述访问地址的访问；在所述访问地址的安全标记位为第二值时，确定所述访问地址为普通访问地址，判断所述访问地址是否为配置在专有寄存器组内的内存地址，若是，禁止执行对所述访问地址的访问。

[0021] 可选的，还包括安全模块，所述安全模块包括安全控制单元、密钥存储单元和安全启动控制单元；

[0022] 其中，所述安全控制单元用于控制所述安全模块仅允许基于安全访问地址的访问；所述密钥存储单元用于存储固件密钥；所述安全启动控制单元用于执行安全启动初始化进程。

[0023] 可选的，所述安全控制单元用于控制所述安全模块仅允许基于安全访问地址的访问，具体为：

[0024] 根据所述访问地址的安全标记位，判断对应所述访问地址的访问权限；其中，在所述访问地址的安全标记位为第一值时，确定所述访问地址为安全访问地址，允许执行对所述安全模块的访问；在所述安全标记位为第二值时，确定所述访问地址为普通访问地址，禁止执行对所述安全模块的访问。

[0025] 可选的，所述安全处理器核包括高级可编辑中断寄存器，所述高级可编辑中断寄存器用于执行与普通处理器核的交互；其中，所述安全处理器核屏蔽所述高级可编辑中断寄存器的初始化中断消息和启动中断消息。

[0026] 可选的，所述内存还包括安全交互内存，所述安全交互内存用于写入所述安全处理器核与所述普通处理器核的交互数据；所述安全处理器核还包括数据交换寄存器，所述数据交换寄存器配置有所述安全交互内存的地址范围。

[0027] 在本发明的一个实施例中，还提供了一种缓存的访问控制方法，所述缓存包括普通缓存和安全缓存，所述方法包括：

[0028] 获取总线传输的访问请求，所述访问请求由多个处理器核中的一个处理器核发送，所述访问请求中包括访问地址；其中，所述多个处理器核中的至少一个处理器核为安全处理器核，所述安全处理器核发送的访问请求中的访问地址为安全访问地址；所述多个处理器核中的至少一个处理器核为普通处理器核，所述普通处理器核发送的访问请求的访问地址为普通访问地址，所述安全访问地址与所述普通访问地址不同；

[0029] 确定所述访问请求中的访问地址是否为安全访问地址；

[0030] 若是，执行对安全缓存的访问；

[0031] 若否，执行对普通缓存的访问。

[0032] 可选的，所述确定所述访问请求中的访问地址是否为安全访问地址，包括：

[0033] 判断访问地址中的安全标记位是否为第一值，在访问地址中的安全标记位为第一值时，确定所述访问地址为安全访问地址，在访问地址中的安全标记位为第二值时，确定所述访问地址非安全访问地址。

[0034] 在本发明的一个实施例中，还提供了一种内存的访问控制方法，所述内存包括安全专有内存和普通内存，所述方法包括：

[0035] 获取总线传输的访问请求，所述访问请求由多个处理器核中的一个处理器核发送，所述访问请求中包括访问地址；其中，所述多个处理器核中的至少一个处理器核为安全处理器核，所述安全处理器核发送的访问请求中的访问地址为安全访问地址；所述多个处理器核中的至少一个处理器核为普通处理器核，所述普通处理器核发送的访问请求的访问地址为普通访问地址，所述安全访问地址与所述普通访问地址不同；

[0036] 确定所述访问请求中的访问地址是否为安全访问地址；

[0037] 若是，执行对所述访问地址的数据访问；

[0038] 若否，判断所述访问请求中的访问地址是否为安全专有内存的内存地址，若否，允许执行对所述访问地址的访问，若是，禁止执行对所述访问地址的访问。

[0039] 可选的，所述判断所述访问请求中的访问地址是否为安全专有内存的内存地址，具体为：

[0040] 判断所述访问请求中的访问地址是否为配置在专有寄存器组内的内存地址。

[0041] 可选的，所述确定所述访问请求中的访问地址是否为安全访问地址，具体为：

[0042] 判断所述访问请求中的访问地址的安全标记位是否为第一值，在访问地址中的安全标记位为第一值时，确定所述访问地址为安全访问地址，在访问地址中的安全标记位为第二值时，确定所述访问地址非安全访问地址。

[0043] 在本发明的一个实施例中，还提供了一种安全模块访问控制方法，所述方法包括：

[0044] 获取总线传输的访问请求，所述访问请求由多个处理器核中的一个处理器核发送，所述访问请求中包括访问地址；其中，所述多个处理器核中的至少一个处理器核为安全处理器核，所述安全处理器核发送的访问请求中的访问地址为安全访问地址；所述多个处理器核中的至少一个处理器核为普通处理器核，所述普通处理器核发送的访问请求的访问地址为普通访问地址，所述安全访问地址与所述普通访问地址不同；

[0045] 确定所述访问请求中的访问地址是否为安全访问地址；

[0046] 若是，执行对所述安全模块的访问；

[0047] 若否，禁止执行对所述安全模块的访问。

[0048] 可选的，所述确定所述访问请求中的访问地址是否为安全访问地址，具体为：

[0049] 判断所述访问请求中的访问地址的安全标记位是否为第一值，在访问地址中的安全标记位为第一值时，确定所述访问地址为安全访问地址，在访问地址中的安全标记位为第二值时，确定所述访问地址非安全访问地址。

[0050] 在本发明的一个实施例中，还提供了一种数据交互方法，应用于上述实施例所述

的计算机架构，所述方法包括：

[0051] 普通处理器核获取安全处理器核的安全交互内存的内存地址，并将待交互数据写入所述安全交互内存；

[0052] 安全处理器核根据所述安全交互内存中的待交互数据，生成响应数据，并将所述响应数据写入所述安全交互内存；

[0053] 普通处理器核获取所述响应数据。

[0054] 可选的，所述安全处理器核根据所述安全交互内存中的待交互数据，生成响应数据的步骤中，包括：

[0055] 安全处理器核基于总线传输的中断消息，读取所述安全交互内存中的待交互数据；其中，所述安全处理器核屏蔽总线传输的初始化中断消息和启动中断消息。

[0056] 在本发明的一个实施例中，还提供了一种安全启动方法，应用于上述实施例所述的计算机架构，所述方法包括：

[0057] 基于安全模块中的密钥信息，执行密钥验证；

[0058] 配置安全专有内存和安全交互内存，校验并加载基本输入输出系统；

[0059] 释放并启动普通处理器核。

[0060] 本发明实施例提供的计算机架构，包括：多个处理器核，与所述多个处理器核连接的总线，与所述总线连接的主桥，和，与所述主桥连接的内存；所述多个处理器核中的至少一个处理器核为安全处理器核，所述安全处理器核产生的访问地址为安全访问地址；所述多个处理器核中的至少一个处理器核为普通处理器核，所述普通处理器核产生的访问地址为普通访问地址，所述安全访问地址与所述普通访问地址不同；所述安全处理器核和所述普通处理器核通过所述总线与所述主桥进行数据交互；所述主桥基于所述总线传输的安全访问地址和普通访问地址，执行对所述内存的数据访问。

[0061] 可以看出，安全处理器和与普通处理器核均基于所述总线执行数据交互，其中，安全处理器核产生的访问地址为与普通访问地址不同的安全访问地址，从而可以在系统中基于不同类型的访问地址实现地址的区分，使安全处理器核实现基于总线进行数据交互的同时，还为安全处理器核的数据交互提供数据隔离的基础，保证安全处理器的信息安全。也就是说，本发明实施例在保证安全处理器的安全的前提下，采用系统内的总线执行信息传输，从而提高了数据传输的效率，进而提高了系统的性能。

附图说明

[0062] 为了更清楚地说明本发明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发明的实施例，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，还可以根据提供的附图获得其他的附图。

[0063] 图1为基于虚拟化技术的云服务可选架构图；

[0064] 图2为的云主机的系统架构示意图；

[0065] 图3为本发明实施例提供的一种计算机架构的可选示意图；

[0066] 图4为本发明实施例提供的一种访问地址的地址位示例图；

[0067] 图5为本发明实施例提供的共享缓存结构示意图；

[0068] 图6为本发明实施例提供的专有寄存器组结构示例图；

[0069] 图7A为本发明实施例提供的处理器核的数据访问流程示意图；

[0070] 图7B为本发明实施例提供的处理器核的数据访问流程的一个可选示例图；

[0071] 图8为本发明实施例提供的另一计算机架构示意图；

[0072] 图9为本发明实施例提供的安全模块的结构示意图；

[0073] 图10为本发明实施例提供的高级可编程中断寄存器的结构示例图；

[0074] 图11为本发明实施例提供的数据交互流程示意图；

[0075] 图12为本发明实施例一种安全处理器核和普通处理器核的交互示例图；

[0076] 图13为本发明实施例提供的安全处理器核的安全启动流程示意图；

[0077] 图14为本发明实施例提供的步骤S30的流程示意图；

[0078] 图15为本发明实施例提供的步骤S31的流程示意图。

具体实施方式

[0079] 下面将结合本发明实施例中的附图，对本发明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发明一部分实施例，而不是全部的实施例。基于本发明中的实施例，本领域普通技术人员在没有做出创造性劳动前提下所获得的所有其他实施例，都属于本发明保护的范围。

[0080] 虚拟化技术可应用于多种场景，特别的，随着云服务的发展，虚拟化技术在云服务这一场景中得到了越来越广泛的应用；为便于理解虚拟化技术，下面以基于虚拟化技术的云服务为例，对虚拟化技术进行介绍。

[0081] 参照图1所示基于虚拟化技术的云服务可选架构，该云服务架构可以包括：云主机100，网络20，用户31至3n；

[0082] 其中，云主机100为云服务提供方(如云服务厂商)部署在网络侧的用于提供云服务的主机设备(主机设备的形式可以是服务器)；基于不同用户的需求，云主机可通过虚拟化技术为不同用户创建一台或多台虚拟机，例如，用户可以根据业务需求，请求云主机为用户创建适应其业务需求的多台虚拟机，从而用户可以在属于其的多台虚拟机上分别运行应用，以通过多台虚拟机运行的应用来协作完成用户特定的业务；

[0083] 网络20可以认为是互联网，或者其他形式的具有通信功能的网络，云主机与用户之间通过网络实现通信和数据传输，本发明实施例并不限制网络20的具体网络形式；

[0084] 用户31至3n为使用云服务的注册用户，其数量可以为多个，本发明实施例并不限制n的具体数值；在云服务场景下，每一个用户可以在云主机中拥有属于其的一台或多台虚拟机，以利用属于用户的虚拟机来完成用户特定的业务。

[0085] 为便于进一步清楚的了解云服务场景的虚拟化技术，参照图2所示的云主机的系统架构示意图，如图2所示，云主机包括：处理器核(CPU core)1，主桥(host bridge)2，内存(DRAM)3和安全处理器(Platform Secure Processor，PSP)4；可选的，处理器核1，主桥2和安全处理器4可集成于SOC(片上系统)。

[0086] 其中，处理器核1是物理主机的运算和控制核心，用于解释计算机指令以及处理计算机软件中的数据。处理器核1可以通过虚拟化技术可以虚拟出多台虚拟机，并实现虚拟机的运行和管理，处理器核1例如可以为x86处理器。

[0087] 主桥2是控制内存3，并且使内存3与其它组件(如处理器核1)之间交换数据的硬件；在处理器的运行过程中，处理器核1可通过总线与主桥2交互。

[0088] 安全处理器4是安全虚拟化技术专门设置的负责虚拟机数据安全的处理器。安全处理器4中可以包括安全处理器核41、安全模块42和密码引擎43。

[0089] 安全处理器核41用于实现可信管理过程中的运算和控制，为虚拟机中的安全数据提供加解密，并为处理器核1实现安全启动(secure boot)。其中，安全处理器核41通常为与处理器核1异构的处理器，例如，安全处理器核可以为具有较小能耗的ARM处理器。

[0090] 安全模块42存储有固件签名和用于初始化secure boot(安全启动)安全机制的代码，并为安全启动的初始化(bootrom)阶段提供存储空间。

[0091] 在这一系统架构中，安全处理器4只能通过专门的系统集线器(System Hub)5桥接到主桥2上，以实现对内存3的访问，以及与安全处理器4以外的硬件的数据交互，并保证安全处理器4以外的硬件无法直接访问安全处理器4内的安全资源，从而在硬件架构上保证系统的安全。

[0092] 然而，此种数据交互方式，数据传输效率低，从而导致系统性能有待提高。

[0093] 基于此，本发明实施例提供了一种计算机架构，安全处理器和与普通处理器核均基于所述总线执行数据交互，其中，安全处理器核产生的访问地址为与普通访问地址不同的安全访问地址，从而可以在系统中基于不同类型的访问地址实现地址的区分，使安全处理器核实现基于总线进行数据交互的同时，还为安全处理器核的数据交互提供数据隔离的基础，保证安全处理器的信息安全。也就是说，本发明实施例在保证安全处理器的安全的前提下，采用系统内的总线执行信息传输，从而提高了数据传输的效率，进而提高了系统的性能。

[0094] 在一个可选的示例中，参照图3所示的计算机架构可选示意图，如图3所示，计算机架构包括：多个处理器核10，总线11，主桥12和内存13；

[0095] 多个处理器核10中，可以定义至少一个处理器核为安全处理器核(以图中阴影下的处理器核为安全处理器核)，并使该安全处理器核作为安全处理器的处理器核，实现可信管理过程中的运算和控制。同时，定义至少一个处理器核为普通处理器核(以图中没有阴影的处理器核为普通处理器核)，并使该普通处理器核作为虚拟机的处理器核，实现虚拟机的运行和管理。可选的，在定义安全处理器核之后，可以定义多个处理器核中剩余的处理器核均为普通处理器核。

[0096] 可以理解的是，在一个多核结构的片上系统，多个处理器核可以为同构结构，所述同构结构指的是，具有相同的架构，或者，具有相同的结构。通过在同构的多个处理器核中定义一个处理器核为安全处理器核，有利于提高系统的可扩展性。同时，与普通处理器同构的安全处理器，更利于采用相同的数据交互机制，从而可以提高数据交互的效率，提升系统的性能。

[0097] 其中，处理器核通过总线实现与其他硬件的信息交互，可选的，总线11可以包括地址总线11a、数据总线11b和控制总线11c，在进行数据访问时，可以基于地址总线11a中的地址确定即将访问的访问地址，其中，访问地址可以为内存访问地址，也可以为其他存储单元中的访问地址，例如，安全模块中的存储空间的访问地址。

[0098] 在本发明实施例中，可以将访问地址区分为安全访问地址和普通访问地址，其中，

安全处理器核产生的访问地址为安全访问地址，普通处理器核产生的访问地址为普通访问地址，安全访问地址与普通访问地址不同。可选的，安全访问地址和普通访问地址可以通过在访问地址中定义安全标记位，并基于安全标记位的不同值进行区分，确定访问地址为安全访问地址还是普通访问地址。

[0099] SOC片内的地址总线通常为多位(bit)，例如，64位处理器的片内地址总线为48位，从而可以支持多达256TB的内存。然而，系统中的内存通常远小于256TB，因此，可以定义地址总线的高位为安全标记位。参考图4示出的一种访问地址的地址位示例图，其中，48位的访问地址中，可以定义0～45位为实际地址位(add-bit)，46位为安全标记位(SP-bit)，以标记该访问地址是安全访问地址/普通访问地址，47位为加密位(c-bit)，以标记该访问地址所指向的存储空间(例如内存)是加密/非加密状态。

[0100] 在一个可选的示例中，可以定义所述安全访问地址的安全标记位为第一值，例如1，所述普通访问地址的安全标记位为第二值，例如0。具体的例子如，定义SP-bit＝1，指示当前的访问地址为由安全处理器核产生的安全访问地址；定义SP-bit＝0，指示当前的访问地址为由普通处理器核产生的普通访问地址。

[0101] 可以理解的是，所述安全标记位可以基于不同的需求进行适应性设置，例如，设置在访问地址的最高位、中间位等，本发明在此不做具体的限定。在确定具体的安全标记位后，可以通过硬件进行设置，在安全处理器核发出的访问地址中，始终设置安全标记位为第一值，而其他普通处理器核发出的访问地址中，安全标记位为第二值。

[0102] 可以理解的是，参考图3所示的计算机架构示意图，在处理器核中，通常包括独立的缓存L1 cache(图中未标示)和L2 cache，其中，L1 cache又分为指令缓存Icache和数据缓存Dcache(如图3所示)，从而，在安全处理器核中，可以仅基于其内部的独立的缓存L1 cache和L2 cache执行相应的数据交互，也可以同时基于共享缓存(例如L3 cache)执行相应的数据交互。

[0103] 在本发明实施例中，参考图3所示的计算机架构示意图，所述计算机架构进一步包括共享缓存14，在图3的示例中，以共享缓存为L3 cache为例进行说明。结合图5示出的共享缓存结构示意图，共享缓存中可以包括相互隔离的安全缓存14a和普通缓存14b。所述安全缓存14a可以配置为仅执行基于安全访问地址的数据访问，所述普通缓存14b可以配置为仅执行基于普通访问地址的数据访问。所述安全缓存14a和所述普通缓存14b相互隔离，以实现缓存数据的隔离。可以理解的是，基于普通处理器核的数据访问量远大于安全处理器核的数据访问量，可以设置普通缓存14b的缓存空间大于安全缓存14a的缓存空间。

[0104] 继续参考图3，在所述共享缓存14与处理器核10进行数据交互时，基于总线11获取相应的地址数据，具体的，基于总线11中的地址总线11a获取地址数据，该地址数据可以为安全访问地址，也可以为普通访问地址。

[0105] 所述共享缓存14可以配置有基于访问地址进行访问控制的逻辑，具体的，所述共享缓存判断访问地址是否为安全访问地址，若是，则由安全缓存执行对应所述访问地址的访问，若否，则由普通缓存执行对应所述访问地址的访问。

[0106] 具体的，在所述共享缓存14的缓存控制器中增加安全标记位(总线SP-bit)处理机制，从而可以基于安全标记位实现判断访问地址是否为安全访问地址的逻辑，进而控制所述安全缓存仅基于所述安全访问地址执行数据访问，所述普通缓存仅基于所述普通访问地

址执行数据访问。具体的，所述缓存控制器的访问控制逻辑可以为：判断访问地址中的安全标记位是否为第一值，若是，则由安全缓存执行对应所述访问地址的访问，若否，则由普通缓存执行对应所述访问地址的访问。举例来说，当SP-bit＝1时，仅读写L3 cache中的安全缓存(图3中的阴影部分)；当SP-bit＝0时，仅读写L3 cache中的普通缓存(图3中的没有阴影的部分)。

[0107] 进一步的，参考图3，在本发明实施例中，内存13配置有独立的仅供安全处理器核访问的安全专有内存(SP_DRAM)，从而将安全处理器核的数据与普通处理器核的数据隔离存储，保证安全处理器的数据安全。

[0108] 具体的，可以通过对内存的访问控制逻辑进行相应的配置，从而实现安全专有内存仅供安全处理器核访问。可以理解的是，主桥12用于实现内存的访问控制。具体的，所述主桥12包括内存控制器12a(UMC)，主桥12通过所述内存控制器12a实现对内存的访问控制。

[0109] 在一个可选的示例中，所述内存控制器12a包括对应安全专有内存的专有寄存器组SP_DRAM_REG，所述专有寄存器组SP_DRAM_REG内配置有安全专有内存SP_DRAM的地址范围，并使配置所述专有寄存器器组SP_DRAM_REG的地址范围内的内存，仅安全处理器核可访问，普通处理器核禁止访问。

[0110] 具体的，参考图6所示的专有寄存器组结构示例图，专有寄存器组可以包括用于标识起始地址的起始寄存器SP_DRAM_START，其初始值可以为0，和，用于标识结束地址的结束寄存器SP_DRAM_END，其初始值可以为0xfffffffffff。

[0111] 需要说明的是，在内存控制器中，专有寄存器组可以为一组，也可以为多组，在具有多组专有寄存器时，可以划定多个不同的地址范围作为安全专有内存。其中，所述专有寄存器仅允许安全处理器核进行配置，普通处理器核无法进行所述专有寄存器的配置。

[0112] 需要说明的是，在系统启动阶段，认为全部内存均为是安全处理器的安全专有内存，仅安全处理器核可以访问，其他处理器核无法访问内存。

[0113] 在内存控制器的控制逻辑中，所有内存地址允许基于安全访问地址的数据访问，配置在专有寄存器组内的内存地址禁止基于普通访问地址的数据访问，从而避免普通处理器核访问安全专有内存。

[0114] 继续参考图3，在具体的内存访问过程中，主桥12基于总线11获取相应的地址数据。具体的，主桥12基于总线11中的地址总线11a获取相应的地址数据。可以理解的是，主桥12获取的地址数据，可以为安全访问地址，也可以为普通访问地址。在获取相应的地址数据后，由内存控制器12a执行相应的内存访问控制。

[0115] 可选的，内存控制器12a可以根据从地址总线获取的地址数据确定访问地址，并进一步基于访问地址的安全标识位，确定对应访问地址的访问权限。其中，在所述访问地址的安全标记位为第一值时，确定所述访问地址为安全访问地址，允许执行所述访问地址的访问；在所述访问地址的安全标记位为第二值时，确定所述访问地址为普通访问地址，判断所述访问地址是否为专有寄存器组内的内存地址，若是，禁止执行所述访问地址的访问。

[0116] 结合前述举例，当访问请求中的访问地址中，SP-bit＝1，对应该访问地址的访问请求可以访问所有内存，即安全处理器核具有访问所有内存的权限；当访问请求中的访问地址中，SP-bit＝0，则对应该访问地址的访问请求仅可以访问SP_DRAM_REG寄存器组配置的内存地址范围之外的普通内存，即普通处理器核不具有访问安全专有内存的权限。

[0117] 基于上述计算机架构，本发明实施例进一步提供了一种处理器核的数据访问流程，参考图7A示出的处理器核的数据访问流程示意图，所述数据访问流程包括：

[0118] 步骤S10：处理器核发出访问请求，所述访问请求包括访问地址；

[0119] 其中，所述访问请求由多个处理器核中的一个处理器核发送，发出访问请求的处理器核可以为安全处理器核，也可以为普通处理器核；所述访问请求可以为内存访问请求，所述访问地址为可以内存的访问地址，其中，安全处理器核发出的访问请求中，访问地址为安全访问地址，普通处理器核发出的访问请求中，访问地址为普通访问地址。所述安全访问地址与所述普通访问地址不同，在一个可选的示例中，安全访问地址和普通访问地址在安全标记位的值不同。

[0120] 其中，处理器核通过总线发出所述访问请求。具体的，在所述总线中的地址总线，传输所述访问请求的访问地址。

[0121] 步骤S11：共享缓存基于所述访问请求，执行对应所述访问地址的访问；

[0122] 在处理器核通过总线发出所述访问请求后，共享缓存可以获取总线传输的访问请求。

[0123] 其中，在共享缓存中存储有所述访问地址的数据时，所述共享缓存命中所述访问请求，执行对应所述访问地址的访问；在共享缓存中未存储所述访问地址的数据时，所述共享缓存未命中所述访问请求，执行步骤S12；

[0124] 在本发明实施例中，基于所述共享缓存进一步划分为安全缓存和普通缓存，相应的，所述访问请求中的访问地址为安全访问地址时，访问所述安全缓存，所述访问请求中的访问地址为普通访问地址时，访问所述普通缓存。

[0125] 具体的，参考图7B示出的处理器核的数据访问流程的一个可选示例图，步骤S11可以包括：

[0126] 步骤S110：确定所述访问请求中的访问地址是否为安全访问地址；

[0127] 在所述访问请求中的访问地址为安全访问地址时，执行步骤S111；在所述访问请求中的访问地址为普通访问地址时，执行步骤S112。

[0128] 其中，可以基于访问地址中的安全标记位确定所述访问地址是否为安全访问地址，具体的，判断访问地址中的安全标记位是否为第一值，在访问地址中的安全标记位为第一值时，例如1，确定所述访问地址为安全访问地址，在访问地址中的安全标记位为第二值时，例如0，确定所述访问地址非安全访问地址，即为普通访问地址。

[0129] 步骤S111：执行对安全缓存的访问；

[0130] 其中，在对安全缓存的访问过程中，确定访问是否命中，命中时，执行步骤S113，未命中时，返回所述访问请求，由总线传输所述访问请求至所述内存控制器，执行步骤S12。

[0131] 步骤S112：执行对普通缓存的访问。

[0132] 其中，在对普通缓存的访问过程中，确定访问是否命中，命中时，由执行步骤S113，未命中时，返回所述访问请求，由总线传输所述访问请求至所述内存控制器，执行步骤S12。

[0133] 步骤S113：向总线返回所述访问请求所访问的数据。

[0134] 在访问命中时，共享缓存可以向总线传输所述访问请求所访问的数据。

[0135] 继续参考图7A，执行步骤S12：内存控制器基于所述访问请求，执行对应所述访问地址的数据访问；

[0136] 其中，所述访问请求基于总线传输，相应的，主桥中的内存控制器，可以在获取总线传输的访问请求后，执行相应的数据访问。

[0137] 基于本发明实施例中将内存划分为安全专有内存和普通内存，其中，安全专有内存仅允许基于安全访问地址访问，而普通内存则同时允许基于安全访问地址的访问和基于普通访问地址的访问。相应的，内存控制器需要基于所述访问请求中的访问地址进行访问权限的判断。

[0138] 内存控制器中可以基于访问请求中的访问地址，确定所述访问请求的访问权限，并进一步基于所述访问权限，执行所述访问请求。其中，在所述访问请求的访问地址为安全访问地址时，所述访问请求具有对所有内存的访问权限，从而，允许执行对所述访问地址的访问；在所述访问请求的访问地址为普通访问地址时，所述访问请求仅具有对普通内存的访问权限，从而，需要判断所述访问地址是否为安全专有内存的内存地址，基于安全专有内存的地址范围配置在专有寄存器组内，可以基于专有寄存器组内的地址数据进行相应的判断。具体的，判断所述访问请求中的访问地址是否为配置在专有寄存器组内的内存地址，若否，允许执行对所述访问地址的访问，若是，则禁止执行对所述访问地址的访问。

[0139] 具体的，参考图7B示出的处理器核的数据访问流程的一个可选示例图，步骤S12可以包括：

[0140] 步骤S120：确定所述访问请求中的访问地址是否为安全访问地址。

[0141] 在所述访问请求中的访问地址为普通访问地址时，执行步骤S121；在所述访问请求中的访问地址为安全访问地址时，执行步骤S122。

[0142] 具体的，可以基于访问地址中的安全标记位确定所述访问地址是否为安全访问地址，具体的，判断所述访问请求中的访问地址的安全标记位是否为第一值，在访问地址中的安全标记位为第一值时，例如1，确定所述访问地址为安全访问地址，在访问地址中的安全标记位为第二值时，例如0，确定所述访问地址为普通访问地址。

[0143] 步骤S121：判断所述访问请求中的访问地址是否为安全专有内存的内存地址。

[0144] 在所述访问请求的访问地址为普通访问地址时，则需要进一步判断所述访问请求中的访问地址是否为安全专有内存的内存地址。具体的，可以判断所述访问请求中的访问地址是否为配置在专有寄存器组内的内存地址，若否，允许执行对所述访问地址的访问，执行步骤S122，若是，则认为访问异常，禁止执行对所述访问地址的访问，并将所述访问异常的结果反馈至总线，由总线将该访问结果传输中相应的处理器核。

[0145] 需要说明的是，对访问地址是否为配置在专有寄存器组内的内存地址，可以基于访问地址的实际地址位进行判断，例如，在图4中的地址数据中，位于0～45位的值为地址数据，从而可以基于该位置的地址数据进行判断。

[0146] 步骤S122：执行对应所述访问地址的数据访问。

[0147] 在所述访问请求的访问地址为安全访问地址时，所述访问请求具有对所有内存的访问权限，从而可以直接执行对应所述访问地址的数据访问。

[0148] 步骤S123：向总线返回所述访问请求所访问的数据。

[0149] 在访问到对应所述访问地址的数据时，内存控制器可以向总线传输所述访问请求所访问的数据，由总线将相应的数据传输至共享缓存，并进一步由共享缓存传输至相应的处理器核。

[0150] 可以理解的是，在本发明实施例中，安全处理器核可以访问所有内存，同时，基于安全处理器核与普通处理器核的同构结构，使得安全处理器核上的安全软件在业务安全的前提下可以直接调用普通处理器核的服务程序，从而可以有效提高安全处理器的效率。而通过相应的权限设置，使得普通处理器核无法访问安全处理器专用的安全资源，保证了安全处理器核的安全性。

[0151] 在本发明的另一实施例中，提供了另一计算机架构，参考图8所示的另一计算机架构示意图，所述计算机架构进一步包括安全模块15，结合参考图9所示的安全模块的结构示意图，所述安全模块包括：安全控制单元(secure controller)15a，密钥存储单元15b和安全启动控制单元15c；其中，安全模块15基于总线获取相应的地址数据，且所述地址数据可以为安全访问地址，也可以为普通访问地址。

[0152] 其中，所述安全控制单元15a用于控制所述安全模块仅允许基于安全访问地址的访问；可选的，所述安全控制单元可以根据访问地址的安全标记位，确定所述访问地址是否为安全访问地址，进而判断对应所述访问地址的访问权限。具体的，可以判断所述访问请求中的访问地址的安全标记位是否为第一值。在访问地址的安全标记位为第一值时，例如SP-bit＝1，则确定访问地址为安全访问地址，进而允许执行对安全模块中的访问。在访问地址的安全标记位为第二值时，例如SP-bit＝0，则确定访问地址为普通访问地址，进而禁止执行对安全模块的访问。

[0153] 所述密钥存储单元15b用于存储固件密钥。可选的，所述密钥存储单元可以为eFUSE(一次性编程存储模块)，供OEM(Original Equipment Manufacturer，原始设备制造商)厂商使用，OEM厂商可以在eFuse中写入厂商密钥等，以通过厂商密钥实现对BIOS(Basic Input Output System，基本输入输出系统)的验签认证等。

[0154] 所述安全启动控制单元15c用于执行安全启动初始化进程。可选的，所述安全启动控制单元15c可以为bootrom，为SOC内置的一段只读代码，拥有最高的执行权限，用于执行secure boot初始化进程，实现加载密钥，验证固件等功能。

[0155] 在一个可选的示例中，所述安全模块还可以进一步包括安全启动存储单元15d，用于在安全模块内部为安全启动初始化进程提供存储空间，避免安全启动初始化进程使用安全模块以外的存储设备，保证安全启动控制单元在执行相应进程时的安全性。

[0156] 继续参考图8，本发明实施例中，所述安全处理器核和所述普通处理器核均包括高级可编辑中断寄存器(APIC，Advanced programmable interrupt controller)16，以使得所述安全处理器核基于与普通处理器核相同的中断体系执行与普通处理器核之间的交互。

[0157] 可以理解的是，处理器核10可以通过发送中断(IPI，inter-processor interrupt)消息至另一处理器核10的高级可编程中断寄存器，从而使得接收中断消息的处理器核基于该中断消息执行相应的操作。同时，处理器核还可以基于高级可编辑中断寄存器接口(IO-APIC)实现与主桥的数据交互。

[0158] 具体的，参考图10示出的高级可编程中断寄存器(下称APIC)的结构示例图，可以看出，APIC可以为64位，其中的MT字段用于指示中断消息的消息类型，VEC字段用于指示矢量值，在本发明实施例中，基于中断消息的不同的消息类型，可以进行相应中断消息的屏蔽，以保证安全处理器核的安全。

[0159] 其中，中断消息主要包括如下类型：

[0160] 1)fixed,固定中断消息,根据vector(VEC字段的值)确定中断号;

[0161] 2)Lowest priority,最低优先级请求中断;

[0162] 3)SMI(System Management Interrupt),系统管理中断;

[0163] 4)Remote read,读远程处理器核的APIC数据;

[0164] 5)NMI,非可屏蔽中断;

[0165] 6)INIT,初始化中断,用于恢复处理器核到初始状态;

[0166] 7)STARTUP,启动中断,用于指定本地处理器核的启动路径;

[0167] 8)External interrupt,外部中断,用于处理外设交互;

[0168] 其中,为保证安全处理器的安全,可以配置安全处理器核屏蔽普通处理器核的启动核间中断(SIPI,start inter-processor interrupt)消息,从而防止响应该消息造成的安全处理器核被植入非法执行路径。可选的,所述安全处理器核屏蔽初始化中断(INIT)消息和启动中断(STARTUP)消息,通过屏蔽这两种类型的中断消息,从而防止普通处理器核篡改安全处理器核状态。

[0169] 在本发明实施例中,还进一步对所述高级可编程中断寄存器进行扩展,从而进一步设置数据交换寄存器(SCDXR)(图中未示出),所述数据交换器寄存器中可以配置用于实现安全处理器核与普通处理器核实现数据交互的安全交互内存的内存地址范围,从而可以将需要交互的数据写入该安全交互内存中,实现安全处理器核与普通处理器核的数据交互。

[0170] 基于上述计算机架构,本发明实施例进一步提供了一种数据交互方法,应用于安全处理器核与普通处理器核间的交互,参考图11示出的数据交互流程示意图,所述数据交互流程包括:

[0171] 步骤S20:普通处理器核获取安全处理器核的安全交互内存的内存地址,并将待交互数据写入所述安全交互内存;

[0172] 具体的,普通处理器核可以通过总线(例如控制总线)向安全处理器核发起remote read类型的中断消息,从而读取SCDXR寄存器,获取安全交互内存的地址和大小。其中,所述安全交互内存可以称为REQ/RSP buffer。

[0173] 具体的,可以将待交互的数据写入REQ/RSP buffer内,然后利用自定义的中断消息,例如基于TEE(Trusted Execution Environment可信执行环境)的中断消息IPI TEE request,通知安全处理器核执行相应处理。REQ/RSP buffer可以为多种形式组织,比如环形队列,先入先出队列等,与TEE软件实现细节相关,TEE软件细节不再本设计考虑范围内。

[0174] 步骤S21:安全处理器核根据所述安全交互内存中的待交互数据,并生成响应数据,并将所述响应数据写入所述安全交互内存;

[0175] 其中,安全处理器核可以基于总线传输的中断消息,读取所述安全交互内存中的待交互数据,从而获取所述待交互数据,并进一步根据该待交互数据,生成对应的响应数据。

[0176] 需要说明的是,为保障安全处理器核的安全,安全处理器核并不会响应所有类型的中断消息。具体的,安全处理器核屏蔽总线传输的初始化中断消息和启动中断消息。

[0177] 在生成对应的响应数据后,安全处理器核进一步将所述响应数据写入安全交互内存REQ/RSP buffer中,并发送中断消息,通知所述普通处理器核。例如基于TEE(Trusted

Execution Environment可信执行环境)的中断消息IPI TEE response,通知普通处理器核执行相应处理。

[0178] 步骤S22:普通处理器核获取所述响应数据;

[0179] 普通处理器核可以基于中断消息,读取所述安全交互内存中的响应数据。

[0180] 在一个可选的示例中,参考图12示出了一种安全处理器核和普通处理器核的交互示例图,其中,普通处理器核的操作系统可以在执行支付、电子合同或区块链等业务时,基于REQ/RSP buffer,与安全处理器核进行交互,安全处理器核则基于TEE操作系统中的可信平台模块(TrustedPlatform Module,TPM)、可信执行环境或密钥管理(Key Manage)等业务,反馈相应的交互数据,从而实现业务的可信管理。

[0181] 基于上述计算机架构,本发明实施例进一步提供了一种处理器核的安全启动流程,参考图13示出的安全处理器核的安全启动流程示意图,所述安全启动流程包括:

[0182] 步骤S30:基于安全模块中的密钥信息,执行密钥验证;

[0183] 具体的,安全启动阶段,首先执行安全启动的初始化(bootrom)进程,执行对安全模块中的密钥信息的密钥验证。

[0184] 需要说明的是,在系统上电启动时,可以设置安全处理器核的指令指针(IP)寄存器指向安全模块的安全启动控制单元(bootrom),使系统启动首先执行bootrom的代码,并锁定其他普通处理器核。

[0185] 具体的,参考图14所示的步骤S30的流程示意图,步骤S30可以包括:

[0186] 步骤S301:获取安全模块中BOOTROM的代码,并在安全模块的安全启动存储单元(例如SRAM)中建立代码运行所需的内存堆栈;

[0187] 步骤S302:读取安全模块的密钥存储单元(例如EFUSE)中的固件密钥信息;

[0188] 步骤S303:读取服务提供商接口存储单元(SPI-ROM,其中SPI为Service Provider Interface的缩写;ROM为Read-Only Memory的缩写,只读存储器)中的启动装载(bootloader)代码,并执行所述启动装载进程的签名校验;

[0189] 其中,可以通过主桥读取所述启动装载(bootloader)代码。具体的,基于总线传输相应的访问请求至所述主桥,从而通过主桥读取启动装载(bootloader)代码。

[0190] 进一步的,基于所述启动装载代码,可以执行所述启动装载进程的签名校验。其中,在校验失败时,执行异常退出;在校验成功时,执行步骤S31。

[0191] 可以理解的是,在安全处理器核与安全模块的数据交互过程中,基于总线进行相应的数据传输,同时,安全模块会基于安全处理器核发出的访问请求,确定访问地址为安全访问地址,进而允许安全处理器核获取或读取其中的代码或信息,保证安全处理器核与安全模块的数据交互安全。

[0192] 步骤S31:配置所述安全处理器核的安全专有内存和安全交互内存,校验并加载基本输入输出系统;

[0193] 具体的,执行安全启动阶段初始化进程后,执行片外初始化(off chip bootloader)进程。

[0194] 具体的,参考图15所示的步骤S31的流程示意图,步骤S31可以包括:

[0195] 步骤S311:执行启动装载(bootloader)代码;

[0196] 基于读取的启动装载代码,执行所述启动装载进程。

[0197] 步骤S312:配置所述主桥的专有寄存器,指定安全专有内存的内存范围;

[0198] 其中,厂商可以根据安全处理器核实际运行的可信安全服务的内存需求来设置。

[0199] 具体的,安全处理器核可以基于总线传输相应的指令至所述主桥,从而配置所述主桥的专有寄存器,指定安全专有内存的内存范围。

[0200] 步骤S313:向内存控制器申请安全交互内存,并将申请的安全交互内存的内存地址范围写入数据交换寄存器(SCDXR);

[0201] 其中,安全交互内存用于实现安全处理器核与普通处理器核之间的交互。在普通处理器核需要与安全处理器核交互时,普通处理器核和读取数据交互寄存器中的安全交互内存的地址,并基于该安全交互内存的地址执行相应的交互流程。

[0202] 步骤S314:读取服务提供商接口存储单元(SPI-ROM)中的BIOS(Basic Input Output System,基本输入输出系统)代码,若安全启动进程(secure boot)使能,则校验BIOS,失败则异常退出。

[0203] 其中,可以通过主桥读取所述BIOS代码。具体的,基于总线传输相应的访问请求至所述主桥,从而通过主桥读取BIOS代码。

[0204] 进一步的,基于所述BIOS代码,可以执行所述BIOS的校验。其中,在校验失败时,执行异常退出;在校验成功时,执行步骤S315。

[0205] 步骤S315:加载BIOS到内存的指定地址;

[0206] 其中,所述内存的指定地址通常为首个加载的普通处理器核的初始IP指针所指向的地址,从而可以在后续执行普通处理器核的启动。

[0207] 步骤S32:释放并启动普通处理器核。

[0208] 其中,基于普通处理器核在系统上电时处于锁定状态,通过释放普通处理器核,以执行普通处理器核的加载进程。

[0209] 需要说明的是,在普通处理器核的加载过程中,需要首先释放并启动首个加载的普通处理器(可以称为BSP),在BSP启动后,才能执行其他普通处理器的加载启动。

[0210] 可以理解的是,在执行上述安全启动方法后,可以进入普通处理器核的初始化进程,在本发明实施例中,进一步提供了普通处理器核的安全启动进程,所述进程包括BSP初始化进程和剩余普通处理器(AP)初始化进程。

[0211] 具体的,所述BSP初始化进程包括:

[0212] 步骤S40:BSP执行系统加载程序,初始化硬件资源;

[0213] 其中,系统加载程序可以为BIOS程序或UEFI程序。其中,UEFI(Unified Extensible Firmware Interface,统一可扩展固件接口)程序是一种用于操作系统自动从预启动的操作环境加载到一种操作系统上的程序。

[0214] 步骤S41:执行操作系统启动装载(QS bootloader),选择待启动操作系统;

[0215] 步骤S42:操作系统初始化,并发送启动中断消息到AP处理器核。

[0216] 其中,通过发送启动中断消息(下称SIPI)到AP处理器核,以使AP处理器核执行初始化流程。

[0217] 具体的,所述AP初始化进程包括:

[0218] 步骤S50:响应接收的SIPI后,执行初始化工作;

[0219] 本发明实施例提供的计算机架构,安全处理器和与普通处理器核均基于所述总线

执行数据交互，其中，安全处理器核产生的访问地址为与普通访问地址不同的安全访问地址，从而可以在系统中基于不同类型的访问地址实现地址的区分，使安全处理器核实现基于总线进行数据交互的同时，还为安全处理器核的数据交互提供数据隔离的基础，保证安全处理器的信息安全。也就是说，本发明实施例在保证安全处理器的安全的前提下，采用系统内的总线执行信息传输，从而提高了数据传输的效率，进而提高了系统的性能。

[0220] 虽然本发明实施例披露如上，但本发明并非限定于此。任何本领域技术人员，在不脱离本发明的精神和范围内，均可作各种更动与修改，因此本发明的保护范围应当以权利要求所限定的范围为准。

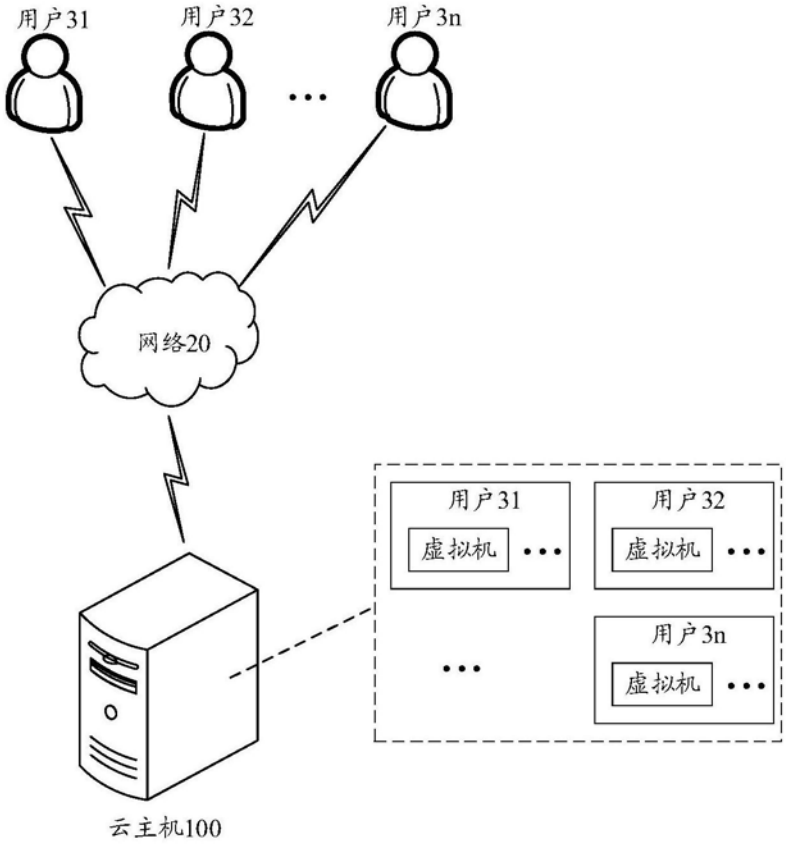

用户31
用户32
用户3n
网络20
用户31
虚拟机
用户32
虚拟机
用户3n
虚拟机
云主机100

图1

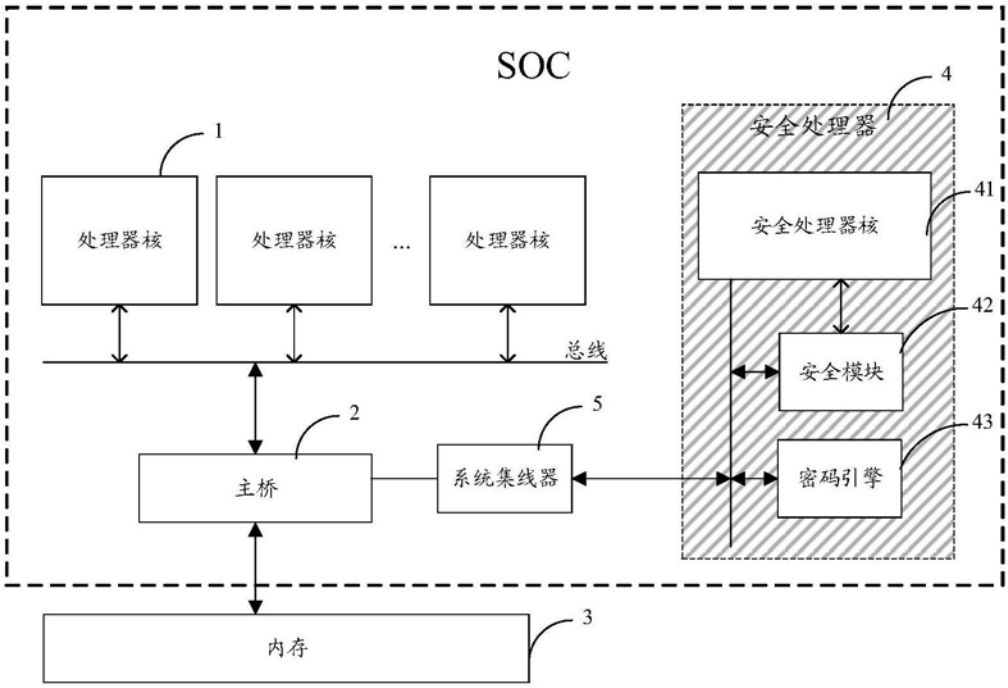

SOC
4
安全处理器
1
处理器核
处理器核
处理器核
41
安全处理器核
42
安全模块
总线
2
5
43
主桥
系统集线器
密码引擎
3
内存

图2

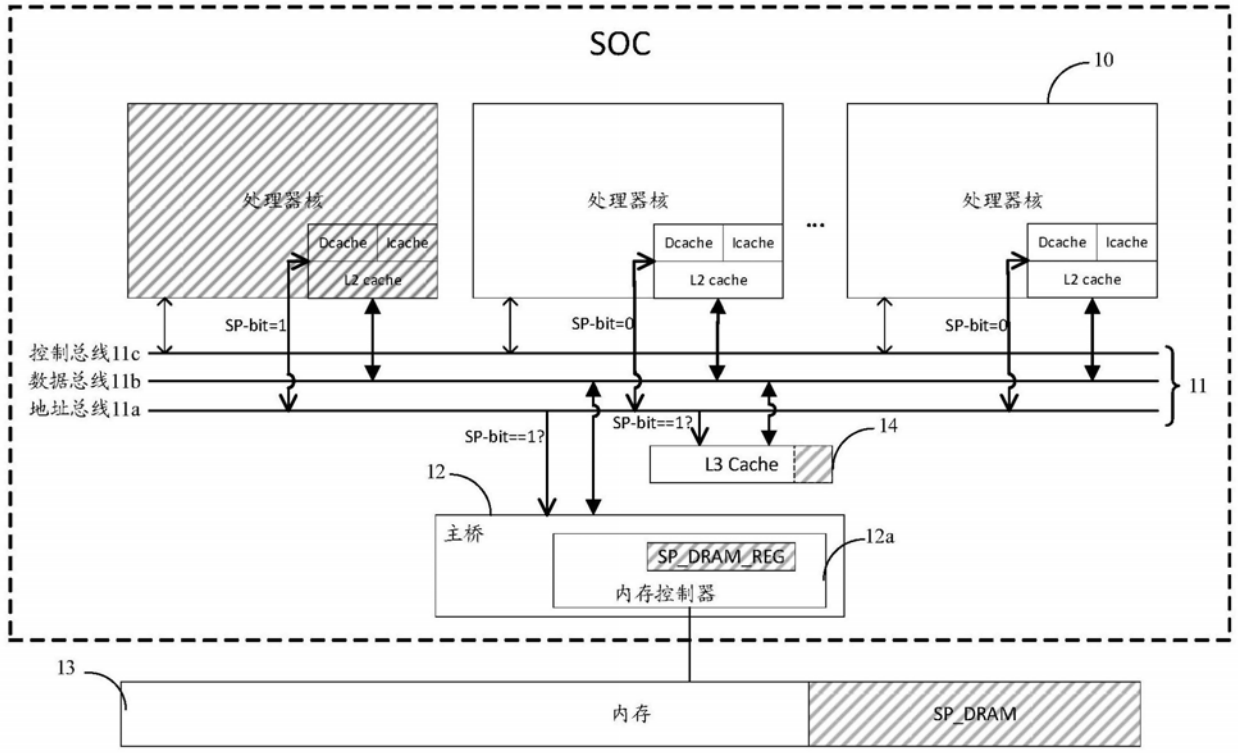

SOC
10
处理器核
Dcache
Icache
L2 cache
处理器核
Dcache
Icache
L2 cache
...
处理器核
Dcache
Icache
L2 cache
SP-bit=1
SP-bit=0
SP-bit=0
控制总线11c
数据总线11b
地址总线11a
11
SP-bit==1?
SP-bit==1?
14
L3 Cache
12
主桥
SP_DRAM_REG
12a
内存控制器
13
内存
SP_DRAM

图3

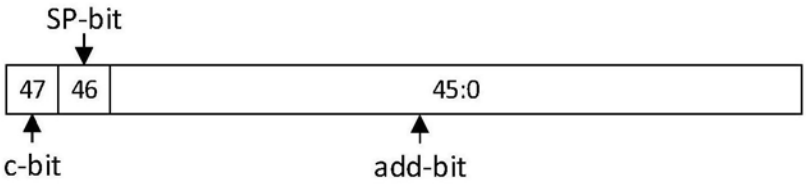

SP-bit
47
46
45:0
c-bit
add-bit

图4

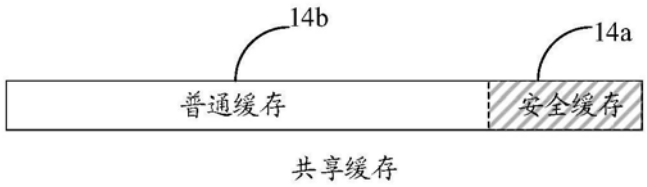

14b
14a
普通缓存
安全缓存
共享缓存

图5

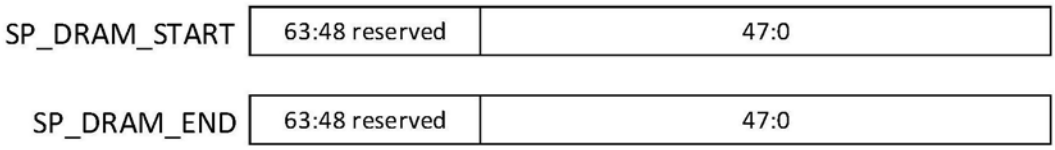

SP_DRAM_START
63:48 reserved
47:0
SP_DRAM_END
63:48 reserved
47:0

图6

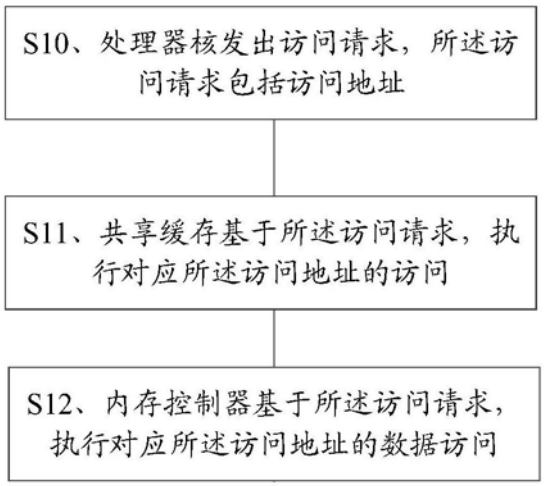

S10、处理器核发出访问请求，所述访问请求包括访问地址
S11、共享缓存基于所述访问请求，执行对应所述访问地址的访问
S12、内存控制器基于所述访问请求，执行对应所述访问地址的数据访问

图7A

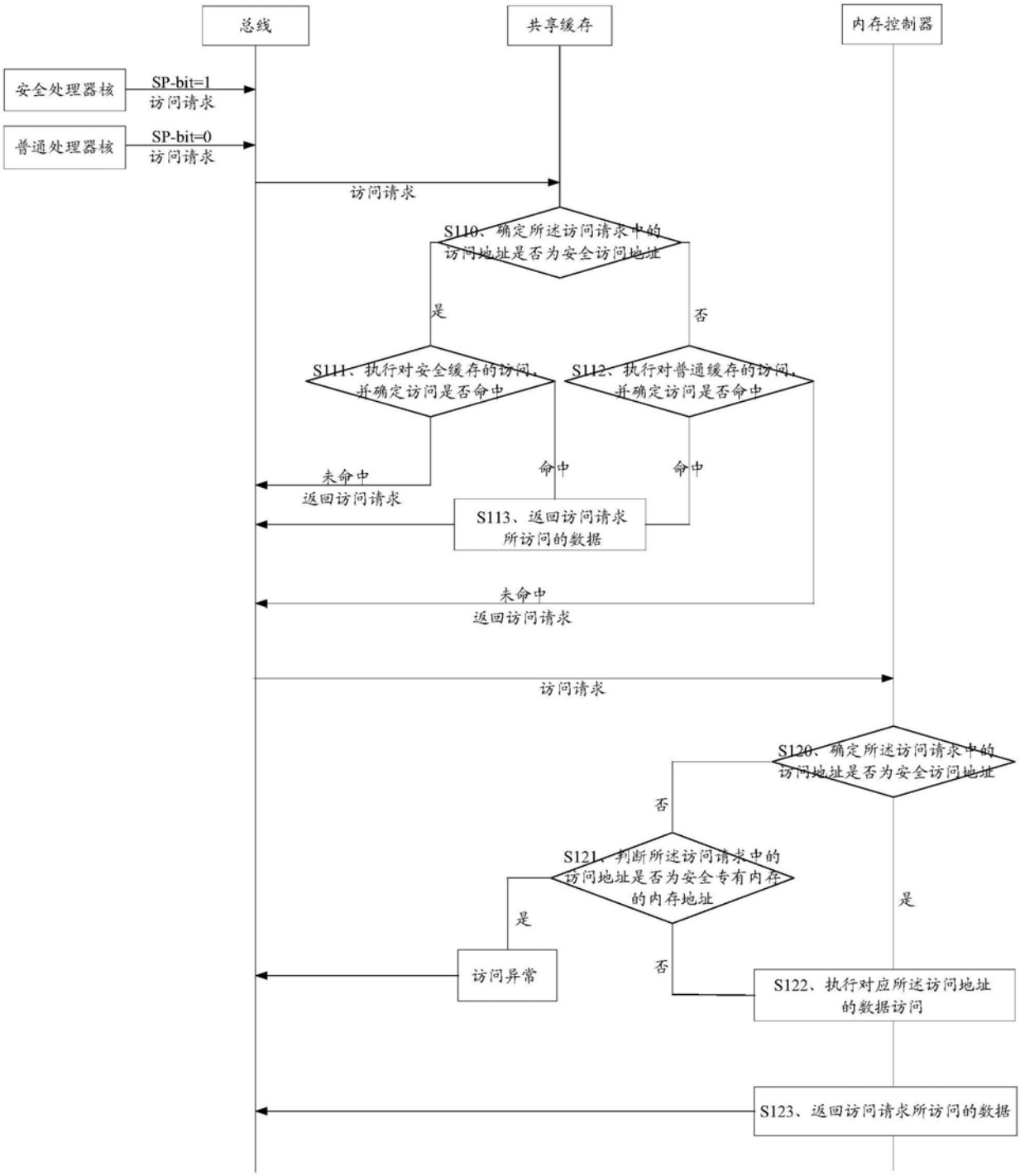

总线
共享缓存
内存控制器
安全处理器核
SP-bit=1
访问请求
普通处理器核
SP-bit=0
访问请求
访问请求
S110、确定所述访问请求中的访问地址是否为安全访问地址
是
否
S111、执行对安全缓存的访问，并确定访问是否命中
S112、执行对普通缓存的访问，并确定访问是否命中
未命中
返回访问请求
命中
命中
S113、返回访问请求所访问的数据
未命中
返回访问请求
访问请求
S120、确定所述访问请求中的访问地址是否为安全访问地址
否
S121、判断所述访问请求中的访问地址是否为安全专有内存的内存地址
是
是
否
访问异常
S122、执行对应所述访问地址的数据访问
S123、返回访问请求所访问的数据

图7B

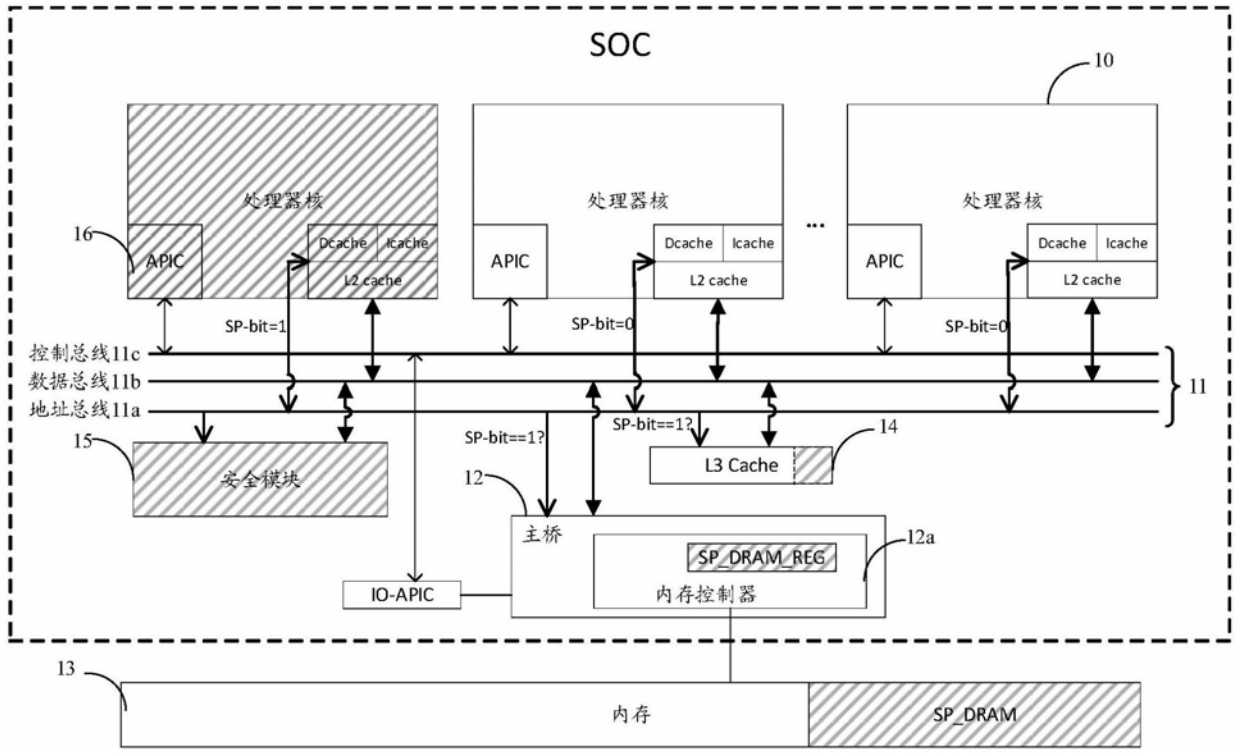

SOC
10
处理器核
处理器核
处理器核
16
APIC
Dcache
Icache
L2 cache
APIC
Dcache
Icache
L2 cache
APIC
Dcache
Icache
L2 cache
SP-bit=1
SP-bit=0
SP-bit=0
控制总线11c
数据总线11b
地址总线11a
11
15
安全模块
SP-bit==1?
SP-bit==1?
14
L3 Cache
12
主桥
SP_DRAM_REG
12a
内存控制器
IO-APIC
13
内存
SP_DRAM

图8

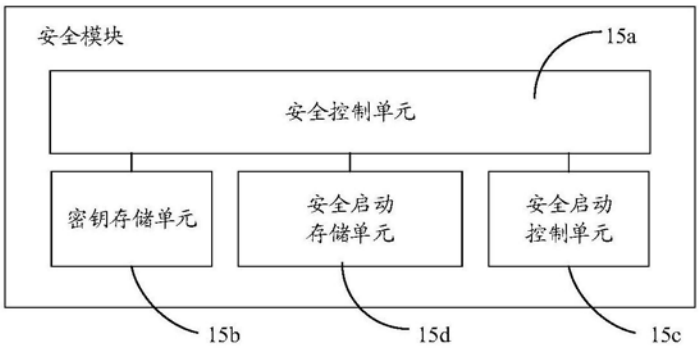

安全模块
15a
安全控制单元
密钥存储单元
安全启动
存储单元
安全启动
控制单元
15b
15d
15c

图9

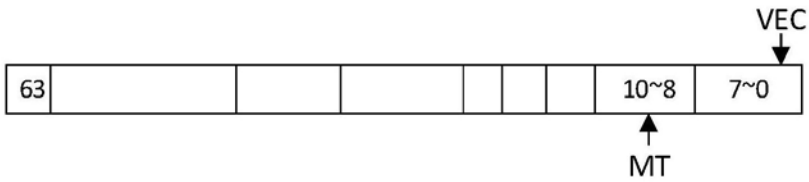

VEC
63
10~8
7~0
MT

图10

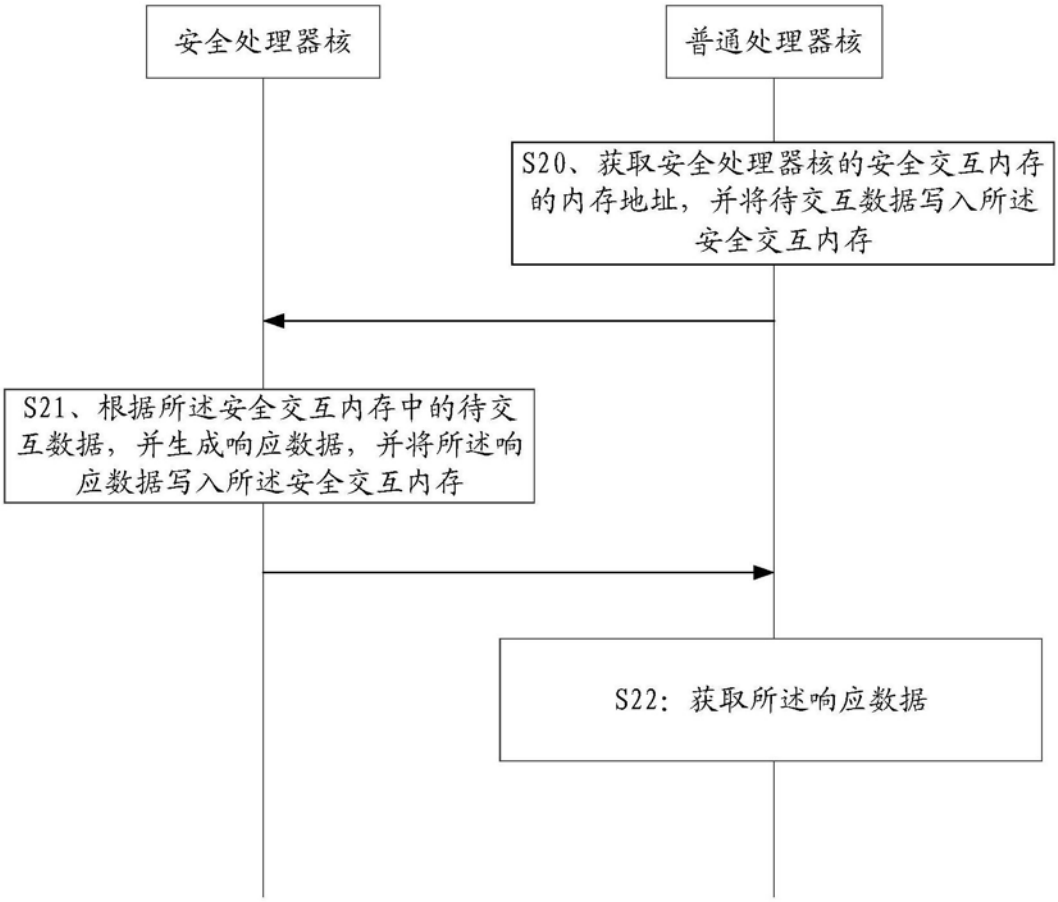

安全处理器核
普通处理器核
S20、获取安全处理器核的安全交互内存的内存地址，并将待交互数据写入所述安全交互内存
S21、根据所述安全交互内存中的待交互数据，并生成响应数据，并将所述响应数据写入所述安全交互内存
S22：获取所述响应数据

图11

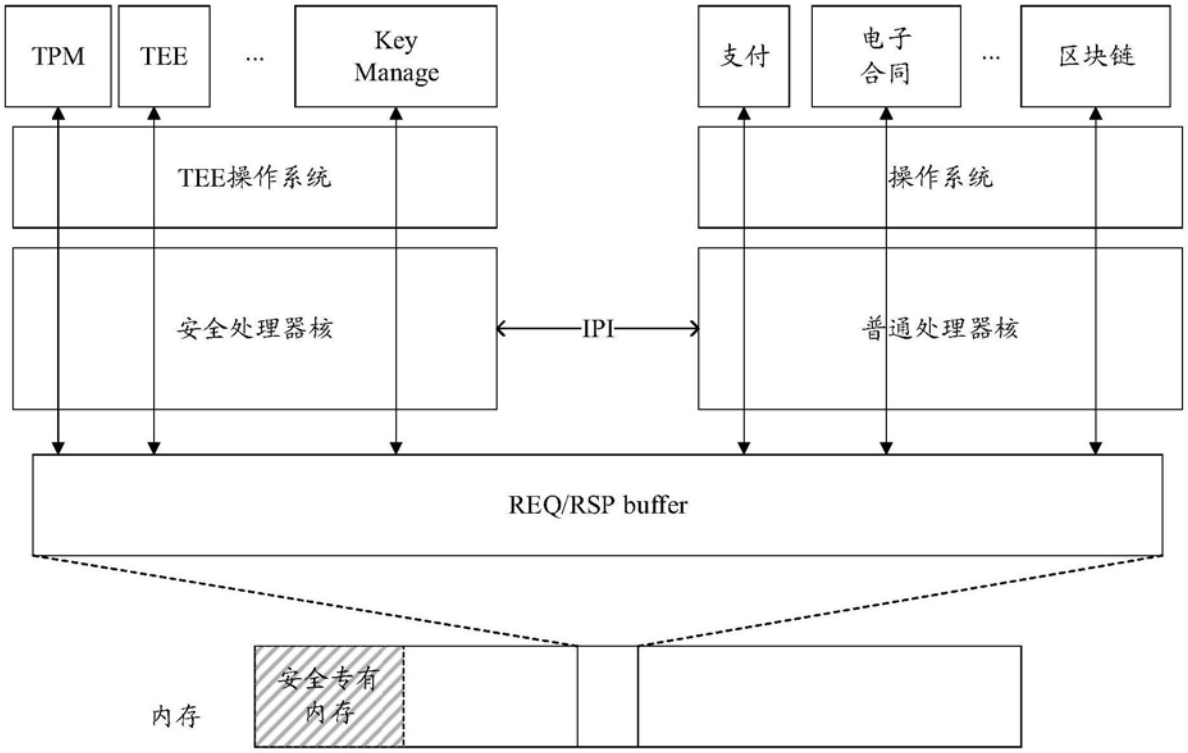

TPM
TEE
…
Key Manage
支付
电子合同
…
区块链
TEE操作系统
操作系统
安全处理器核
IPI
普通处理器核
REQ/RSP buffer
内存
安全专有内存

图12

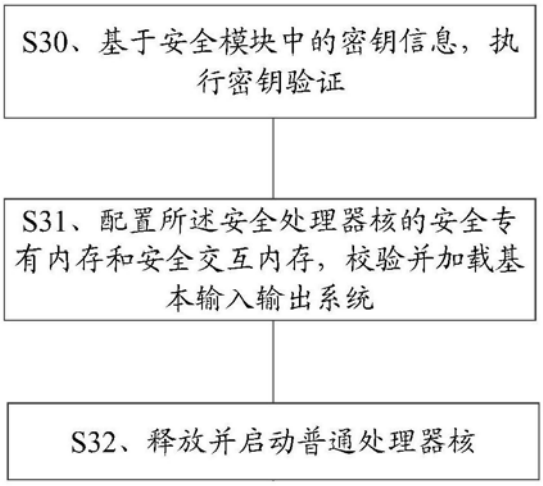

S30、基于安全模块中的密钥信息，执行密钥验证
S31、配置所述安全处理器核的安全专有内存和安全交互内存，校验并加载基本输入输出系统
S32、释放并启动普通处理器核

图13

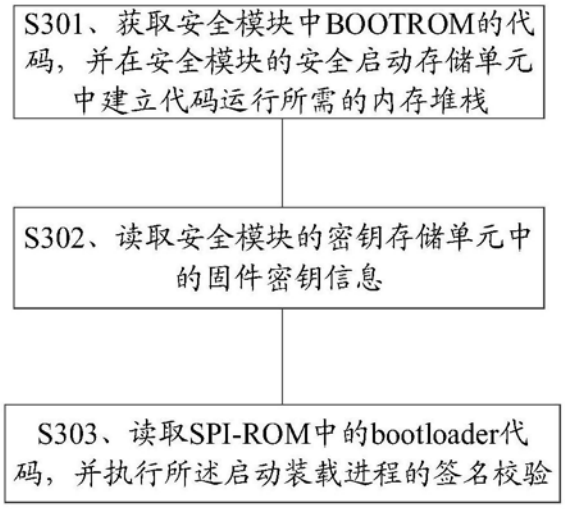

S301、获取安全模块中BOOTROM的代码，并在安全模块的安全启动存储单元中建立代码运行所需的内存堆栈
S302、读取安全模块的密钥存储单元中的固件密钥信息
S303、读取SPI-ROM中的bootloader代码，并执行所述启动装载进程的签名校验

图14

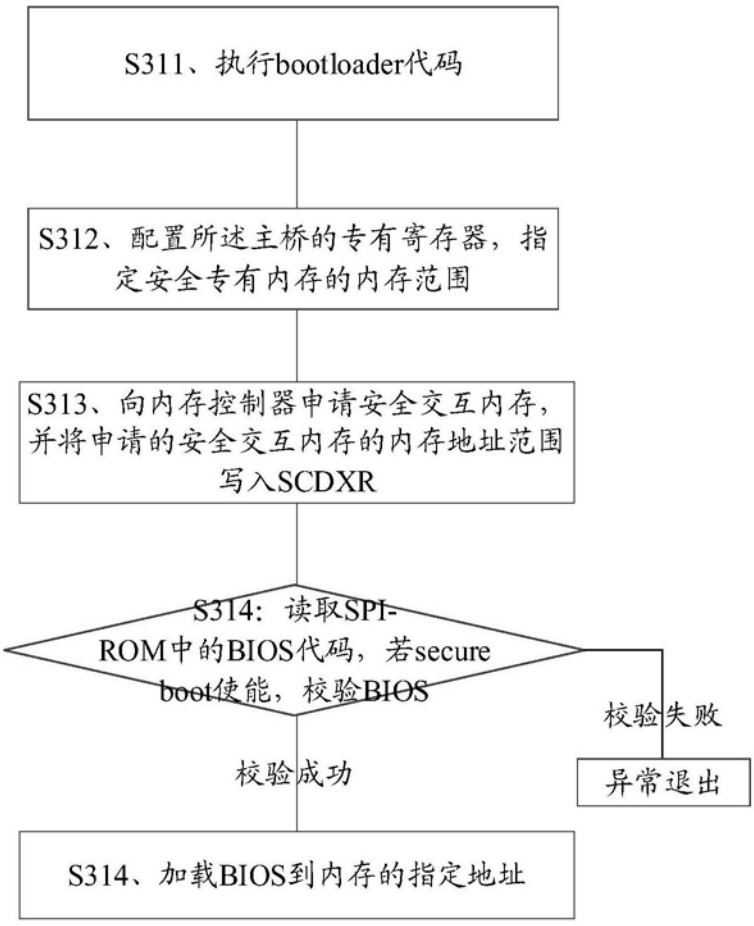

S311、执行bootloader代码
S312、配置所述主桥的专有寄存器，指定安全专有内存的内存范围
S313、向内存控制器申请安全交互内存，并将申请的安全交互内存的内存地址范围写入SCDXR
S314：读取SPI-ROM中的BIOS代码，若secure boot使能，校验BIOS
校验失败
异常退出
校验成功
S314、加载BIOS到内存的指定地址

图15