

(12) 发明专利

(10) 授权公告号 CN 101533447 B

(45) 授权公告日 2010. 12. 01

(21) 申请号 200910137752. 1

US 5615263 A, 1997. 03. 25, 全文.

(22) 申请日 2009. 04. 29

WO 0146800 A2, 2001. 06. 28, 全文.

(30) 优先权数据

审查员 曹妹妹

61/055, 980 2008. 05. 24 US

12/263, 177 2008. 10. 31 US

(73) 专利权人 威盛电子股份有限公司

地址 中国台湾台北县

(72) 发明人 G·葛兰·亨利 泰瑞·派克斯

(74) 专利代理机构 北京市柳沈律师事务所

11105

代理人 钱大勇

(51) Int. Cl.

G06F 21/02 (2006. 01)

(56) 对比文件

CN 1711524 A, 2005. 12. 21, 全文 .

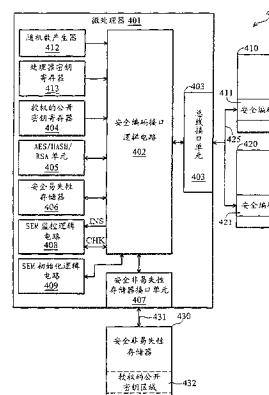
权利要求书 2 页 说明书 27 页 附图 12 页

(54) 发明名称

提供安全执行环境的微处理器及其执行安全
编码的方法

(57) 摘要

一种提供安全执行环境的微处理器装置，包
括安全非易失性存储器及微处理器。安全非易
失性存储器储存一安全应用程序。安全应用程序
根据加密演算规则被加密。微处理器通过私密总线
耦接安全非易失性存储器且通过系统总线耦接系
统存储器，以执行多个非安全应用程序与安全应
用程序。非安全应用程序通过系统总线而存取自
系统存储器。私密总线上的数据传输隔离于系统
总线以及微处理器内的多个对应系统总线资源。
微处理器包括加密单元，其配置在一执行逻辑电
路内，用以对安全应用程序进行加密以储存在安
全非易失性存储器，且对安全应用程序进行解密
以由微处理器来执行。



1. 一种微处理器装置,用以在安全执行环境中执行安全编码,该微处理器装置包括:

一安全非易失性存储器,用以储存一安全编码,其中,该安全编码根据一加密演算规则来被加密;以及

一微处理器,通过一私密总线耦接该安全非易失性存储器且通过一系统总线耦接一系统存储器,用以执行多个非安全编码与该安全编码,其中,该微处理器包括:

一总线接口单元,用以实现该系统总线上的多个系统总线数据传输,以存取该系统存储器内的所述非安全编码;

一安全非易失性存储器接口单元,用以通过该私密总线将该微处理器耦接至该安全非易失性存储器,其中,该私密总线上用来存取该安全非易失性存储器的多个私密总线数据传输被隐藏,以避免被该微处理器内的多个系统总线资源以及耦接该系统总线的任何装置所得知察觉;以及

一加密单元,配置在一执行逻辑电路内且耦接该安全非易失性存储器接口单元,用以对该安全编码进行加密以储存在该安全非易失性存储器,且对该安全编码进行解密以由该微处理器来执行。

2. 如权利要求 1 所述的微处理器装置,其中,该微处理器更包括:

一处理器密钥寄存器,耦接该加密单元,用以储存该微处理器所特有的一加密密钥,其中,在该微处理器的制造期间,该加密密钥被编程至该处理器密钥寄存器,且该加密密钥用来对该安全编码进行加密,以储存至该安全非易失性存储器。

3. 如权利要求 2 所述的微处理器装置,其中,该处理器密钥寄存器包括多个熔丝,所述熔丝完全地分布在一微处理器管芯上,且该处理器密钥寄存器只可由该加密单元来读取。

4. 如权利要求 2 所述的微处理器装置,其中,该加密单元产生该安全编码的一个或多个哈希,该加密单元对所述哈希进行加密,并将加密后的哈希储存至该安全非易失性存储器。

5. 如权利要求 1 所述的微处理器装置,其中,该安全编码是藉由使用一非对称加密密钥并根据一非对称密钥算法来被初始加密,该微处理器通过该系统总线自该系统存储器撷取非对称加密格式的该安全编码,且该微处理器利用该加密单元并根据该非对称密钥算法来对该安全编码进行解密,

其中该加密单元对根据该非对称密钥算法解密后的安全编码进行加密以储存在该安全非易失性存储器。

6. 如权利要求 5 所述的微处理器装置,其中,该微处理器更包括:

一公开密钥寄存器,耦接该加密单元,用以储存一公开密钥,该公开密钥被该加密单元来使用以对该安全编码进行解密。

7. 一种在安全执行环境中执行安全编码的方法,包括:

通过一私密总线将一安全非易失性存储器耦接至一微处理器,以储存一根据一加密演算规则来加密的安全编码,其中,该私密总线隔离于该微处理器内的所有系统总线资源且位于该微处理器的外部;

通过配置在该微处理器内的一加密单元,对该安全编码进行加密;

通过实现于该私密总线上的多个私密数据传输来将加密后的安全编码储存于该安全非易失性存储器,其中,该私密总线只由该微处理器内的一安全执行逻辑电路所得知与存

取；以及

通过该加密单元对该安全编码进行解密以由该微处理器来执行。

8. 如权利要求 7 所述的在安全执行环境中执行安全编码的方法，更包括：

存取一处理器密钥寄存器，该处理器密钥寄存器耦接该加密单元，用以储存该微处理器所特有的一加密密钥，其中，在该微处理器的制造期间，该加密密钥被编程至该处理器密钥寄存器，且该加密密钥用来对该安全编码进行加密，以储存至该安全非易失性存储器。

9. 如权利要求 8 所述的在安全执行环境中执行安全编码的方法，其中，该处理器密钥寄存器包括多个熔丝，所述熔丝完全地分布在一微处理器管芯上，且该处理器密钥寄存器只可由该加密单元来读取。

10. 如权利要求 8 所述的在安全执行环境中执行安全编码的方法，更包括：

通过该加密单元来产生该安全编码的一个或多个哈希、对所述哈希进行加密、并将加密后的哈希储存至该安全非易失性存储器。

11. 如权利要求 8 所述的在安全执行环境中执行安全编码的方法，其中，该安全编码是藉由使用一非对称加密密钥并根据一非对称密钥算法来被初始加密，该微处理器通过该系统总线自该系统存储器撷取非对称加密格式的该安全编码，且该微处理器利用该加密单元并根据该非对称密钥算法来对该安全编码进行解密，

其中该加密单元对根据该非对称密钥算法解密后的安全编码进行加密以储存在该安全非易失性存储器。

12. 如权利要求 11 所述的在安全执行环境中执行安全编码的方法，更包括：

存取一公开密钥寄存器，其中，该公开密钥寄存器耦接该加密单元，用以储存一公开密钥，该公开密钥被该加密单元来使用以对该安全编码进行解密。

提供安全执行环境的微处理器及其执行安全编码的方法

技术领域

[0001] 本发明涉及微电子领域,特别是涉及一种微处理器,其提供一安全执行模式的操作,其允许在微处理器内的安全环境中执行运算码。

背景技术

[0002] 桌上型计算机、笔记型计算机、以及手持式计算机与通信装置可作为机密或专用数据与数字权利控制内容的数字通信平台,计算机产业对于这些装置的使用持续地发展新的安全制度。举例来说,有许多已建立的应用,用以在因特网上免费下载与管理数字声音与影音档案。通过这些应用,使用者被提供在歌曲、电视节目以及电影上的有限的权利。特别注意的是,以上通过使用建立在这些应用中的安全特性来保护这些权利,而这些安全特性通常依据其主机平台所提供的安全机制。

[0003] 除了数字内容权利的保护,持续驱动计算机系统安全性的另一因素是实施在主机平台本身的使用限制。目前已知,手机产业已提供特定通信装置中所谓的“随用随付(Pay-as-you-go)”使用。藉由使用此方案,使用者不需给付月费,但是需预先给付某通话分钟数的金额。当用尽通话分钟数时,除了紧急通话以外,使用者被拒绝存取任何关于通话的手机网络存取。

[0004] 早在 2006 年, MICROSOFT 公司与其合作公司已提供主要指向新兴计算机市场的“随用随付”个人计算机。在此体制下,通过预付卡的购得,当使用这些公司的计算机时使用者则给付费用。此外,归属于 MICROSOFT 公司的美国专利申请案公开编号 20060282899, 公开一种用于模块化操作系统的传递的系统与方法,其包括提供主要操作系统支持的核心功能模块或基础核心,且包括一或多个允许客制化的操作系统定做的附属模块。在此应用中,附属模块可提供对于计算机(其包括硬件、应用软件、接口设备、以及支持设备)的支持或延伸能力。在设置之前,数字签章可使用来确定附属模块的完整性,且核对证明(certification)以判断附加模块的设置是否经过授权。藉由此证明,服务提供者可管理对提供的计算机上的非法或非期望修改。此外,数字权利管理可用来执行与许可配置相配的附属模块的使用项目。

[0005] 并不意外地,目前已发展出技术方法的真正主机,其提供规避安全措施,而这些安全措施是适当地保护且控制对权利控制数字媒体、通信装置、以及计算机平台的存取。最近,“hacking(进行非法入侵,即黑客)”变成研究上的课题。事实上,本案发明人已注意到许多用来篡改或完全地使安全管理无效的作品公开,而这些安全管理用来防护受保护资产的存取及 / 或使用。由 Andrew Huang, San Francisco :No Starch Press, 2003 所提出的著作 Hacking theXbox :An Introduction to Reverse Engineering 则是上述作品的一种。此著作特别着重于教导非法入侵技术以克服 MICROSOFT 所出产的 XBOX 游戏平台的安全机制,且更提供计算机安全与反向工程的教导主题,并讨论所谓“安全的”计算机平台的弱点。

[0006] 因此,平台建置者与设计者持续从事在避免未被授权的平台处理上更有效的技术与机制,不论此存取是良性的(例如探测或窥察)、恶意的(例如破坏性的或违背权利的入

侵)、或是介于两者之间(例如篡改)。这些机制中许多者用来防止入侵者实际上存取平台,例如将平台放置在安全底座上(例如一上锁的金属围场)或者将有弱点的电路封装入环氧化物内。但是已知这些类型的技术增加了系统成本与复杂性。其它机制则利用特定计算机架构本身提供的安全特性。

[0007] 考虑已知 x86 架构所提供的两个主要安全特性 :分页虚拟存储器 (paged virtual memory) 以及特许执行 (privileged execution)。在分页虚拟存储器的情况下,基本的操作系统定义一个分别的虚拟位置空间以及存取权利(例如只执行、只读取)给每一正被执行的应用程序,因此阻止另一秘密鬼祟的应用程序在所定义的区域内执行,且阻止其修改数据。但是,由于与虚拟地址译文相关(即分页窗体)的数据存在于系统存储器,且其出现于主机微处理器外的系统总线上,因此此数据可轻易地被窥察且被改变。

[0008] 在特许执行的情况下, x86 结构提供数种阶级的执行特权 CPL0 至 CPL3。因此,某些系统资源与指令只可由正在较高特权阶级上执行的应用程序来存取。一般得知操作系统组件操作在最高特权阶级 CPL0, 以及使用者应用归类于最低特权阶级 CPL3。但是,熟知此技术领域的人士将查知,这些架构特征主要是发展来阻止软件错误所导致的系统当机,且在防止有意或经指导的侵入 (directed hacks) 而言不是非常有效。

[0009] 因此已发展多种方法与装置,其更仔细地集中防止对平台的有意侵入与接管。在美国专利编号 5615263 中, Takahashi 教导一种在双模 (dual mode) 处理器中的安全模式。在一般 / 外部模式中,此双模处理器执行由外部来源所提供的指令。这些指令通过双模处理器的输入 / 输出来提供给双模处理器。当接收到专用软件或硬件发出的中断时,此双模处理器进入安全 / 内部模式。此中断是指储存在双模处理器中只读存储器内的安全功能。根据此接收的中断,双模处理器的输入 / 输出被禁能。此已确认的安全功能由双模处理器来执行。在此安全功能的执行期间,欲插置非来自只读存储器的指令的任何企图皆被忽略。然而,双模处理器可存取由正在执行的安全功能所特别确认的数据。当安全功能执行完成,则执行一退出程序,以使能双模处理器的输入 / 输出,并通过输入 / 输出重新开始执行由双模处理器的外部来源所提供的指令。

[0010] Takahashi 教导此安全模式是用作加密与解密,且其中双模处理器处理通过总线且由外部控制信道 (external control channel) 处理器所提供的正常指令与数据,其中,此总线符合一标准总线架构,例如工业标准体系结构 (Industry Standard Architecture, ISA)。此双模处理器在非安全模式下开启,且安全模式通过软件或硬件发出的中断来初始化。在安全模式下,可执行关于加密与解密的有限数量的功能(即指令)。这些功能储存在一个只读存储器中 (ROM),其位于双模处理器的内部。本案的发明人注意到, Takahashi 的双模处理器并不适当,因为 Takahashi 的双模处理器只能执行内部 ROM 所提供的有限数量的功能。因此,包括一般目的指令的应用程序(即在微处理器的指令集中任何的指令)则无法在安全模式下执行。

[0011] 在美国专利编号 7013484 中, Ellison 揭露一种建立安全环境的芯片组,用于一隔离的储存器所执行的隔离执行模式,此储存器被至少一处理器来存取。在正常执行模式或此隔离执行模式下,此至少一处理器具有多线程与操作。Ellison 的安全环境依据一外部芯片组(被隔离的执行电路),其提供机制给一处理器以在隔离执行模式下操作。此外部芯片组因此配置一个安全存储器区域,其管理隔离指令的译码与转译、隔离总线周期的产生、以

及中断的产生。当此外部芯片组主动地隔离存储器区域、指令执行等时，注意到此外部芯片组通过一般系统总线而耦接此至少一处理器，因此在任何安全线程的执行期间内容许在总线上的窥察与流量篡改。

[0012] 在美国专利编号 7130951 中，Christie 揭露一种方法，用以控制有安全执行模式能力之处理器，此处理器包括多中断，以使得当其正操作在非安全执行模式时，中断此有安全执行模式能力的处理器。此方法包括当此有安全执行模式能力的处理器正操作在一安全执行模式时，禁能多中断以避免此处理器中断。尽管禁能中断是在安全执行环境中所期望的安全特性，根据 Christie 的处理器处理通过系统总线且由一操作系统所提供的指令与数据。一旦这些指令被提供时，中断即被禁能。如同 Ellison 的机制，此一装置明确地可被通过总线而提供至处理器的指令来做总线窥察与篡改。

[0013] 在美国专利编号 6983374 中，Hashimoto 揭露一种抗篡改微处理器，其保存关于其执行将被中断的一个程序的内容信息，其中，此处理器状态被加密且储存在系统存储器。Hashimoto 也教导了自系统存储器撷取加密指令的技术，以及对加密指令进行解密且执行此加密指令的装置。此外，Hashimoto 教导了使用一对密钥来提供在存储器内的加密指令，且接着使用非对称密钥算法来对储存在存储器内的对称密钥加进行加密。对于程序创造者来说，对称密钥是已知的，且使用读取自处理器的公开密钥来对此对称密钥进行加密。此处理器包括一独特私密密钥，其对应此公开密钥，且使用者无法存取。因此，根据分支指令的执行，程控被转移成“起始加密执行”指令，其传送一指标至加密对称密钥。此处理器撷取加密对称密钥，且使用其内部私密密钥来对其解密。接着，加密程序指令自系统存储器被撷取，且藉由使用解密对称密钥来被解密，并由处理器来执行。假使发生中断或异常，处理器的状态则对称地被加密且储存至存储器。Hashimoto 揭露了对于非加密与加密编码的共通快取机制、中断逻辑、异常处理逻辑的使用。

[0014] 本案的发明人已注意到，Hashimoto 的微处理器限定编码者已知对应安全编码的对称密钥，且对称密钥可能被泄漏，因此，将具有此编码的所有系统将有被攻击的风险。此外，本案的发明人已注意到，Hashimoto 的微处理器缺点在于，必须在撷取指令运作中执行安全编码的解密，其花费非常多的时间，因此导致微处理器的处理能力变为缓慢。此外，注意到，Hashimoto 的安全编码利用现存的非安全资源，例如系统存储器、分页窗体、中断、与异常机制，这些全部都会遭受到窥察。

[0015] 因此，本案的发明人了解，显然期望提供一种微处理器，其能在安全执行环境中执行包括一般目的指令（即在微处理器的指令集中任何的指令）的应用程序或应用线程。

[0016] 此外，同时也期望此安全执行环境隔离于任何已知的窥察与篡改方法。因此，需要由一安全执行模式微处理器来执行指令，且此安全执行模式微处理器隔离于处理器中提供存取（例如快取窥察、系统总线流量、中断、以及错误与追踪特征）的硬件。

[0017] 此外，更期望当此微处理器加载应用程序并安全执行时，提供一机制来混淆来自任何现存监控装置的应用的结构与内容，且提供一机制来证明此应用的来源且确认其诚实性。

发明内容

[0018] 本发明适用于解决前述问题与对付习知技术的其它问题、缺点与限制。本发明提

供较佳的技术,以在一般目的微处理器平台上使能安全应用程序的执行。在一实施例中,揭露一种提供安全执行环境的微处理器装置,用以在安全执行环境中执行安全编码,此微处理器装置包括安全非易失性存储器以及微处理器。安全非易失性存储器储存一安全应用程序。安全应用程序根据一加密演算规则来被加密。微处理器通过私密总线耦接安全非易失性存储器且通过系统总线耦接系统存储器,用以执行多个非安全应用程序与安全应用程序。微处理器包括总线接口单元、安全非易失性存储器接口单元以及加密单元。总线接口单元实现系统总线上的多个系统总线数据传输,以存取系统存储器内的非安全应用程序。安全非易失性存储器接口单元通过该私密总线将微处理器耦接至安全非易失性存储器。私密总线上用来存取安全非易失性存储器的多个私密总线数据传输被隐藏,以避免被微处理器内的多个系统总线资源以及耦接系统总线的任何装置所得知察觉。加密单元配置在一执行逻辑电路内且耦接安全非易失性存储器接口单元,用以对安全应用程序进行加密以储存在安全非易失性存储器,且对安全应用程序进行解密以由微处理器来执行。

[0019] 本发明的又一实施例提供一种在安全执行环境中执行安全编码的方法,包括:通过私密总线将安全非易失性存储器耦接至微处理器,以储存一安全编码,其中,私密总线隔离开于微处理器内的所有系统总线资源且位于微处理器的外部;通过配置在微处理器内的加密单元,对安全编码进行加密;以及通过实现于私密总线上的多个私密数据传输来将安全编码储存于安全非易失性存储器,其中,私密总线只由微处理器内的安全执行逻辑电路所得知与存取。

[0020] 关于产业应用性,本发明可实现于一微处理器内,且此微处理器使用于一般目的或特殊目的的计算机装置。

附图说明

- [0021] 图 1 表示根据本发明的安全执行模式 (SEM) 微处理器的方块示意图;
- [0022] 图 2 表示说明图 1 的微处理器中最高阶级操作模式的状态图;
- [0023] 图 3 表示根据本发明的微处理器中 SEM 逻辑电路的方块示意图;
- [0024] 图 4 表示在根据本发明的微处理器内,安全编码如何被储存、存取、初始化以及执行的方块示意图;
- [0025] 图 5 表示在图 1 的微处理器中,SEM 监控逻辑电路的详细方块示意图;
- [0026] 图 6 表示在根据本发明的微处理器内操作模式转换的状态图;
- [0027] 图 7 表示在本发明的微处理器中使能安全执行模式操作的高阶方法流程图;
- [0028] 图 8 表示在本发明的微处理器中禁能安全执行模式操作的高阶方法流程图;
- [0029] 图 9 表示在本发明的微处理器内初始化安全编码执行的方法流程图;
- [0030] 图 10 表示本发明微处理器中执行安全执行模式使能重置操作的方法流程图;
- [0031] 图 11 表示在本发明微处理器中终止安全执行模式操作的方法流程图;以及
- [0032] 图 12 表示在本发明的微处理器内安全实时时钟的详细方块示意图。主要组件符号说明

- [0033] 100 ~ 系统板; 101 ~ 安全执行模式微处理器;
- [0034] 102 ~ 系统总线; 103 ~ 总线主控装置;
- [0035] 104 ~ 总线管理装置;

- [0036] 105 ~ 安全执行模式逻辑电路；
- [0037] 106 ~ 私密总线； 107 ~ 安全非易失性存储器；
- [0038] C1、C2 ~ 连接路径 / 信号；
- [0039] PSNT ~ 存储器检测总线 / 信号；
- [0040] VP ~ 电池； VP1、VP2 ~ 连接路径 / 信号；
- [0041] X1 ~ 石英器；
- [0042] 200 ~ 状态图；
- [0043] 201 ~ 非安全执行模式（原生未受控模式）；
- [0044] 202 ~ 安全执行模式（SEM- 使能模式）；
- [0045] 203 ~ 降级模式； 204 ~ 硬件关机模式；
- [0046] 300 ~ 安全执行模式微处理器；
- [0047] 301 ~ SEM 逻辑电路； 302 ~ 安全易失性存储器；
- [0048] 303 ~ 处理器状态； 304 ~ 安全编码；
- [0049] 305 ~ SEM 初始化逻辑电路；
- [0050] 306 ~ SEM 监控逻辑电路；
- [0051] 307 ~ SEM 中断逻辑电路；
- [0052] 308 ~ SEM 例外逻辑电路；
- [0053] 309 ~ SEM 定时器； 310 ~ SEM 实时时钟；
- [0054] 311 ~ AES/HASH/RSA 单元；
- [0055] 312 ~ 处理器密钥寄存器；
- [0056] 313 ~ 处理器执行单元； 314 ~ 正常例外逻辑电路；
- [0057] 315 ~ 正常追踪 / 除错逻辑电路；
- [0058] 316 ~ 正常中断逻辑电路；
- [0059] 317 ~ 对应安全编码的安全数据；
- [0060] 318 ~ 授权的公开密钥寄存器；
- [0061] 319 ~ 随机数产生器；
- [0062] 320、321、324、326、327 ~ 总线；
- [0063] 322 ~ 电源管理逻辑电路；
- [0064] 323 ~ 地址逻辑电路； 325 ~ 非安全存储器；
- [0065] 328 ~ 非易失性使能指示寄存器；
- [0066] 329 ~ SEM 机械专用寄存器存储体；
- [0067] 400 ~ 图示； 401 ~ 微处理器；
- [0068] 402 ~ 安全编码接口逻辑电路；
- [0069] 403 ~ 总线接口单元；
- [0070] 404 ~ 授权的公开密钥寄存器；
- [0071] 405 ~ AES/HASH/RSA 单元；
- [0072] 406 ~ 安全易失性存储器；
- [0073] 407 ~ 安全非易失性存储器接口单元；
- [0074] 408 ~ SEM 监控逻辑电路； 409 ~ SEM 初始化逻辑电路；

- [0075] 410 ~ BIOS 存储器 ; 411、421 ~ 安全编码 ;
[0076] 412 ~ 随机数产生器 ; 413 ~ 处理器密钥寄存器 ;
[0077] 420 ~ 系统存储器 ; 425 ~ 系统总线 ;
[0078] 430 ~ 安全非易失性存储器 ;
[0079] 431 ~ 私密总线 ; 432 ~ 授权的公开密钥区域 ;
[0080] CHK、INS ~ 总线 ;
[0081] 500 ~ SEM 监控逻辑电路 ; 501 ~ 物理环境监控器 ;
[0082] 502 ~ 总线时钟监控器 ;
[0083] 503 ~ 频率参考单元 ; 504 ~ 处理器电压监控器 ;
[0084] 505 ~ 温度监控器 ; 506 ~ 数据监控器 ;
[0085] 507 ~ 安全时戳计数器 ;
[0086] 508 ~ 正常时戳计数器 ;
[0087] 509 ~ 比率机械专用寄存器 ;
[0088] 510 ~ 样式监控器 ; 511 ~ 指令监控器 ;
[0089] 512 ~ 指令阵列 ; 513 ~ 监控管理器 ;
[0090] BUSTERM、BUS CLK、CORE CLK、TEMP、VDD、CLASS1、CLASS2、CLASS3、DISABLE ~ 信号 ;
[0091] DESTRUCT、INS、NOBOOT、PINCHK、TAMPER、CHK ~ 总线 ;
[0092] 600 ~ 详细操作模式图标 ;
[0093] 601 ~ 原生未受控模式 (非安全执行模式) ;
[0094] 602 ~ SEM 使能重置模式 [1:N] ;
[0095] 603 ~ SEM 使能正常执行模式 ;
[0096] 604 ~ SEM 使能安全执行模式 ;
[0097] 605 ~ 降级模式 ; 606 ~ 硬件关机模式 ;
[0098] 700 ~ 流程图 ; 701...705 ~ 流程步骤 ;
[0099] 800 ~ 流程图 ; 801...806 ~ 流程步骤 ;
[0100] 900 ~ 流程图 ; 901...912 ~ 流程步骤 ;
[0101] 1000 ~ 流程图 ; 1001...1009 ~ 流程步骤 ;
[0102] 1100 ~ 流程图 ; 1101...1112 ~ 流程步骤 ;
[0103] 1200 ~ 安全实时时钟 ; 1201 ~ 振荡器 ;
[0104] 1202 ~ 计数器 ; 1203 ~ 转换逻辑电路 ;
[0105] VP、ENV ~ 信号 ; V0、CNT0 ~ 输出信号 ;
[0106] CNT0 ~ 输出信号 ;
[0107] TEMP、BATT、COMP、XTAL ~ 信号 ;
[0108] TIME ~ 双向总线。

具体实施方式

[0109] 为使本发明的上述目的、特征和优点能更明显易懂，下文特举一较佳实施例，并配合所附图式，作详细说明如下。

[0110] 本发明虽以较佳实施例揭露如上，然其并非用以限定本发明的范围，任何所属技术领域中具有通常知识者，在不脱离本发明的精神和范围内，当可做些许的更动与润饰，因此本发明的保护范围当视后附的权利要求书所界定者为准。

[0111] 鉴于上述关于在一微处理器中应用程序的安全并隔离地执行且关于用来防止窥察、侵入、篡改、或黑客的现今技术的背景讨论，本发明的讨论将通过图 1 至 12 来呈现。

[0112] 参阅图 1，其表示根据本发明实施例的安全执行模式 (secure execution mode, SEM) 微处理器 101 的示意图。此示意图描述 SEM 微处理器 101 配置所在的系统板 100 (或主机板)。此微处理器 101 通过系统总线 102 耦接一个或多个总线主控装置 (bus master) 103 以及 / 或者一个或多个总线管理装置 (bus agent) 104。在一实施例中，SEM 微处理器 101 为 x86 兼容微处理器 101，其通过 x86 兼容系统总线 102 耦接一个或多个 x86 兼容总线主控装置 103 以及 / 或者一个或多个 x86 兼容总线管理装置 104。

[0113] 此外，SEM 微处理器 101 耦接一电池 VP，其配置在系统板 (主机板) 100 上，且通过连接路径 VP1 与 VP2 来耦接至微处理器 101。在一实施例中，电池 VP 的电压为 1.8V 直流电压 (DC)。

[0114] 石英器 X1 也配置在系统板 100 上，且通过连接路径 C1 与 C1 来耦接至微处理器 101。微处理器 101 包括 SEM 逻辑电路 105。根据本发明的 SEM 逻辑电路 105 系配置来提供在微处理器内一安全执行模式的初始化、操作、以及终止，将于下文详细说明。此 SEM 逻辑电路 105 包括逻辑、电路、装置、或微码 (即微指令或原生指令)、或者是逻辑、电路、装置、或微码的结合、又或者是用来初始化安全执行模式的等效组件，使得 SEM 逻辑电路 105 可加载安全应用程序来执行、在一安全环境中执行这些应用程序、为了侦测且阻止篡改而监控一些微处理器与系统特性、在适当情况下终止安全执行模式、且假使侦测到篡改则暂停处理。用来执行这些功能与 SEM 逻辑电路 105 内其它功能的组件，可共享用来执行微处理器 101 内其它功能的其它电路、微码等等。根据本申请案的范围，微码是涉及多个微指令的名词。一微指令 (也称为原生指令) 是在一单元执行所处的层级上的指令。例如，微指令直接由精简指令集运算 (Reduced Instruction Set Computing, RISC) 微处理器来执行。对于复杂指令集运算 (Complex Instruction Set Computing, CISC) 微处理器 (例如 x86 兼容微处理器) 而言，x86 指令首先转译为相关的微指令，且此相关的微指令接着直接由 CISC 微处理器中一单元或多单元来执行。

[0115] 安全非易失性存储器 107 也配置在系统板 100 上，其通过私密总线 (PVTBUS) 106 与存储器检测总线 (presence detection bus) PSNT 来耦接至微处理器 101。根据本发明，安全非易失性存储器 107 为一种经过电源的除去与重新施加后其内容仍存留的存储器。即是，当提供至系统板的电源关闭或开启时，安全非易失性存储器 107 的内容不会改变。在一实施例中，安全非易失性存储器 107 包括快闪只读存储器 (ROM)，其大小相当于将在安全执行模式中执行的安全应用程序的大小。在一实施例中，考虑以 4MB 快闪只读存储器来作为安全非易失性存储器 107。在私密总线 106 上的数据传输 (transactions) 完全地隔离开于系统总线 102、总线主控装置 103 以及总线管理装置 104，且私密总线 106 位于微处理器 101 的外部。在一实施例中，快闪只读存储器 107 可程序化高达 100000 次。在一实施例中，私密总线 106 考虑以一串行总线来实现，其提供介于安全非易失性存储器 107 与微处理器 101 之间的数据传输。此私密总线 106 可符合标准界面协议，例如串行外围接口

(Serial Peripheral Interface, SPI) 协议。

[0116] 在操作上,电池 VP 与石英器 X1 提供在 SEM 逻辑电路 105 内实时时钟 (Real Time Clock, RTC) (未显示) 的持续操作,其将于下文详细说明。包括来自主机结构指令集的一或多个安全应用程序,通过系统总线 102 而撷取自系统存储器 (未显示),且储存在安全非易失性存储器 107。在一实施例中,使用属于授权者 (authorizing party) 的一私密非对称密钥并通过非对称加密演算规则来加密一个或多个安全应用程序,且安全应用程序以其非对称加密格式而被存取自系统存储器。在一实施例中,考虑通过 RSA 演算规则来加密一个或多个安全应用程序。在此一个或多个安全应用程序撷取自系统存储器后,微处理器 101 利用一对对应的公开密钥来解码此一个或多个安全应用程序并确认此一个或多个安全应用程序。根据安全执行模式的使能以及依据一“起始安全执行”指令的执行,SEM 逻辑电路 105 利用微处理器内的多加密资源,以根据一对称密钥算法并使用处理器独特加密密钥来对此一个或多个安全应用程序进行加密,此外,SEM 逻辑电路 105 通过私密总线 106 来将已加密的一个或多个安全应用程序传送至安全非易失性存储器 107。之后,SEM 逻辑电路 105 利用微处理器 101 内的多加密或其它资源,来对此一个或多个安全应用程序进行存取、确认以及解密,此一个或多个安全应用程序接着加载至微处理器 101 内的一安全且隔离的随机存取存储器 (RAM) 或一高速缓存 (未显示)。

[0117] 当执行起始安全执行指令时 (当进入至该安全执行模式),SEM 逻辑电路 105 禁能安全应用程序得知察觉所有的系统资源,而这些系统资源提供了包括非安全中断、非安全例外逻辑以及追踪 / 除错逻辑电路等等的监视以及或篡改。储存在隔离的内部 RAM 的一个或多个安全应用程序藉由使用 SEM 逻辑电路 105 内的专用安全执行资源来被执行。此一个或多个安全应用程序接着可将处理器状态由安全操作模式恢复至正常执行模式,或者假使侦测到潜在的篡改,他们可将微处理器转换至具有有限的功能的降级模式。假使确定发生篡改,SEM 逻辑电路 105 接着使微处理器完全地关机 (硬件关机模式)。

[0118] 关于此一个或多个安全应用程序 (或“安全编码”) 的功能类型包括 (但不受限于此) 执行关键安全任务,例如凭证确认、数据加密以及数据解密;监控正常系统软件活动;确认正常系统软件的完整性;追踪资源使用;新软件的安装。

[0119] 在一实施例中,在本发明的安全处理系统中考虑使用表面黏着式微处理器 101、表面黏着式安全非易失性存储器 107、以及表面黏着式石英器 X1。这些表面黏着式组件包括球栅阵列 (ball-grid array) 组件或焊接在系统板 100 上的其它相似技术。

[0120] 本发明的微处理器 101 也执行储存在系统存储器内 (未显示) 的非安全应用程序,这些非安全应用程序的指令通过系统总线 102 来提供。在本发明的观念中,微处理器 101 能如中央处理单元 (Centralized Processing Unit, CPU) 般操作,而不用因应协同处理器 (coprocessor) 的要求。即是,本发明的微处理器 101 能执行主机指令集的所有指令,且能执行全部的应用程序。与只能执行自一主要 CPU 转移的单一指令、程序线程或程序片断的类似功能协同处理器与处理器比较起来,本发明的微处理器 101 直接执行在对应应用程序中的所有指令,不论此应用程序是否是储存安全非易失性存储器 107 的安全应用程序或者是通过系统总线 102 撷取的非安全应用程序。

[0121] 接着参阅图 2,状态图 200 说明在图 1 的微处理器中最高阶级操作模式。在此最高阶级中,微处理器 101 提供三个主要操作模式 201-203 与一个硬件关机模式 204。非安全执

行模式 201 是在微处理器 101 制造后,当第一次供给电源时所默认 (default) 的第一个状态。非安全执行模式 201 也称为“原生未受控 (born free)”模式 201。原生未受控模式 201 是微处理器 101 的制造状态,其提供非安全应用程序的正常执行,其中,这些非安全应用程序通过系统总线 102 而于系统存储器中存取。在此状态中,无法得知且无法操作任何与安全应用程序的安全执行相关联的资源。这些资源包括 SEM 逻辑电路 105、安全非易失性存储器 107 以及一些其它专用寄存器,这些专用寄存器包括含有对称与非对称加密密钥、安全中断、安全存储器 (RAM) 以及其它硬件,将于下文详细说明。藉由提供原生未受控模式 201,可实施与非安全微处理器所共通的制造行动类型 (type of manufacturing activities)。此外,由于原生未受控模式 201 提供非安全应用程序的执行,因此本发明的微处理器 101 的相同的管芯设计 (the same die design) 可实施在非安全微处理器。在一实施例中,非安全微处理器的接脚配置 (pinout) 不同于 SEM 微处理器 101,且假使非安全微处理器配置在安全系统板 100 时,非安全微处理器的 SEM 逻辑电路 105 将因电源应用不同而无法操作。

[0122] 在一实施例中,SEMENABLE (SEM 使能) 指令的执行导致微处理器 101 的模式转换为安全执行模式 202。在安全执行模式 202 下,安全与非安全应用程序都可执行,但是非安全应用程序无法存取安全资源。安全执行模式 202 也称为 SEM- 使能模式 202。在一安全应用程序的控制下 (简称为程控),微处理器的状态可转换回原生未受控模式 201,然而,转换为原生未受控模式 201 的次数是有限的。在一实施例中,处理器转换回原生未受控模式可高达 64 次。在另一实施例中,以可确认的授权者来对特殊 (particular) 机械专用寄存器 (Machine Specific Register, MSR) 进行写入,导致微处理器 101 的模式转换为安全执行模式 202。

[0123] SEM 逻辑电路 105 监控对应微处理器且与潜在篡改相关状态,并根据这些状态的一使微处理器自安全执行模式 202 转换至降级 (操作) 模式 203。假使某些已定义的状态被 SEM 逻辑电路 105 侦测到,微处理器 101 自动地转换为降级模式 203。在降级模式 203 中,允许执行 BIOS 指令,以提供使用者输入与信息的显示的功能,但是更多复杂的软件 (例如操作系统) 的执行则不被允许。在降级模式 203 中,在微处理器 101 的安全执行模式 202 的安全编码操作被关闭,但是仍允许执行 BIOS 指令。在一实施例中,BIOS 指令通过发出一外部中断与传递状态给该微处理器且经由一机械专用寄存器来执行。在 x86 兼容的实施例中,在此降级模式 203 中实施 SMI 中断以执行 BIOS 指令。

[0124] 这些导致微处理器由安全执行模式 202 转换为降级模式 203 的已定义状态可以是执行安全编码的结果、或是多硬件侦测状态、或是安全编码执行结果与硬件侦测状态的结合。此硬件侦测状态包括与潜在安全暴露或篡改相关联的监控状态。在一实施例中,根据这些已定义状态的一侦测结果,SEM 逻辑电路 105 试图清除微处理器内一安全易失性存储器的一数据区域,且试图将侦测结果纪录至安全非易失性存储器 107。根据该数据区域的成功清除与该侦测结果的成功纪录,SEM 逻辑电路 105 将微处理器转换至降级模式 203。此外,执行在降级模式 203 的安全编码,亦即在一安全应用程序的控制下 (简称为程控),微处理器的状态转换回安全执行模式 202。

[0125] 某些与配置和完整性确认有关的已定义状态可导致微处理器 101 转换为硬件关机模式 204。在一实施例中,根据这些已定义状态的一侦测结果,SEM 逻辑电路 105 试图清除微处理器内一安全易失性存储器的一数据区域、试图将该侦测结果纪录至安全非易失性

存储器 107、且使微处理器进入至硬件关机模式 204。在此硬件关机模式下,只可藉由重置微处理器来退出此硬件关机模式。在安全执行模式 202 或降级模式 203 中一安全应用程序的控制下(简称为程控),微处理器 202 可进入硬件关机模式 204。

[0126] 现在参阅图 3,其表示在本发明实施例的微处理器 300 中的 SEM 逻辑电路 301 的详细方块图。SEM 逻辑电路 301 包括授权的公开密钥寄存器 318、处理器密钥寄存器 312、SEM 初始化逻辑电路 305、SEM 监控逻辑电路 306、SEM 中断逻辑电路 307、SEM 例外(exception)逻辑电路 308、SEM 定时器 309、SEM 实时时钟(RTC)310、非易失性使能指示寄存器 328、SEM 机械专用寄存器存储体(bank)329 以及安全易失性存储器 302。SEM 逻辑电路 301 耦接在微处理器 300 中的一些其它资源,包括通过总线 326 耦接非安全存储器 325、通过总线 324 耦接地址逻辑电路 323、通过总线 320 耦接随机数产生器 319、通过总线 321 耦接 AES/HASH/RSA 单元 311、通过总线 327 耦接其它处理器执行单元 313(例如整数单元、浮点单元、MMX/SSE 单元)、耦接正常例外逻辑电路 314、耦接正常追踪/除错逻辑电路 315、耦接正常中断逻辑电路 316 以及电源管理逻辑电路 322。

[0127] 在一实施例中,由授权者提供公开密钥,且微处理器 300 的制造期间中,公开密钥永久地编程在授权的公开密钥寄存器 318。在一实施例中,此公开密钥为 1024 位的 RSA 密钥,且授权的公开密钥寄存器 318 包括 1024 位的熔丝库(fuse bank)。因此,此公开密钥可在微处理器 300 的制造期间被编程,而不是在制造之后。或者,公开密钥藉由离线(off-line)大规模的初始化而被编程至安全非易失性存储器 107,其中,此离线大规模的初始化是用来编程一些安全非易失性存储器 107。使能与初始化安全执行模式 202 的能力是非常关键的安全操作,且木马程序(Trojan Horse)有可能被安装(installation)进安全非易失性存储器 107。因此,利用提供公开密钥的方法以避免窥察与篡改来控制安全执行模式初始化程序。

[0128] 处理器密钥寄存器 312 是多熔丝的聚集体,其实际分布在微处理器管芯上。这些熔丝在制造期间以独特且随机产生的状态组来编程以形成处理器的独特密钥,其只可被 AES/HASH/RSA 单元 311(也可称加密单元 311)来读取,并无提供自处理器密钥寄存器 312 读取处理器密钥的程序接口。在一实施例中,处理器密钥寄存器 312 包括 128 个熔丝,这些熔丝被编程为 128 位的 AES(Advanced Encryption Standard, AES)密钥,而使用此 AES 密钥来对安全非易失性存储器 107 的内容进行加密与解密。即是,使用此处理器对称密钥来对安全编码进行加密,以储存在安全非易失性存储器中。依据通过私密总线 106 来对安全编码的撷取,来自处理器密钥寄存器 312 的密钥被使用来对安全编码进行解密以进一步执行。因此,私密总线 106 的状态的观察者无法决定何者正在微处理器 300 与非易失性存储器 107 之间转移。

[0129] 在一实施例中,处理器密钥寄存器 312 包括 128 熔丝,其随机地分布在微处理器 300 中一熔丝库内的许多其它熔丝之中。此熔丝库配置在微处理器管芯上一些金属层的下方。

[0130] 根据 SEMENABLE 指令的执行或其它进入安全执行模式 202 的预期机制,SEM 初始化逻辑电路 305 提供安全执行模式 202 的初始化。为了详细说明,下文将以用来使能且执行来自安全执行模式 202 的指令(例如 SEMENABLE)执行的方式来说明根据本发明的微处理器 300 的操作,然而,此技术领域的人士将理解有其它方法能使安全执行模式 202 并执

行来自安全执行模式的安全编码,例如对一隐密寄存器 (hidden register) 写入等等。根据 SEMENABLE 指令的执行成功,SEM 初始化逻辑电路 305 将微处理器 300 的状态记录在非易失性使能指示寄存器 328。由安全执行模式 202 转换至原生未受控模式 201 时,SEM 初始化逻辑电路 305 将微处理器 300 的状态(安全执行模式被使能的状态)记录在非易失性使能指示寄存器 328。亦即,非易失性使能指示寄存器 328 用以指示微处理器 300 是否处于安全执行模式或一非安全执行模式。在微处理器的电源移除与重新施加的期间,非易失性使能指示寄存器 328 的内容持续存在。在一实施例中,非易失性使能指示寄存器 328 包括配置在微处理器 300 内的多熔丝,且微处理器 300 可由安全执行模式 202 转换至原生未受控模式 201 的次数对应在这些熔丝中的一特定熔丝数量。微处理器 300 包括配置在一单一管芯上的一单一集成电路。在一实施例中,SEM 逻辑电路根据进入至该安全执行模式而对非易失性使能指示寄存器 328 进行第一次写入,以指示出微处理器处于安全执行模式。SEM 逻辑电路根据退出该安全执行模式而对非易失性使能指示寄存器 328 进行第二次写入,以指示出微处理器处于该非安全执行模式(原生未受控模式)。

[0131] SEM 监控逻辑电路 306 用来监控安全编码与数据的诚实性,以监控系统的环境与物理属性,包括温度、电压、总线时钟、电池 VP 的存在、石英器 X1 的存在以及安全非易失性存储器 107 的存在。SEM 监控逻辑电路 306 将篡改或疑似的篡改情况指示给 SEM 逻辑电路 301,其导致微处理器 300 转换至降级模式 203 或硬件关机模式 204。

[0132] SEM 中断逻辑电路 307 提供多中断与相关的中断逻辑装置(例如安全中断描述符表 (Interrupt Descriptor Table, IDT)),这些只显现给正在安全执行模式 202 下执行的安全应用程序,且由此安全应用程序来存取。中断安全编码执行的机制类似于执行正常模式的机制。亦即,依据 SEM 中断的设置 (assertion),且藉由 SEM IDT 的出现使得安全编码状态被保存并转移至安全中断管理者 (secure interrupt handler)。由中断指令的恢复 (return) 执行将控制权恢复至安全编码中的断点。当微处理器 300 正操作在安全执行模式时,SEM 中断逻辑电路 307 提供安全中断以中断安全应用程序。SEM 中断逻辑电路 307 不被系统总线资源或非安全应用程序所得知或存取。当微处理器 300 正操作在非安全执行模式时,微处理器 300 的正常中断逻辑电路 316 提供非安全中断以中断非安全应用程序。

[0133] 同样地,SEM 例外逻辑电路 308 提供多安全例外与相关的例外管理逻辑装置。当该微处理器正操作在安全执行模式 202 时,SEM 例外逻辑电路 308 提供多安全例外并禁能多非安全例外。SEM 例外逻辑电路 308 无法被该等系统总线资源或该等非安全应用程序所得知或存取,其只显现给正在安全执行模式 202 下执行的安全应用程序,且由此安全应用程序来存取。所有安全编码程序例外与中断利用预设的 IDT,此预设 IDT 存在于 SEM 中断逻辑电路 307 内,以在中断与例外期间内控制分支。在一实施例中,根据该等安全例外的一者的使能,微处理器的状态被储存且程控转移至一对应安全例外管理者,其中微处理器的状态无法被该等非安全应用程序所存取。在安全应用程序执行之前,SEM 逻辑电路 301 禁能正常例外逻辑电路 314,以及当微处理器 300 正操作在非安全执行模式时,正常例外逻辑电路 314 提供对应该等非安全应用程序的多非安全例外。在一实施例中,假使在该等非安全应用程序的任一者执行的期间发生该等安全中断的任一者或该等安全例外的任一者,微处理器的状态被储存且微处理器 300 进入安全执行模式。

[0134] 这些安全中断被配置来提供微处理器 300 外部事件所导致的程控转移,例如键盘

事件、I/O 端口事件等等。安全例外是用来提供微处理器 300 内部事件所导致的程控转移，例如非定义的运算码 (opcode)、机械检查错误 (machine check errors)、以及在一实施例中对一个或多个安全机械专用寄存器存储体 329 的安全编码写入。IDT 包括多安全寄存器，其被加载多指标，而这些指标是指向在安全编码中的安全中断管理者与安全例外管理者 (secureexception handler)。IDT 提供转移至该安全应用程序内的多安全中断管理者与多安全例外管理者) 的程控。此预设 IDT 包括关于程控转移至该微处理器将执行的一安全执行模式重置操作的数据。在一实施例中，根据该等安全中断的一者的使能，该微处理器的状态被储存且程控转移至一对应安全中断管理者，以及该微处理器的状态无法由该等非安全应用程序来存取。在一实施例中，根据该等非安全中断的一者的使能，该微处理器的状态被储存且程控转移至一对应非安全中断管理者，以及该微处理器的状态无法由该等非安全应用程序来存取。

[0135] SEM 定时器 309 是只显现给正行在安全执行模式 202 下执行的安全应用程序且由此安全应用程序来存取的多定时器。SEM 定时器 309 包括多中断，而这些中断可由操作在安全执行模式 202 下的安全编码来存取。SEM 实时时钟 310 其提供持续时间 (persistent time)，其只显现给正在安全执行模式 202 下执行的安全应用程序且由此安全应用程序来存取。SEM 实时时钟 310 的值无法由不同于操作在安全执行模式 202 下的安全编码的任何对象来改变。SEM 机械专用寄存器存储体 329 包括多机械专用寄存器，且这些机械专用寄存器只显现给正在安全执行模式 202 下执行的安全应用程序且由此安全应用程序来存取。这些机械专用寄存器用来使能对安全非易失性存储器 107、SEM 实时时钟 310 以及 SEM 定时器 309 的加载 / 储存存取。

[0136] 非安全存储器 325 作为给正在执行的非安全应用程序的指令与数据高速缓存 (instruction and data cache)。非安全存储器 325 用以储存多非安全应用程序以由微处理器来执行。在微处理器 300 内的这些程序与其它系统总线资源可得知且存取非安全存储器 325。安全易失性存储器 302 系作为给正在安全执行模式 202 下执行的安全应用程序的一指令与数据高速缓存。进入至安全执行模式 202，安全易失性存储器 302 的一堆栈 (stack) 提供来储存处理器状态 303，其用于对应该等非安全应用程序的该微处理器的状态的储存与取回。安全易失性存储器 302 的其它堆栈提供来储存安全编码 304 与对应安全编码的安全数据 317。安全易失性存储器 302 根据微处理器的重置而被清除，且其完全地隔离于系统总线，因此，安全易失性存储器 302 无法被非安全系统资源窥察、加载、除错或其它方法的存取。安全编码 (安全应用程序) 可使用正常处理器加载与储存指令来存取安全易失性存储器 302，以加载 / 储存安全数据 317，其中，这些正常处理器加载与储存指令是参考地址逻辑电路 323 内的正常片段寄存器 (normal segment register)，此正常片段寄存器是当于安全易失性存储器 302 (而不是正常系统存储器) 进入至安全执行时而被初始化。此正常系统存储器也被执行在安全执行模式的安全编码，通过地址逻辑电路 323 且使用正常加载与储存指令来存取。然而，根据安全编码的执行，SEM 逻辑电路 301 通过总线 324 来命令地址逻辑电路 323 以停止虚拟地址转译。亦即，因为虚拟 - 实体地址转译为了指令与数据而被禁能，因此，通过总线 324 且由安全编码所提供的地址必须为实体地址。藉由这种作法，SEM 逻辑电路阻止了分页错误，藉以消除此篡改来源。

[0137] 在一实施例中，安全易失性存储器 302 完全地属于在微处理器 300 内的芯片上

(on-chip) 高速缓存,但安全易失性存储器 302 快取线具有将这些快取线完全地隔离于微处理器总线的特定内部属性。这些快取线没有耦接至外部系统存储器,因此这些快取线无法自系统存储器装载或存入至系统存储器,这些快取线也无法被任何汇流窥探资源来外部地或内部地窥察。

[0138] 在一实施例中,安全易失性存储器 302 包括 4K 64 位快取线。在安全易失性存储器 302 中,一快取线依据由将数据移动至先前没有涉及 (referenced) 的一快取线来分配。在一实施例中,安全易失性存储器 302 包括具有 4096 个位置的一 64 位高速缓存,该等位置的每一者包括一内部属性,且该内部属性完全地隔离该等位置的每一者。

[0139] 在另一实施例中,安全易失性存储器 302 包括随机存取存储器,其与微处理器 300 内的芯片上高速缓存分离。

[0140] SEMENTER 指令的执行提供了安全执行模式 202 内安全编码的执行。在一 x86 相容的实施例中,安全执行模式 202 根据修改的 32 位 x86 真实模式来提供安全编码的执行。在执行安全编码时,禁止由安全执行模式 202 进入一 x86 保护模式。在安全执行模式执行之前,SEM 初始化逻辑电路 305 藉由设置一使能信号 DISIL 来禁能正常 (即非安全) 中断逻辑电路 316。在安全执行模式执行之前,SEM 初始化逻辑电路 305 也藉由设置一使能信号 DISEL 来禁能正常 (即非安全) 例外逻辑电路 314,也藉由设置一使能信号 DISDL 来禁能正常 (即非安全) 追踪 / 除错逻辑电路 315。此外,在安全执行模式执行之前,电源管理逻辑电路 322 藉由信号 DISPML 的设置而被禁能。通过这些安全措施,不会发生正常总线中断,阻止了除错例外、避免总线追踪周期、且禁能除错输出入埠。此外,信号 DISIL 用来在安全编码的执行期间内禁能所有的剩余处理器资源 (例如 JTAG、探测模式、快取测试)。否则,电源管理逻辑电路 322 允许微处理器 300 进入降低功耗状态,例如在 x86 兼容实施例中的 P 状态与 C 状态。因此,信号 DISPML 用来在安全编码执行期间避免功耗状态的转换。

[0141] 通过总线 320、321 及 327,安全编码可存取处理器执行单元 (处理器 300 内的执行单元) 313、随机数产生器 319 与 AES/HASH/RSA 单元 311,以执行微处理器指令集的所有指令,其中,这些指令包括真实随机数的硬件产生且可由编程的巨集指令来使用的硬件实施功能,以执行 RSA 加密、解密以及识别核对;AES 加密与解密、以及 SHA-1/SHA-256 哈希产生 (Secure HashAlgorithm, SHA, 安全哈希算法)。这些硬件实施功能由 AES/HASH/RSA 单元 311 来执行。

[0142] 现在参阅图 4,图示 400 表示在本发明的微处理器内安全编码如何被储存、存取及初始化。图标 400 说明能进行安全执行模式 (SEM) 的微处理器 401,其通过系统总线 425 而耦接 BIOS 存储器 410 与系统存储器 420。根据本发明,微处理器 401 也通过私密总线 431 而耦接至安全非易失性存储器 430。微处理器 401 包括安全编码接口逻辑电路 402,其耦接至随机数产生器 412、处理器密钥寄存器 413、授权的公开密钥寄存器 404、AES/HASH/RSA 单元 405 (或称加密单元 405)、安全易失性存储器 406、SEM 监控逻辑电路 408 以及 SEM 初始化逻辑电路 409。安全编码接口逻辑电路 402 另外耦接总线接口单元 403 与安全非易失性存储器接口单元 407。

[0143] 图标 400 也表示储存在系统存储器 420 与 BIOS 存储器 410 的安全编码 411 及 421。在一实施例中,储存在 BIOS 存储器 410 的安全编码 411 主要是用来提供微处理器 401 在降级模式 203 中的操作,而储存在系统存储器 420 的安全编码 421 是用来提供微处理器 401

在安全执行模式 202 中的操作。

[0144] 在操作上,图标 400 所示的组件的运作,实质上相似于先前参阅第 1-3 图而已叙述的相似名称组件。参阅图 4 的讨论目的是为了更加明确集中注意在那些组件与技术,而那些组件与技术是用来储存、存取、初始化、执行在本发明的安全环境中的安全编码。

[0145] 此外,关于安全编码执行的环境是隔离于非安全编码执行的环境。如先前所述,原生未受控模式 201 只允许非安全编码的执行。安全执行模式则允许非安全编码与安全编码两者的执行。在安全编码 421 执行之前,微处理器 401 的状态被保存。根据回到非安全编码的执行的转换,此状态恢复 (restored)。此状态储存在安全易失性存储器 406 内的一个区域,且此状态不会出现在微处理器总线 425 上。此外,安全编码 411、421 是执行自安全易失性存储器 406。除了将安全易失性存储器 406 隔离于与微处理器总线 425 联系的硬件与软件,所有其它“从属通道 (side channels)”(例如除错例外与执行追踪特征) 被禁能,如关于图 1-3 的讨论。安全编码 411、421 只提供给 SEM 中断逻辑电路 307、SEM 例外逻辑电路 308、SEM 实时时钟 310、SEM 定时器 310 以及只可由安全编码 411、421 利用的其它处理器资源独占存取。

[0146] 此外,微处理器 401 提供 SEM 监控逻辑电路 408,其包括的异步监控与监视机制,其中,此异步监控与监视机制独立于安全编码 411、421 以及非安全编码的执行。SEM 监控逻辑电路 408 监控微处理器的环境(例如电压、温度、总线运作)与物理特性,也核对安全编码 411、421(安全应用程序)与相关数据的诚实性,将于下文详细说明。当侦测到安全暴露 (security exposure) 时,SEM 监控逻辑电路 408 可通过总线 CHK 将程控转移至安全编码 411、421 的安全编码错误管理装置 (secure-code error handler),或者,在侦测到严重的安全暴露情况下,SEM 监控逻辑电路 408 将通过总线 CHK 来使微处理器 401 进入降级模式 203。

[0147] 在一实施例中,安全编码接口逻辑电路 402 监控存在于安全编码 411、421 中的多指令,且通过总线 INS 将这些指令提供至 SEM 监控逻辑电路 408,以支持微处理器 401 的限定的指令集架构 (Instruction set Architecture, ISA) 操作。根据此实施例,当微处理器 401 正操作在安全执行模式时,本发明的微处理器 401 只被允许执行主机 ISA 中的某些指令。即是,限定的 ISA 操作使得 SEM 逻辑电路阻止多非安全指令的执行,而此非安全指令的执行是授权者欲阻止的,且该些非安全指令包括取自对应微处理器的一指令集架构的一个或多个运算码。举例来说,在 x86 相容的实施例中,超过 100 个微指令的产生与执行的指令或某类指令要求会被阻止。另一方面,当微处理器 401 正操作在安全执行模式时,一授权者可能期望阻止所有指令的执行,例如任务切换、呼寻闸 (call gates) 等等。藉由将安全编码 411、421 内每一指令提供给 SEM 监控逻辑电路 408,本发明的微处理器 401 使能限定的 ISA 操作。在一实施例中,在限定的 ISA 指令集中的指令(即提供在安全执行模式下执行的指令),由 SEM 监控逻辑电路 408 内指令阵列(未显示)的值来表示,将于下文详细说明。当遭遇到上述被阻止的指令时,SEM 监控逻辑电路 408 使微处理器 401 进入降级模式 203。

[0148] 在一实施例中,安全编码接口逻辑电路 402 将安全编码 411、421 中的指令提供给 SEM 监控逻辑电路 408,提供时将安全编码 411、421 加载至安全易失性存储器 406 以进行后续执行。

[0149] 使能与初始化安全执行模式 202 的能力是非常关键的安全操作,此外,其表示了

关于木马程序 (Trojan Horse) 安装有可能进入至包含安全编码 411、421 的存储器 410、420 的区域。通过非对称加密算法与一组对应的非对称加密密钥的使用, 本发明的微处理器 401 藉由控制安全执行模式初始化程序而有利地阻止此暴露。在一实施例中, 非对称密钥算法是 RSA 算法, 且对应密钥则是由授权者所产生的 1024 位 RSA 公开与私密密钥。在一实施例中, 此授权者或授权实体 (entity) 提供执行的安全编码 411、421。如前文关于图 3 的说明, 在微处理器 401 的制造期间, 两密钥中的一者储存在授权的公开密钥寄存器 318, 且用来根据非对称密钥算法来对数据解密, 其中, 此数据已由授权者的其它非对称密钥 (即私密密钥) 来加密。

[0150] 因此, 在一实施例中, 此操作系统执行 SEMENABLE 指令 (或相似机制)。此指令传送通过授权者的私密密钥来加密的一 SEM 使能参数。安全编码接口逻辑电路 402 接着通过授权的公开密钥寄存器 404 来存取公开密钥, 且利用 AES/HASH/RSA 单元 405 来对此 SEM 使能参数解密。根据核对 SEM 使能参数, SEM 初始化逻辑电路 409 初始化安全执行模式 202, 亦即使能安全执行模式 202 以执行安全应用程序。除此之外, SEM 初始化逻辑电路 409 指示微处理器 401 自 SEMENABLE 指令恢复 (return) 后, 微处理器 401 保持在非安全执行模式 201。在一实施例中, 无论是否接受进入安全执行模式 202 的授权 (以及有一对应错误状态时, 假使有的话) 都会提供一响应编码 (returncode)。

[0151] 相对于在微处理器 401 的制造期间将授权的公开密钥直接编程至授权的公开密钥寄存器 404, 在另一实施例中, 授权者将授权的公开密钥编程至安全非易失性存储器 430 的授权的公开密钥区域 432。因此, 当微处理器 401 开机 (power up) 时, 安全非易失性存储器接口单元 407 自此区域 432 侦测并撷取此公开密钥。安全编码接口逻辑电路 402 接着将此密钥以及之后指示此密钥已被烧录的参数, 烧录至授权的公开密钥寄存器 404。此供选择的实施例在安全非易失性存储器 430 的制造阶段上, 提供了更弹性地公开密钥配置。安全非易失性存储器接口单元 407 通过私密总线 431 将微处理器 401 耦接至安全非易失性存储器 430, 其中, 在私密总线 431 上用来存取安全非易失性存储器 430 的多私密总线数据传输被隐藏, 以避免被微处理器 401 内多系统总线资源以及耦接该系统总线的任何装置所得知察觉。

[0152] 安全非易失性存储器接口单元 407 是由安全编码接口逻辑电路 402 所管理。根据核对一 SEM 使能参数, 安全非易失性存储器接口单元 407 藉由执行随机数写入来清除安全非易失性存储器 430 的内容。在一实施例中, 在安全非易失性存储器 430 中的每一个位置以随机数写入 64 次。在一实施例中, 每次写入的随机数是由随机数产生器 412 所产生。

[0153] SEMENABLE 指令 (或是 SEM 使能机制) 也传送关于安全编码 411、421 在 BIOS 存储器 410 或系统存储器 420 的位置的指针和任何初始安全数据 (亦即使能参数)。此指针与数据 (亦即使能参数) 是根据一预设结构来被格式化, 且根据非对称密钥算法而被加密。被加密的指针与数据被解密, 且格式化被核对。不成功的核对导致错误码的回应。

[0154] 假使在结构方面此指针与数据被确认且证实, 安全编码接口逻辑电路 402 则指示总线接口单元 403 去自 BIOS 存储器 410 以及 / 或系统存储器 420 撷取安全编码 411 及 421。安全编码 411、421 也已藉由使用授权者的私密密钥并根据非对称密钥算法而被加密, 且必须与预设结构相称。安全编码接口逻辑电路 402 利用授权的公开密钥寄存器 404 与 AES/HASH/RSA 单元 405 来对加密的安全编码 411、421 进行解密。在核对为正确格式后, 安全编

码接口逻辑单元 402 利用 AES/HASH/RSA 单元 405 来根据对称加密算法并使用处理器密钥寄存器 413 的内容（作为对称密钥）来对安全编码与数据进行加密。如前所提及，处理器密钥寄存器 413 的内容是微处理器 401 所特有的 128 位随机产生的密钥，且对称加密算法包括使用 128 位模块 (blocks) 以及电子密码书 (Electronic Code Book, ECB) 模式的高级加密标准 (AES)。此对称加密的安全编码接着通过安全非易失性存储器接口单元 407 而被写入至安全非易失性存储器 430。此外，安全编码接口逻辑电路 402 利用 AES/HASH/RSA 单元 405 与处理器密钥寄存器 413 来产生安全编码中已选择部分的多个哈希，安全编码接口逻辑电路 402 对这些哈希进行加密编码并写入至安全非易失性存储器 430。在一实施例中，这些哈希是根据 SHA-1 算法而产生。

[0155] 此外，SEM 初始化逻辑电路 409 禁能 JTAG、探测模式、快取测试、或者禁能通过图 3 所讨论的机制而提供安全编码监视的其它处理器特性。

[0156] 当被编码且被哈希的安全编码已写入至安全非易失性存储器 430，微处理器 401 设定非易失性使能指示寄存器（如图 3 中 328 所示）指示出处理器 401 正操作于安全执行模式 202 且 SEM 初始化逻辑电路 409 迫使微处理器 401 执行一重置序列 (RESET sequence)。

[0157] 部分的重置序列导致非易失性使能指示寄存器的内容被读取，且假使这些内容指示出处理器 401 处于安全执行模式 202 中，则执行安全执行模式 202 所特有的额外操作。

[0158] 因此，安全编码 411、421 起初被加密，且由授权者加载至存储器 410、420。当安全执行模式被使能时，微处理器 401 根据非对称密钥算法并使用授权者所提供的密钥来撷取且核对安全编码。接着使用处理器独特密钥并根据对称密钥算法来加密且哈希此编码，且对称加密的编码通过私密总线 431 而被写入至安全非易失性存储器 430。

[0159] 以下将进一步详细说明，当安全编码将被执行时，安全编码由安全非易失性存储器接口单元 407 自安全非易失性存储器 430 被撷取，且使用存放于处理器密钥寄存器 413 的处理器密钥来译码，且安全编码被写入至微处理器 401 内的安全易失性存储器 406，其中，安全易失性存储器 406 完全隔离开所有可窥探其内容的硬件及或软件。安全易失性存储器 406 的功能包含可存放安全应用程序执行的指令与数据高速缓存。

[0160] 在一实施例中，安全非易失性存储器接口单元 407 包括多机械专用寄存器，其专有地显现给安全编码，这些机械专用寄存器允许一安全应用程序（或安全编码接口逻辑电路 402）去执行对安全非易失性存储器 430 的加载与储存。即是，根据此实施例，藉由执行对隐藏机械专用寄存器的读取与写入，来执行对安全非易失性存储器 403 的读取与写入。

[0161] 授权者可有利地将微处理器 401 的安全操作与安全执行模式环境结合，且由于通过系统总线 425 与私密总线 431 的数据传输被加密，因此安全编码的结构与功能则被保护以避免任何的反向工程与其它窥察 / 侵入技术。

[0162] 现在参阅图 5，其表示在图 1 的微处理器中的 SEM 监控逻辑电路 500 的详细内容。SEM 监控逻辑电路 500 包括物理环境监控器 501，其通过信号 PSNT 耦接安全非易失性存储器 107、通过信号 VP1 与 VP2 耦接电池 VP，且通过信号 C1 与 C2 耦接石英器。此物理环境监控器 501 通过总线 NOBOOT 提供一输出信号。

[0163] SEM 监控逻辑电路 500 也包括总线时钟监控器 502，其具有频率参考单元 503。总线时钟监控器 502 通过信号 BUS CLK 耦接提供至微处理器的总线时钟，且总线时钟监控器 502 的输出系耦接总线 TAMPER。

[0164] SEM 监控逻辑电路 500 也包括处理器电压监控器 504, 其通过信号 VDD 与 BUSTERM 耦接电源供应电压与多总线终端电压, 其中, 电源供应电压与总线终端电压由系统板提供至微处理器。SEM 监控逻辑电路 500 也包括温度监控器 505, 其通过信号 TEMP 耦接至处理器温度感测逻辑电路 (未显示)。SEM 监控逻辑电路 500 更包括数据监控器 506, 其通过总线 CHK 耦接至安全编码接口逻辑电路 402。总线时钟监控器 502、处理器电压监控器 504、温度监控器 505 以及数据监控器 506 的输出信号则耦接至总线 TAMPER。

[0165] SEM 监控逻辑电路 500 更包括安全时戳计数器 (security time stampcounter) 507, 其耦接正常时戳计数器 (normal time stamp counter) 508、信号 CORE CLK 以及比率 (Ratio) 机械专用寄存器 509。安全时戳计数器 507 的输出信号耦接总线 TAMPER。

[0166] SEM 监控逻辑电路 500 也包括指令监控器 511, 其耦接指令阵列 512 与总线 INS。如关于图 4 的讨论, 当微处理器正执行在安全执行模式时, 在安全应用程序内的指令被提供至 SEM 监控逻辑电路 500, 以支持在主机 ISA 内限制的指令执行。指令监控器 511 的输出信号耦接至总线 TAMPER。

[0167] 最后, SEM 监控逻辑电路 500 具有样式监控器 510, 其耦接总线 PINCHK, 且在总线 DESTRUCT 上产生一输出信号。

[0168] 总线 NOBOOT、TAMPER 以及 DESTRUCT 耦接于监控管理器 513。在一实施例中, 监控管理器 513 产生信号 CLASS1、CLASS2、CLASS3 以及 DISABLE。

[0169] 在操作上, SEM 监控逻辑电路 500 用来执行硬件与软件检验, 其监控本发明微处理器的物理与暂时的属性, 以侦测、识别以及分类操作事件 (operating events), 其中, 操作事件是表示对于安全编码而言不安全的操作环境, 例如改变或移除电池、石英器或者安全非易失性存储器; 以本发明的不安全的微处理器来取代本发明的安全微处理器; 修改总线时钟; 篡改微处理器电源供应电压 VDD; 修改在系统存储器、BIOS 存储器或安全非易失性存储器内的加密安全编码; 以及发生对安全编码本身的过度呼寻 (excessivecalls)。

[0170] 因此, 当操作在安全执行模式时, 物理环境监控器 501 耦接安全非易失性存储器 107, 藉由监控信号 PSNT 的状态来判断安全非易失性存储器 107 是否移除。信号 PSNT 的禁能 (de-assertion) 表示移除安全非易失性存储器 107。同样地, 监控信号 VP1 与 VP2 来判断电池电压是否改变或电池被移除或者判断对应该电池的电压是否被充电。在一实施例中, VP1 的值与电池电压成比例。同样地, 信号 C1 与 C2 的状态表示石英器的存在与否。假使物理环境监控器 501 侦测到上述的任何变化, 此变化则输出至总线 NOBOOT。

[0171] 此外, 当操作在安全执行模式 202 时, 总线时钟监控器 502 估计信号 BUS CLK 的频率, 以判断系统总线时钟的短期与长期完整性, 其中, 系统总线时钟通过系统板而提供至微处理器。此总线时钟通过信号 BUS CLK 被路由 (routed) 至总线时钟监控器 502, 总线时钟监控器 502 使用内部相位锁相回路 (未显示) 来检验短期总线时钟误差, 其中, 内部相位锁相回路与总线时钟同步化且用来产生内部时钟给微处理器。总线时钟监控器 502 判断总线时钟于不适当的周期是否维持平坦, 或者判断时钟变化是否已超出可接受的程度 (例如一特定范围)。在一实施例中, 超过百分的六的变化视为是无法接受的。此外, 总线时钟监控器 502 使用频率参考单元 503 来作为温度与电压非相依的中间速度振荡器电路。频率参考单元 503 产生与系统总线时钟成比例的一参考频率。总线时钟监控器 502 比较系统总线时

钟的衍生 (derivative) 与频率参考单元 503 的输出 (参考频率), 以判断总线时钟的频率是否已经历逐步 (gradual) 的频率变化。假使任何上述事件发生, 此事件通过总线 TAMPER 报导给监控管理器 513 (SEM 逻辑电路 301), 其将导致微处理器进入降级模式或进入硬件关机模式 204。

[0172] 处理器电压监控器 504 估计通过信号 VDD 与 BUSTERM 来提供且施加于微处理器的电源供应电压与多总线终端电压。上述电压的高低限制通过机械专用寄存器 (未显示) 来编程。一旦电源供应电压与多总线终端电压偏离这些编程限制, 处理器电压监控器 504 将通过总线 TAMPER 来报导 (report) 此事件给监控管理器 513。

[0173] 温度监控器 505 包括精准的热监控机制 (除了正常热监控功能以外), 其在预设高与低温度限制下不断地监控管芯温度。该管芯温度的一低温度限制与一高温度限制藉由温度监控器 505 内一机械专用寄存器来编程。此高与低温度限制储存在温度监控器 505 内机械专用寄存器中, 其中, 这些机械专用寄存器可被安全编码写入。一旦该管芯温度偏离上述预设高与低温度限制, 温度监控器 505 将通过总线 TAMPER 来报导此事件给监控管理器 513。

[0174] 数据监控器 506 用来当自安全非易失性存储器撷取该安全应用程序时, 用以侦测与报导与安全编码和安全数据相关的多加密与配置错误。这些多加密与配置错误通过总线 TAMPER 来报导给监控管理器 513。举例来说, 这些错误为与 SEMENABLE 及 SEMENTER 指令的执行相关的错误、当自存储器撷取安全编码时所侦测到的解密错误、以及在安全编码中哈希与格式错误。

[0175] 安全时戳计数器 507 耦接一核心时钟信号 CORE CLK, 用来计算当安全编码正执行时的核心时钟信号 CORE CLK 的周期数。安全时戳计数器 507 耦接一正常时戳计数器 508。正常时戳计数器 508 则是在非安全编码或安全编码执行期间内计算信号 CORE CLK 的周期数。当安全应用程序正在执行时或当安全应用程序非正在执行时, 正常时戳计数器 508 计算信号 CORECLK 的周期数。安全时戳计数器 507 也耦接一比率机械专用寄存器 509, 比率机械专用寄存器 509 只由该安全应用程序所得知且存取。安全执行模式执行期间, 安全编码可对比率机械专用寄存器 509 执行一机械专用寄存器写入, 以建立介于正常时戳计数器 508 与安全时戳计数器 507 的数值之间的一最大比例 (maximum ratio)。此最大比例指示该安全应用程序已被呼寻的次数。假使超过此最大比例, 藉此指示出安全编码已被呼寻多于指定次数, 接着, 安全时戳计数器 507 通过总线 TAMPER 报导此事件 (最大比例何时被超过) 给监控管理器 513。亦即, 安全时戳计数器 507 用以比较信号 CORE CLK 周期数与正常时戳计数器 508 的数值、且将上述最大比例被超过的事件报导给监控管理器 513。上述最大比例藉由 SEM 逻辑电路内的一机械专用寄存器来编程。

[0176] 指令监控器 511 在与主机 ISA 内指令子集的对照下用来确认在安全应用程序内的指令, 且指示出在此安全应用程序内且非在此子集内的指令何时已被编程以进行后续执行。提供来在安全执行模式内执行的指令子集是由指令阵列 512 的数值来表示。在一实施例中, 此子集包括在 ISA 内的一个或多个特殊指令, 如运算码 (opcode) 所识别。在一实施例中, 此子集包括一个或多个指令种类, 如一微码 (microcode) 复杂数值所识别。在一第三实施例中, 此子集包括一个或多个卷标编码 (tag codes), 每一者与一个或多个指令运算码相关联。

[0177] 指令阵列 512 耦接该指令监控器 511, 用以识别对应微处理器的一指令集架构内

的一所有指令的子集,该子集包括允许在一安全执行模式内执行的指令。用来在安全执行模式下执行的指令子集由指令阵列 512 的数值来识别。在一实施例中,此指令阵列 512 包括一机械专用暂存器,其初始地由安全应用程序来写入。在另一实施例中,指令阵列 512 包括多熔丝,其在制造期间被编程(烧断)。

[0178] 在安全执行模式的初始化期间,当安全编码正由安全非易失性存储器传送至安全易失性存储器以进行后续执行时,对应安全编码内每一特定指令的数值由安全编码接口逻辑电路 402 通过总线 INS 而提供至指令监控器 511。在一实施例中 INS 的数值表示每一特定指令对应微处理器的一指令集架构内的特定运算码或是运算码子集。在另一实施例中,此数值表示这些指令的种类(例如简单、复杂等等)。在又一实施例中,此数值是对应在 ISA 内一或多个指令的卷标。

[0179] 在另一实施例中,于安全编码的执行之前,当安全非易失性存储器正被编程时,在安全编码内每一指令的数值由安全编码接口逻辑电路 402 通过总线 INS 来提供。

[0180] 指令监控器 511 比较 INS 的数值与指令阵列 512 的数值,以判断是否允许执行特定指令。假使不允许的话,指令监控器 511 则设置信号于总线 TAMPER。

[0181] 样式监控器 510,耦接总线 DESTRUCT,是侦测本发明的微处理器的非安全版本对系统板的安装,其中,此系统板是配置给本发明的安全微处理器。在一实施例中,非安全微处理器与安全微处理器具有相异的接脚配置(pinout)。在此两版本之间相异的特定脚位的状态系通过总线 PINCHK 作为样式监控器 510 的输入信号。样式监控器估计总线 PINCHK 的状态,且假使判断出此非安全版本被安装时,则通过总线 DESTRUCT 来报导此事件给监控管理器 513。亦即,总线 DESTRUCT 提供对应微处理器的特定多接脚配置的多状态,且样式监控器 510 则估计上述多状态以判断微处理器是否配置一安全版本来操作在该安全执行模式中。

[0182] 监控管理器 513 藉由注意与估计通过总线 NOBOOT、TAMPER 及 DESTRUCT 传递的数据,来动态地监控微处理器的物理与操作环境。监控管理器 513 对上述数据进行分类以指示出与安全应用程序的执行相关的安全层级,且使微处理器内的 SEM 逻辑电路根据安全层级来执行反应操作。对安全应用程序的执行而言,SEM 逻辑电路 500 包括异步监控、监视机制与监控器等系独立地操作。以下某些情况将导致信号 CLASS1 的设置,例如通过总线 TAMPER 报导的总线 BUS CLK 的频率的短暂误差。SEM 逻辑电路响应于 CLASS1 的设置而将此事件纪录(log)(侦测信号 CLASS1 的设置)至安全易失性存储器内的安全事件纪录表,且发出一中断给安全编码。假使此中断没有被收到(acknowledged),则监控管理器 513 设置信号 CLASS3。

[0183] 假使侦测到会导致信号 CLASS1 设置的多事件(多于一个事件),例如 BUS CLK 的误差与 VDD 的误差,监控管理器 513 则设置信号 CLASS2。SEM 逻辑电路则试图清除安全易失性存储器的数据区域,且试图将此事件记录至安全非易失性存储器。此外,检查在 BIOS 的安全编码的哈希。假使安全易失性存储器的数据区域成功清除且此事件(侦测信号 CLASS2 的设置)被纪录,且假使 BIOS 哈希被正确地证明,SEM 逻辑电路则开始转换至降级模式 203。此降级模式提供有限的功能、错误显示以及有限的使用者输入的相关指令。这些动作中任一者的错误会导致信号 CLASS3 的设置。

[0184] 信号 CLASS3 的设置表示有安全侵害。响应于信号 CLASS3 的设置,SEM 逻辑电路

持续试图清除安全易失性存储器且试图将此事件（侦测信号 CLASS3 的设置）记录至安全非易失性存储器，此外，使微处理器进入硬件关机模式 204，即微处理器停止操作。

[0185] 在一实施例中，监控管理器 513 判断样式监控器 510 是否已设置信号 DESTRUCT，因此指示出本发明微处理器的非安全版本的安装。假使信号 DESTRUCT 被设置，且假使在总线 NOBOOT 上的数据指示出石英器与安全非易失性存储器存在时，信号 DISABLE 则被设置。响应于信号 DISABLE 的设置，SEM 逻辑电路使非安全的微处理器停止操作。

[0186] 以上关于监控管理器 513 设置信号 CLASS1、CLASS2、CLASS3 以及 DISABLE 皆用来将程控转移至安全应用程序内多事件管理者之一，例如有安全侵害时，信号 CLASS3 被设置，SEM 逻辑电路则持续尝试清除安全易失性存储器且将此事件记录至安全非易失性存储器，持续尝试迫使微处理器进入硬件关机模式，即微处理器停止操作。关于监控管理器 513 设置信号 CLASS1、CLASS2、CLASS3 以及 DISABLE 的上述情况仅为范例，是用来教导本发明的安全环境管理。此技术领域中具有通常知识者能理解，安全事件类别以及适当反应是受到所需的特定安全环境所约束，因此，本发明包含了上述安全事件类别与适当反应的其它方法。

[0187] 现在参阅图 6，状态图 600 详细说明本发明的微处理器的操作模式转换。状态图 600 包括原生未受控模式 601（或“非安全”执行模式 601）、降级模式 605 以及硬件关机模式 606，如同图 2 中相似命名的组件，相异之处在于，更详细说明原生未受控模式 601 在程控下只可返回至此模式的有限次数。这些返回的有限次数以原生未受控模式 (born free mode, BFM) [1:N] 来表示。此外，更详细地解释在图 2 的安全执行模式 202，以说明多 SEM 使能重置模式 [1:N] 602、一 SEM 使能正常执行模式 603 以及一 SEM 使能安全执行模式 604。即是，当安全执行模式 202 通过 SEMENABLE 指令的执行（或者其它使能机制）而被使能时，本发明的微处理器被重置（即使能重置 [1:N]）其可能正在执行非安全应用程序（使能正常执行模式），或者可能正执行安全编码（使能安全执行模式）。

[0188] 如上所示，本发明的微处理器被制造为初始开机即进入原生未受控模式 601。且如状态图 600 所指示，有关微处理器的安全的不同版本可持续地被使用于原生未受控模式中。然而，SEMENABLE 指令或使能安全执行模式的交替机制（例如 SEM ENABLE）的执行导致微处理器进入 SEM 使能重置模式 602，以迫使微处理器重置，其中可以进入 SEM 使能重置模式 602 的次数为 “:N” 次，且上述为第一次进入 SEM 使能重置模式 602。在 SEM 使能重置模式 602 中，在重置序列期间，微处理器执行关于操作在安全环境的配置与诚实性检查，如前述关于图 5 的叙述。根据在 SEM 使能重置模式下重置的成功执行（即通过），微处理器转换至 SEM 使能正常执行模式 603，以进行非安全应用程序的执行。然而，假使侦测到某些已定义状态，例如前述由监控管理器 513 对信号 CLASS3 与 DISABLE 的设置，微处理器将转换至降级模式 605（即由于 CLASS2 的设置），或转换至硬件关机模式 606（即由于 DISABLE 的设置）。从硬件关机模式 606 离开，微处理器可被重置以导致其返回至 SEM 使能重置模式 602 中。从降级模式 605 离开，微处理器通过 BIOS 提供受限的指令，允许使用者建立用来在程控下使能微处理器以进入 SEM 使能安全执行模式 604 的参数。

[0189] 从 SEM 使能重置模式 602 离开，在重置序列中的硬件呼寻将迫使微处理器直接进入 SEM 使能安全执行模式 604，于其中执行安全编码。此外，发生在 SEM 使能正常执行模式 603 中非安全编码执行期间中或者在 SEMENTER 指令的执行期间中的安全中断、或者使微处

理器开始执行安全编码的交替机制,将导致微处理器转换至 SEM 使能安全执行模式 604。命令微处理器开始执行安全编码的指令与交替机制都参照状态图 600 中的“呼寻”。同样地,SEMENTXIT 指令的执行或命令微处理器终止安全编码执行与开始非安全编码执行的交替机制,参照“返回 (RETURN)”,此返回导致微处理器转换为 SEM 使能正常执行模式 603。如上所述,安全编码可导致微处理器由 SEM 使能安全执行模式 604 转换为降级模式 605。BIOS 内的安全编码允许微处理器由降级模式 605 返回至 SEM 使能安全执行模式 604。

[0190] 最后,在 SEM 使能安全执行模式 604 中执行的安全编码可藉由写入一特殊机械专用寄存器,来引发安全机械检查例外,其导致微处理器转换回 SEM 使能正常执行模式 603 以执行非安全编码。此外,假使在 SEM 使能正常执行模式 603 中发生一安全中断,微处理器的状态自动地改变至 SEM 使能安全执行模式 604。这些执行在本发明微处理器范例中用来导致状态图所述的状态变化的不同的步骤,将通过第 7-11 图来详细说明。

[0191] 参阅图 7,流程图 700 表示本发明微处理器中使能安全执行模式操作的高阶方法。流程图开始于方块 701,于其中,微处理器处于原生未受控模式 601。通过 SEMENABLE 指令的执行或使能安全执行模式的交替机制,例如写入至一隐藏机械专用寄存器,传送一使能参数,其中,此使能参数已藉由使用一对非对称加密密钥中的一者并根据非对称加密算法来被加密,而一对非对称加密密钥中的另一者已被编程至微处理器中授权的公开密钥寄存器内。流程继续进行至方块 702。

[0192] 在方块 702 中,利用在微处理器内的加密单元,解密此使能参数以撷取用来使能安全执行模式的一有效指令以及撷取在存储器内加密安全编码的指针。在 BIOS 中指向安全编码的另一指标以及任何加密的初始化数据也一起被提供。流程继续进行至方块 703。

[0193] 在方块 703 中,加密的安全编码通过系统总线而被撷取自存储器 /BIOS,且被解密。此安全编码与数据接着藉由使用一处理器密钥并根据一对称密钥算法来被加密,其中,此处理器密钥对于本发明的每一处理器而言是独特的,且在制造时被编程至一处理器密钥寄存器。此对称加密的安全编码与数据接着通过私密总线而被写入至一安全非易失性存储器,其中,此私密总线隔离于系统总线资源。写入至安全非易失性存储器的部分程序包括在写入对称加密编码与数据之前,对存储器执行随机写入。流程继续进行至方块 704。

[0194] 在方块 704 中,微处理器内非易失性使能指示寄存器被写入,以指示出安全执行模式被使能。在一实施例中,非易失性使能指示寄存器包括多字元,且这些位中的一者被写入以在安全执行模式每次被使能时用来指示出安全执行模式被使能。这些位中另一者被写入以指示出返回至原生未受控模式。因此,根据本发明的 256 位非易失性使能指示寄存器允许了 128 次由非安全执行模式至安全执行模式的转换。流程继续进行至方块 705。

[0195] 在方块 705 中,重置微处理器,即完成本发明微处理器中使能安全执行模式操作的方法。

[0196] 图 8 的流程图 800 强调用来在本发明的微处理器中禁能安全执行模式操作的高阶方法。即是,流程图 800 叙述操作在安全执行模式的安全编码如何命令微处理器返回至原生未受控模式。流程开始于方块 801,于其中,正于安全执行模式执行安全编码。流程继续进行至方块 802。

[0197] 在方块 802 中,安全编码于安全执行模式执行至非安全执行模式的返回 (return),亦即执行安全执行模式禁能指令。在一实施例中,当安全编码执行对一 SEM 机械

专用寄存器的写入时,开始实施至非安全执行模式的返回(返回至一非安全执行模式),其导致一安全例外(secure exception)。程控接着转移至在于安全编码内一地址上的安全例外管理者,其中,此地址由前述安全中断描述符号窗体的内容来提供。在一实施例中,安全例外管理者对一机械专用寄存器执行写入,以指示接受此返回。假使,此机械专用寄存器没有被正确地写入,此返回被忽略,且微处理器维持在安全执行模式。假使交握被确认,则流程继续进行至方块 803。

[0198] 在判断方块 803 中,评估非易失性使能指示寄存器的内容,以判断是否禁能安全执行模式(支持返回至非安全执行模式)。假使没有被禁能(支持返回至非安全执行模式),流程继续进行至方块 806。假使于此非易失性使能指示寄存器的多字元允许至非安全执行模式的返回,流程则继续进行至方块 804。

[0199] 在方块 806 中,维持安全执行模式,且控制权返回至安全编码。

[0200] 在方块 804 中,更新非易失性使能指示寄存器,以指示此微处理器正操作在非安全执行模式。流程继续进行至方块 805。

[0201] 在方块 805 中,微处理器的状态返回至原生未受控模式,即完成本发明的微处理器中禁能安全执行模式操作的方法。

[0202] 图 9 表示流程图 900,其详细说明本发明微处理器内初始化安全编码执行的方法。即是,流程图 900 的方法包括图 7 的流程图 700 的更详细说明。流程开始于方块 901,于其中,本发明的微处理器正于原生未受控模式中执行非安全应用程序。流程继续进行至方块 902。

[0203] 在方块 902 中,在非安全执行模式的一操作系统执行 SEMENABLE 指令或交替的机制(例如写入至一机械专用寄存器),其传送一个或多个使能参数,其中,此一个或多个使能参数是根据属于授权者的私密密钥来被非对称地加密。此一个或多个使能参数包括用来指向被执行的非对称加密安全编码的指针,此指针可储存在系统存储器以及 / 或 BIOS 存储器。流程继续进行至方块 903。

[0204] 在方块 903 中,微处理器使用一对对应的授权的公开密钥来对传送的一个或多个使能参数进行解密。在一实施例中,于微处理器的制造期间,此授权的公开密钥被编程至一非易失性授权的公开密钥寄存器。在另一交替的实施例中,此授权的公开密钥被编程至本发明的安全非易失性存储器内的一位置,且根据微处理器的初始开机,此授权的公开密钥自此安全非易失性存储器被撷取,且此授权的公开密钥被编程至非易失性授权的公开密钥寄存器,接着,在安全非易失性存储器内的此位置被清除。流程继续进行至方块 904。

[0205] 在方块 904 中,判断解密的使能参数是否有效。假使有效,流程继续进行至方块 905。假使无效,流程则继续进行至方块 907。

[0206] 在方块 905 中,由于已判断出此使能参数是有效的,则执行多随机写入于安全非易失性存储器的所有位置以清除安全非易失性存储器的内容。流程则继续进行至方块 906。

[0207] 在判断方块 906 中,加密的安全编码自系统存储器 / 以及或 BIOS 存储器被撷取。接着,使用授权的公开密钥并根据非对称密钥算法来对此加密的安全编码进行解密。在一实施例中,在微处理器中执行逻辑电路内的一加密单元用来解密此加密的安全编码。在一实施例中,此加密单元能执行 AES 加密操作、SHA-1 哈希操作以及 RSA 加密操作。解密后的安全编码接着被解压缩,且被检查格式是否正确。假使解密后的安全编码格式正确,流程继

续进行至方块 908。假使解密后的安全编码格式不正确，流程则继续进行至方块 907。

[0208] 在方块 907 中，由于解密后的使能参数是无效的，程控则返回至非安全执行模式。

[0209] 在方块 908 中，解密的安全编码（以及对应的初始数据，若有的话）藉由使用处理器密钥并根据对称密钥算法来加密，其中，此处理器密钥是此微处理器所独有的，且在制造时编程至一非易失性处理器密钥暂存器内。在一实施例中，此对称密钥为 128 位的 AES 密钥，且此微处理器利用其加密单元来对安全编码执行 AES 加密。流程继续进行至方块 909。

[0210] 在方块 909 中，此微处理器建立加密安全编码中一个或多个段落的一个或多个哈希。在一实施例中，微处理器内的加密单元用来建立加密编码的一个或多个 SHA-1 哈希。流程继续进行至方块 910。

[0211] 在方块 910 中，微处理器通过私密总线将加密的安全编码（以及数据，若有的话）以及此一个或多个哈希写入至安全非易失性存储器，其中，此私密总线隔离开于系统总线资源。此安全编码与数据被加密，因此阻止了安全编码内容的侦测。流程继续进行至方块 911。

[0212] 在步骤 911 中，设定非易失性使能指示寄存器以指示安全执行模式被使能。流程继续进行至方块 912。

[0213] 在方块 912 中，于微处理器内执行安全执行模式使能重置序列 (resetsequence)。此重置序列包括硬件检查（如同图 5 中相关的讨论）以及初始化安全易失性存储器为多随机数，即完成本发明的微处理器内初始化安全编码执行的方法。

[0214] 接着参阅图 10，流程图 1000 表示本发明微处理器中执行安全执行模式使能重置操作的方法，其中，此微处理器已使能安全执行模式的操作。流程开始于方块 1001，其中，当微处理器完成安全执行模式的初始化时，微处理器执行安全执行模式使能重置串行。流程继续进行至方块 1002。

[0215] 在方块 1002 中，微处理器执行多处理器诚实性检查，包括安全非易失性存储器、电池与石英器的侦测与确认。此外，核对总线时钟的存在与频率诚实性，并确认提供给总线终端与微处理器供应电源的适当电压。微处理器的温度确认处于一可接受的范围内。流程继续进行至方块 1003。

[0216] 在方块 1003 中，微处理器执行非易失性存储器连结 (connectivity) 与哈希检查。自安全非易失性存储体内一位置读取安全签章，并对此安全签章进行解密。解密后的签章被核对以证实非易失性存储器没有被泄漏。此外，微处理器亦读取安全非易失性存储器的特定位置与对应的哈希。通过加密（即 AES/HASH/RSA）单元，产生被选择位置的确认哈希，且与被读取的哈希进行比较。流程继续进行至方块 1004。

[0217] 在方块 1004 中，微处理器执行安全实时时钟的确认。在一实施例中，安全执行模式实时时钟估计石英器的状态，以侦测在频率上大于百分的五的改变，因此表示出石英器与在电池电压上大于百分的五的改变，且表示出潜在的安全威胁征兆。假使上述确认检查的任一者产生不利的结果，根据侦测到事件的严重性与次数，安全执行模式使能重置串行将使此事件被记录下来，或者迫使微处理器进入降级模式，或硬件关机模式。流程继续进行至方块 1005。

[0218] 在方块 1005 中，自非易失性存储器（系统存储器以及 / 或 BIOS 存储器）撷取加密的安全编码以及数据。流程继续进行至方块 1006。

[0219] 在方块 1006 中，译码与解压缩加密的安全编码，且确认格式正确后，安全编码接

着被加载至微处理器内的安全易失性存储器。流程继续进行至方块 1007。

[0220] 在方块 1007 中, 初始化微处理器内的安全资源。这些安全资源无法被非安全编码所得知或存取, 且只对于在安全执行模式中执行的安全编码而言是可利用的。这些资源包括安全定时器、安全中断以及安全例外, 且包括安全中断描述符窗体、以及任何安全机械专用寄存器或为了安全编码的执行而必须被初始化的其它寄存器。初始化包括非安全中断、非安全例外、非安全追踪以及除错逻辑电路的禁能, 也包括微处理器的任何电源管理逻辑电路的禁能, 其中包括导致核心电压、核心时钟频率的变化或者使能或禁能其它组件(例如高速缓存、分支预测单元等等)的任何组件。流程继续进行至方块 1008。

[0221] 在方块 1008 中, 初始化微处理器内的非安全的高速缓存(即 L1 高速缓存、L2 高速缓存)为乱数。流程继续进行至方块 1009。

[0222] 在方块 1009 中, 产生一安全执行模式中断, 且根据存在于安全中断描述符表内的数据来呼寻(call)安全执行模式重置功能, 其中, 此安全中断描述符白在方块 1007 中被初始化, 即完成本发明微处理器中执行安全执行模式使能重置操作的方法。

[0223] 接着参阅图 11, 流程图 1100 表示本发明微处理器中终止安全执行模式操作的方法。此方法开始于方块 1101, 于其中, 安全编码正执行于安全执行模式。概括上, 根据本发明, 具有三种方法使微处理器由非安全执行模式转换为安全执行模式, 并开始安全编码的执行。第一种方法允许程控转移为安全编码的执行。即是, 在安全执行模式下的非安全应用程序如同 SEMENTER 指令般执行。在一实施例中, SEMENTER 指令导致微处理器的状态被储存在安全易失性存储器内的堆栈, 且程控转移至安全编码, 非常类似 x86SYSENTER 指令的操作。第二种方法是, 当执行非安全或安全重置序列时, 导致安全编码的执行是由于一中断或例外所致。导致安全编码执行的最后一个方法, 是起因于来自任何数量的安全监控逻辑组件的中断, 就像关于图 5 的讨论。

[0224] 如上所述, 执行在安全执行模式的安全编码, 永久地存在于安全非易失性存储器, 但是在一安全执行模式使能重置串行的期间, 其已被加载至安全易失性存储器。即是, 此安全编码不再自非安全存储器中执行, 例如系统存储器或非安全的处理器高速缓存。因此, 藉由两种方法, 执行控制由安全执行模式转换回非安全执行模式。第一种方法包括执行 SRESUME 指令, 其引起来自 SEMENTER 指令的响应(return)。在 x86 实施例中, 此 SRESUME 指令以与 x86RESUME 相似的方法来操作。即是, 预先储存在安全易失性存储器中的程序状态被恢复(restored), 且程控转移至操作系统或非安全编码。第二种方法是考虑强迫一安全例外, 其中, 藉由对只可由安全编码来存取的一机械专用寄存器执行写入, 微处理器的安全组件可存取此安全例外。假使确认微处理器将返回至非安全执行模式, 接着产生被操作系统指明且处理的一非安全机械检查例外, 因此影响至非安全执行模式的返回。图 11 的流程图 1100 提出强迫此安全例外以返回至非安全执行模式, 而此技术领域中具有通常知识者将理解, SRESUME 指令的执行导致微处理器去执行下文所述的相似步骤。

[0225] 因此, 流程持续于方块 1102, 于其中, 将安全编码写入至安全执行模式机械专用寄存器(SEM MSR)。SEM MSR 即是, 只可被执行在安全执行模式下的安全编码所存取且得知的多机械专用寄存器中的一者。流程继续进行至方块 1103。

[0226] 在方块 1103 中, 写入至安全执行模式机械专用寄存器产生了由 SEM 逻辑电路内安全例外逻辑电路所处理的安全例外。流程继续进行至方块 1104。

[0227] 在方块 1104 中, 安全例外逻辑电路(例如安全中断描述符号窗体)导致程控分支至安全编码内的安全例外管理者。流程继续进行至方块 1105。

[0228] 在方块 1105 中, 安全例外管理者响应一授权的例外编码。此安全例外管理者执行至安全编码的返回, 藉以将一授权的例外编码传送回安全编码。流程继续进行至方块 1106。

[0229] 在方块 1106 中, 判断由安全例外管理者所响应的例外编码是否正确。假使此例外编码不正确, 则假设有一安全风险, 且流程继续进行至方块 1112。假使此例外编码正确, 则安全编码与安全例外管理者之间的交握则被确认以指示返回至非安全执行模式, 且流程继续进行至方块 1107。

[0230] 在方块 1112 中, 维持安全执行模式, 且控制权返回至安全编码。

[0231] 在方块 1107 中, 微处理器执行多随机写入于安全非易失性存储器的所有位置以清除安全非易失性存储器的内容。安全应用程序利用微处理器内的一随机数产生器来产生随机数数据且对安全非易失性存储器内的所有位置执行随机写入。流程继续进行至方块 1108。

[0232] 在方块 1108 中, 微处理器藉由将“0”写入至安全非易失性存储器的每一位置, 来清除安全非易失性存储器的每一位置。流程继续进行至方块 1109。

[0233] 在方块 1109 中, 设定非易失性使能指示寄存器以指示安全执行模式被禁能, 亦即, 微处理器正操作在一非安全执行模式中。其受限于安全执行模式可被禁能的次数, 如同前文关于图 8 的说明。流程继续进行至方块 1110。

[0234] 在方块 1110 中, 安全例外逻辑电路产生一机械检查例外, 此外回应一状态参数(亦即例外编码指示状态)来将程控转移至非安全应用程序中之一。因此, 在非安全执行模式下的操作系统处理此机械检查例外, 且完成返回至非安全执行模式。流程继续进行至方块 1111。

[0235] 在方块 1111 中, 即完成本发明微处理器中终止安全执行模式操作的方法。

[0236] 图 12 表示一安全实时时钟 1200 的详细方块图, 其位于本发明的微处理中的 SEM 逻辑电路内。安全实时时钟 1200 只可由正操作在安全执行模式下的安全编码来得知且存取。安全实时时钟包括振荡器 1201, 其通过信号 VP 耦接电池且通过信号 C1 及 C2 来耦接石英器。此振荡器产生振荡输出电压信号 V0, 且信号 V0 耦接计数器 1202。此计数器产生输出信号 CNT0, 且输出信号 CNT0 被路由至转换逻辑电路 1203。信号 VP、C1、及 C2 也输入至转换逻辑电路 1203, 此外, 信号 ENV 同样输入至转换逻辑电路, 其中, 信号 ENV 载有对应管芯温度的数值。转换逻辑电路 1203 产生通过信号 TEMP、BATT、COMP、XTAL 以及双向总线 TIME 来提供的多输出。此微处理器通过双向总线 TIME 提供输入至此安全实时时钟。

[0237] 振荡器 1201 与计数器 1202 是专用的, 即是除了被提供来允许微处理器通过双向总线 TIME 对安全实时时钟进行读取和写入的组件以外, 他们无法共享其它电路系统或微处理器的其它组件。此外, 只要电池通过信号 VP 提供可接受的电压时, 安全实时时钟持续其计数。在一交替的实施例中, 电池电压信号 VP 是由系统板上的电容器所产生, 以代替只要系统板开机而持续被充电的电池。

[0238] 在操作上, 振荡器 1201 产生振荡输出电压信号 V0, 其与石英器的频率成比例, 且此振荡输出电压被提供至计数器 1202。计数器 1202 包括多组件, 用来计算通过信号 V0 所提供的周期数, 并将此周期数转换为一计数数值。此计数数值被提供至信号 CNT0 上。转

换逻辑电路 1203 包括多电路,用将 CNT0 的数值转换为持续时间数值,此外,转换逻辑电路 1203 也包括多寄存器(未显示),其可通过双向总线 TIME 而被微处理器来读取与写入。

[0239] 此外,转换逻辑电路 1203 用来侦测电压信号 VP 的显著变化,指示出潜在的篡改,且此一事件由信号 BATT 的设置来表示,其中,信号 BATT 的设置系用来中断正执行的安全编码。在一实施例中,大于百分的五的变化导致 BATT 中断被设置。

[0240] 转换逻辑电路 1203 也用来通过信号 C1 与 C2 来侦测石英器频率的显著变化,因此指示潜在的篡改,且此一事件藉由信号 XTAL 的设置来表示。信号 XTAL 的设置系用来中断正执行的安全编码。在一实施例中,大于百分的五的变化导致 XTAL 中断被设置。

[0241] 信号 ENV 系由转换逻辑电路 1203 来估计,以判断因温度偏离而使计数器 1202 产生不精准的计数。假使判断出温度偏离,信号 TEMP 则被设置,其用来中断正执行的安全编码。

[0242] 转换逻辑电路 1203 也用来估计上述情况中任一者是否足够显著,以指示安全实时钟已被泄漏,例如电池的移动与取代。假使被判断出,信号 COMP 也被设置,因此中断安全编码的执行。

[0243] 本发明提供一些高于现今技术的优点以在安全环境中执行应用程序。例如,根据本发明的设计是以微处理器为基础。即是,本发明的一目的是修改负责安全编码的微处理器,这是因为,相对于着重在修改芯片组或其它组件的其它技术,只有微处理器可提供及时执行安全。使用隔离芯片来监控微处理器的方法有许多的内在安全性缺陷,且对于安全相关的执行而言效能也明显地降低。

[0244] 根据本发明中以 x86 为基础的实施例,由于 x86 程序化技术的普遍性,安全编码的发展相当地平易。x86 架构已被得知,且对于精通非安全 x86 应用发展的任何程序设计者而言,机械专用指令的附加与专用指令(例如 SEMENABLE、SEMENTER、及 SRESUME 指令)仅提供较少的学习挑战。

[0245] 此外,对于微处理器的附加安全执行能力的成本远小于额外芯片组被加至系统设计所呈现的成本。

[0246] 此外,由于安全执行环境被提供至微处理器本身之内,因此内在地对抗那些物理或从属通道攻击,其不需要附加外部电路。

[0247] 此处所揭露的技术非常有利地提供安全的微处理器操作环境,在此环境中,会被泄漏的一般机密(例如一般加密密钥或程序架构)不会储存于其中。即是,本发明的每一处理器只具有需要被特定处理器或系统授权、控制等等的机密。来自一处理器 / 系统的机密不会破坏在另一处理器 / 系统的安全性。此外,得知如何破坏在一处理器的安全性,应当不会使其更容易地去破坏其它处理器上的安全性。即是,这是由于独特的处理器密钥,此独特的处理器密钥是由在安全非易失性存储器总线上的数据传输所提供且导致的,其中,这些数据传输系使用此密钥来加密。

[0248] 与提供对抗俗称拒绝服务攻击(denial-of-service attack)的保护的习知技术比较起来,根据本发明的微处理器具有更多的优点。例如,如图 5 所讨论,提供安全监控组件以侦测并取得在事件上的活动,例如持续对安全执行环境的呼寻(例如来自恶意装置驱动器),实时时钟电池、石英器的持续移除等等。

[0249] 本发明虽以较佳实施例揭露如上,然其并非用以限定本发明的范围,任何所属技

术领域中具有通常知识者，在不脱离本发明的精神和范围内，当可做些许的更动与润饰，因此本发明的保护范围当视后附的权利要求书所界定者为准。

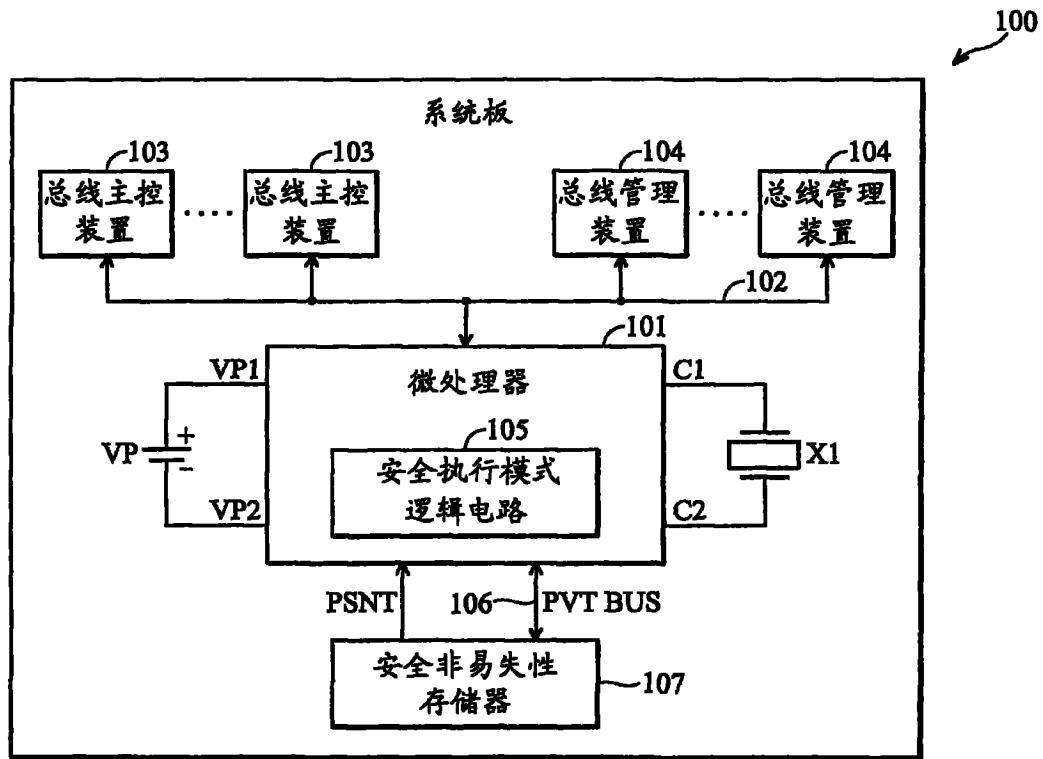


图 1

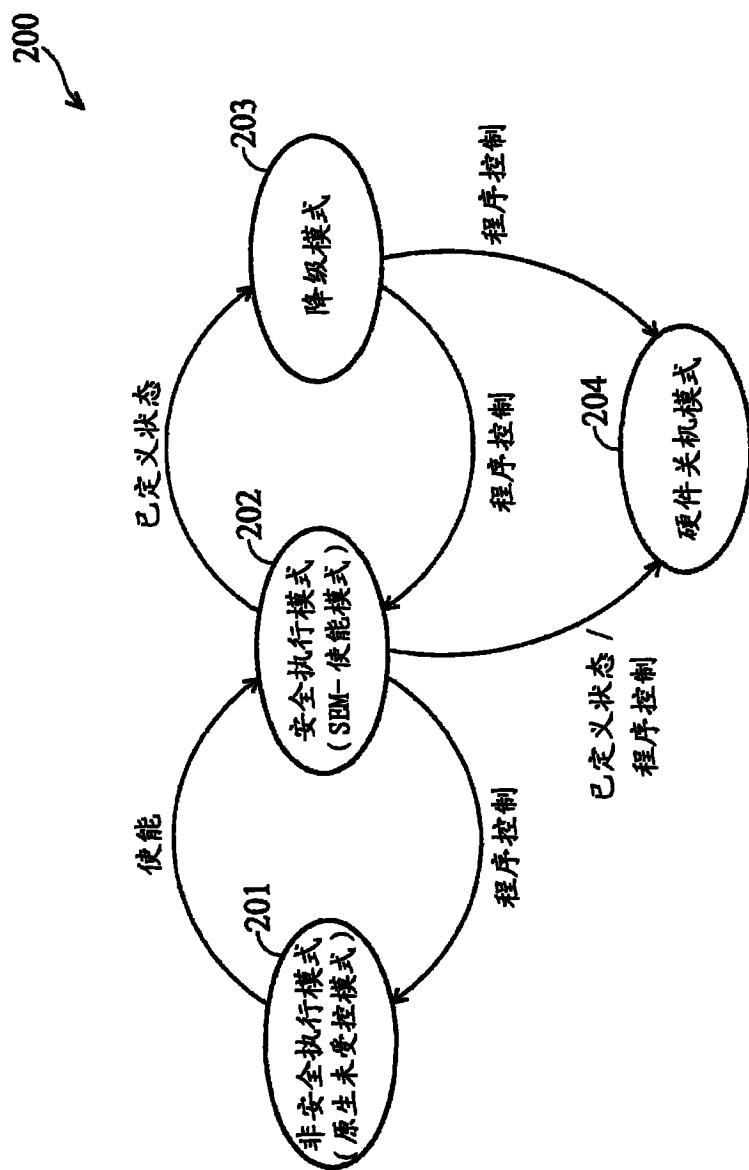


图 2

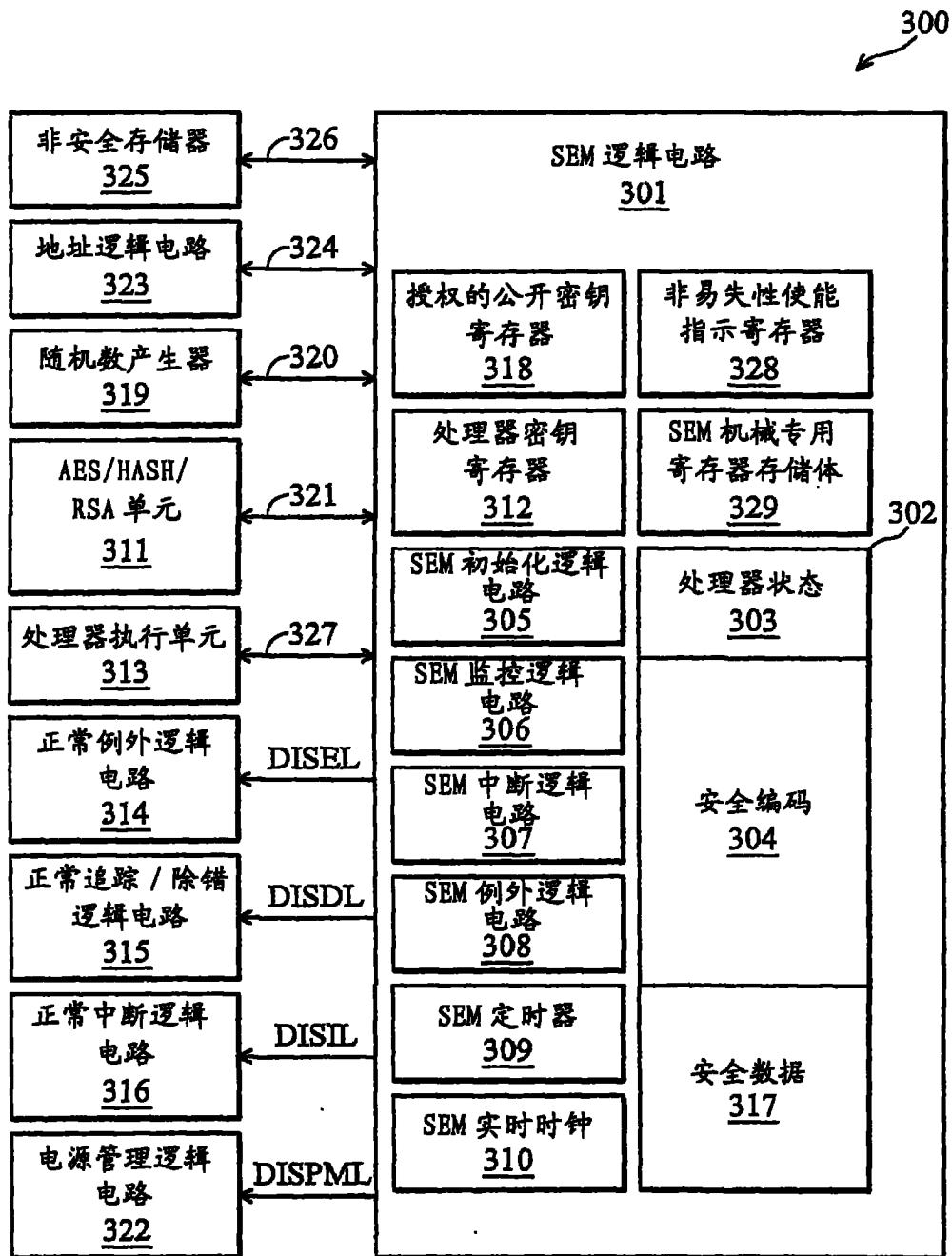


图 3

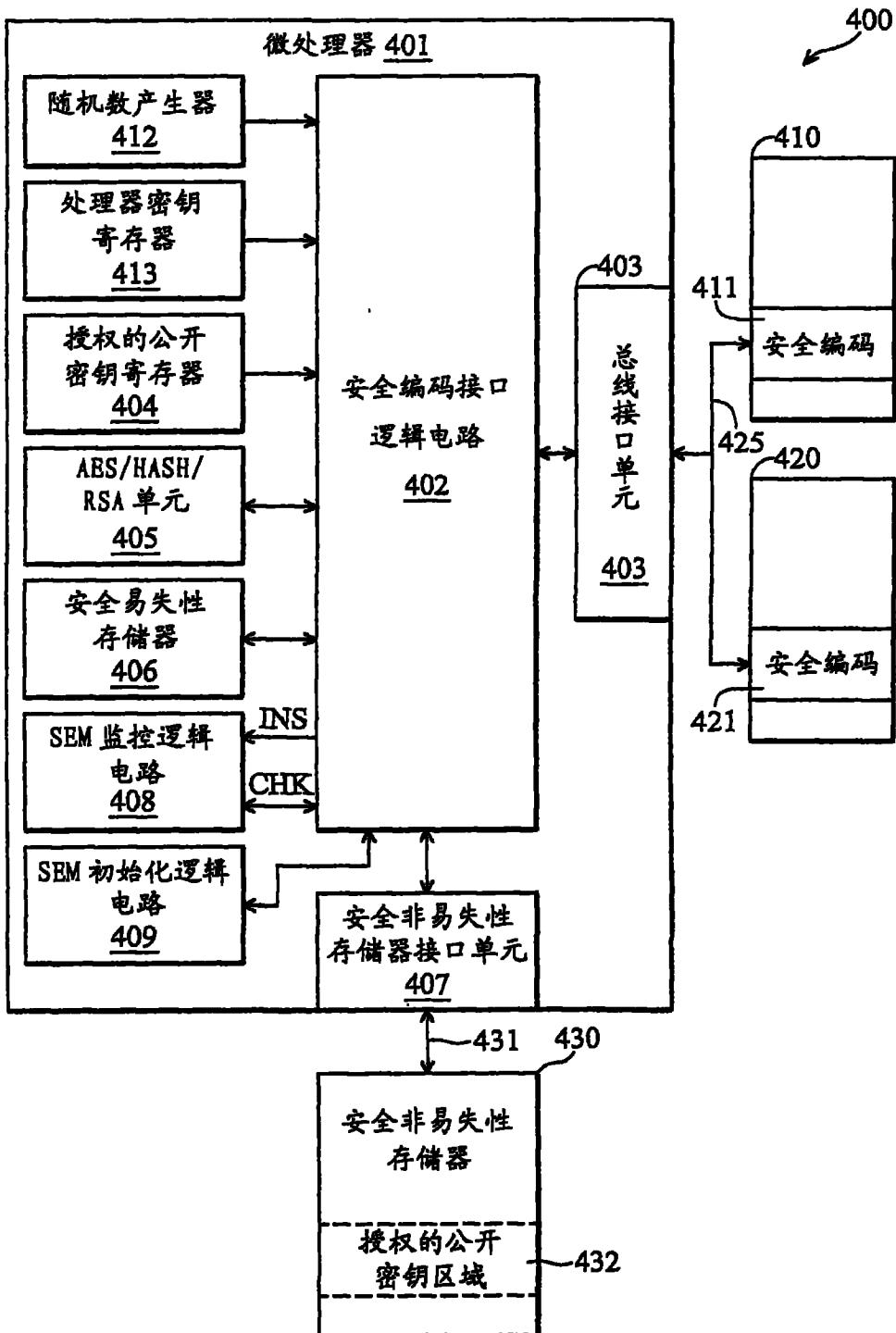


图 4

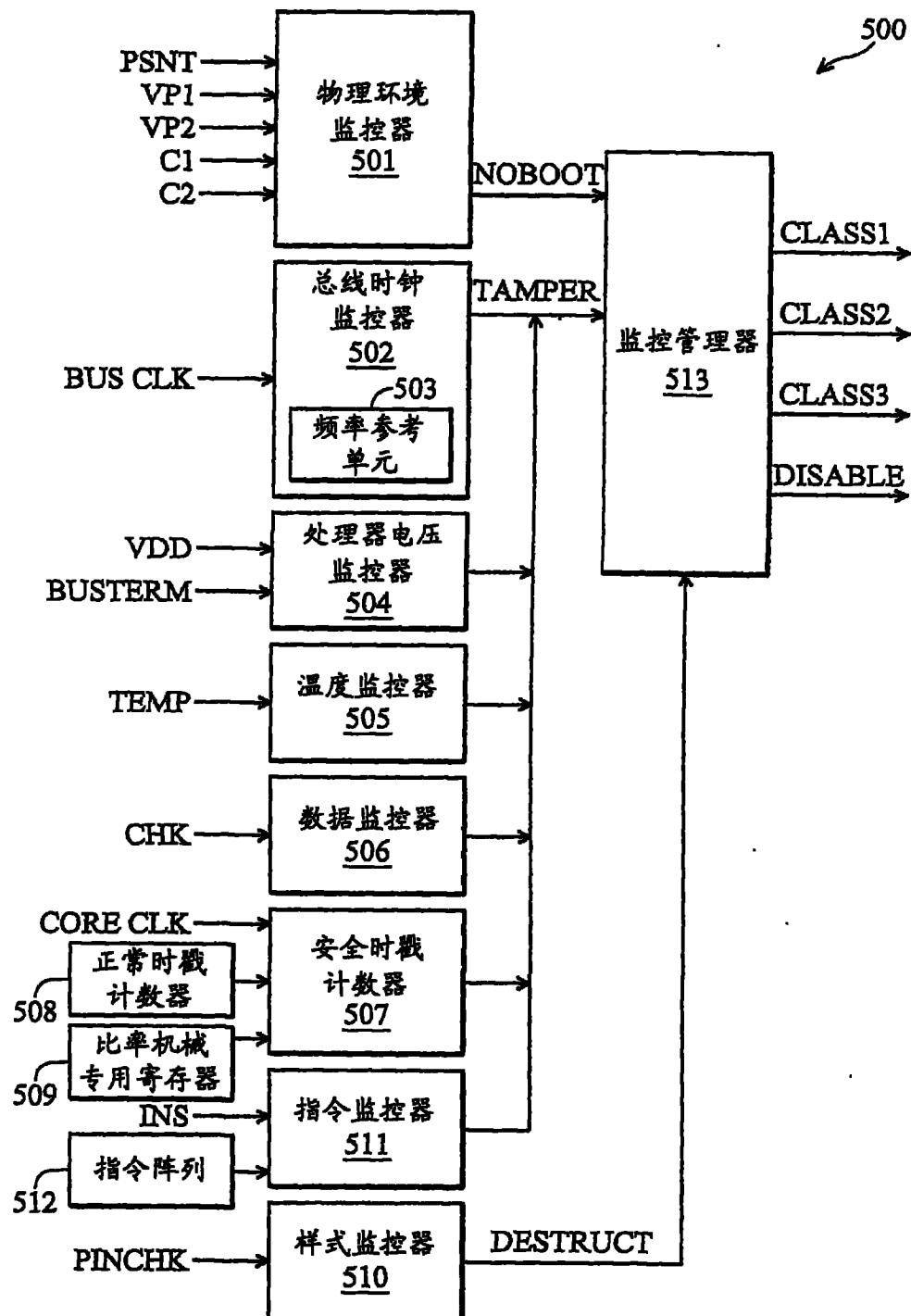


图 5

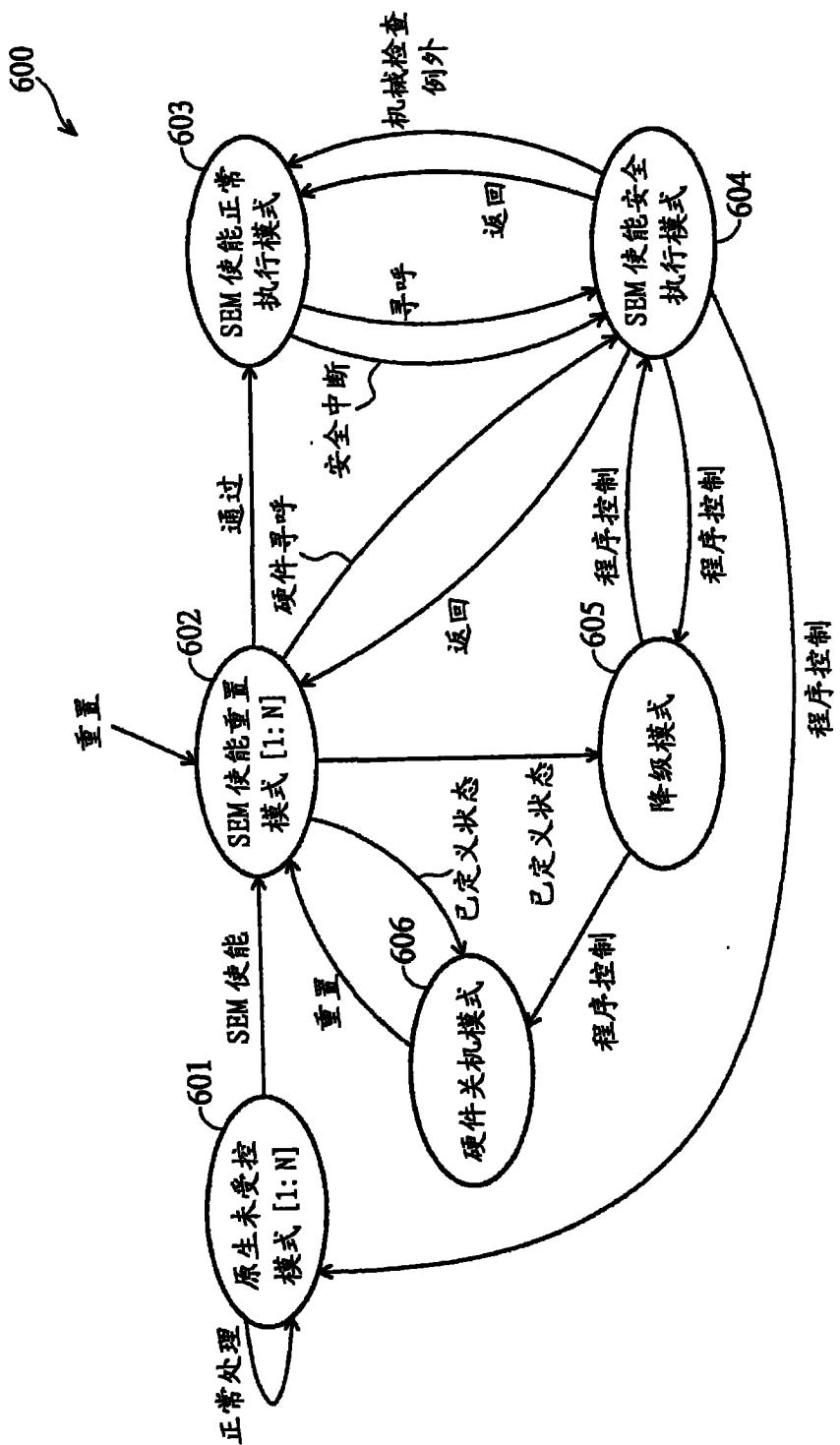


图 6

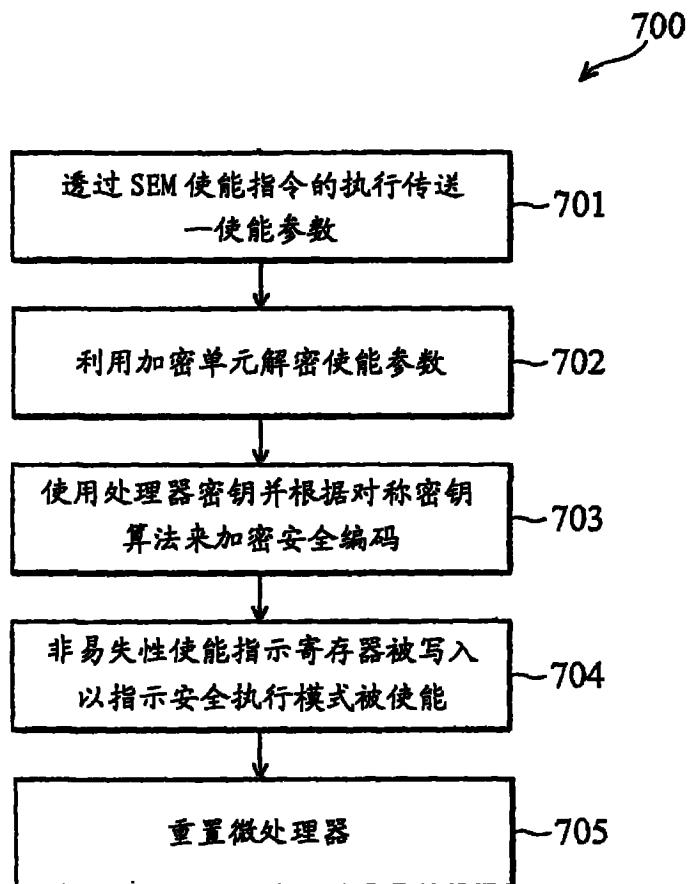


图 7

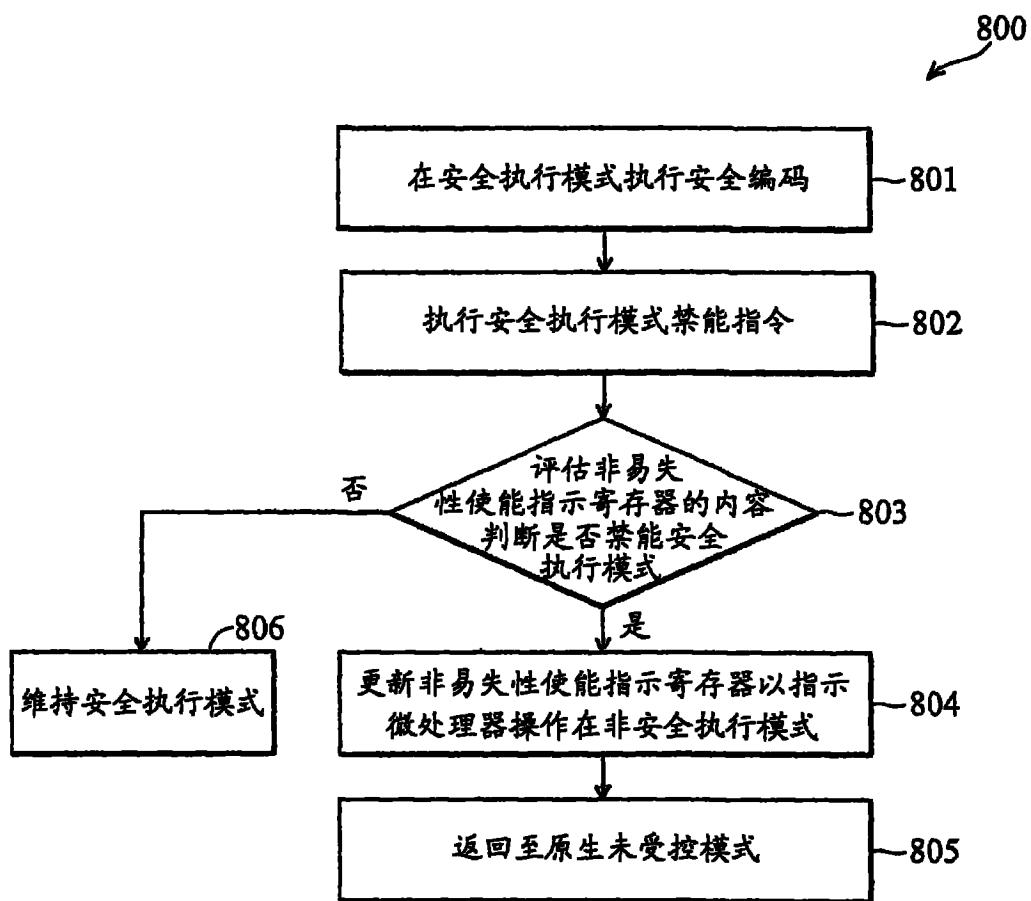


图 8

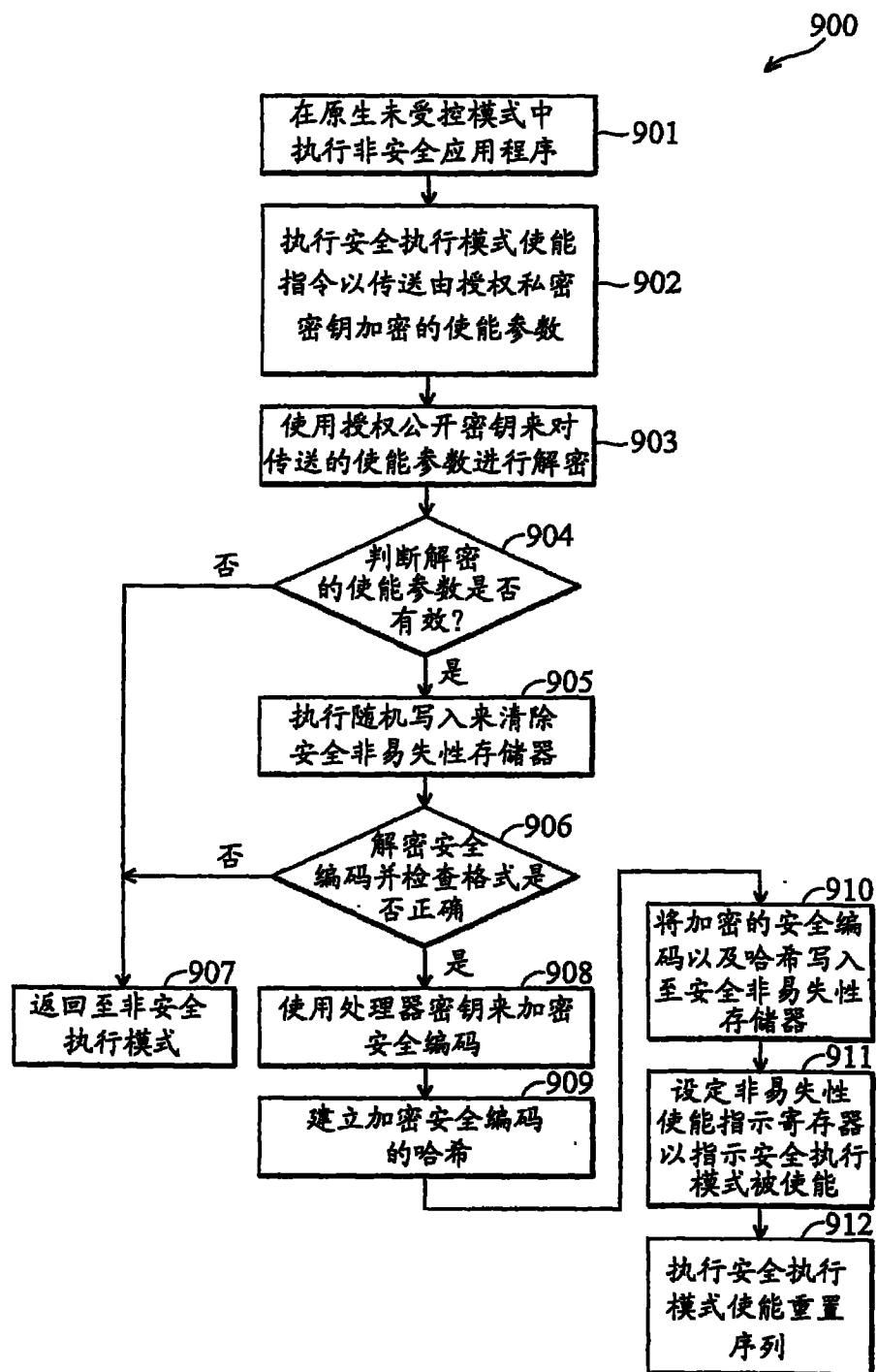


图 9

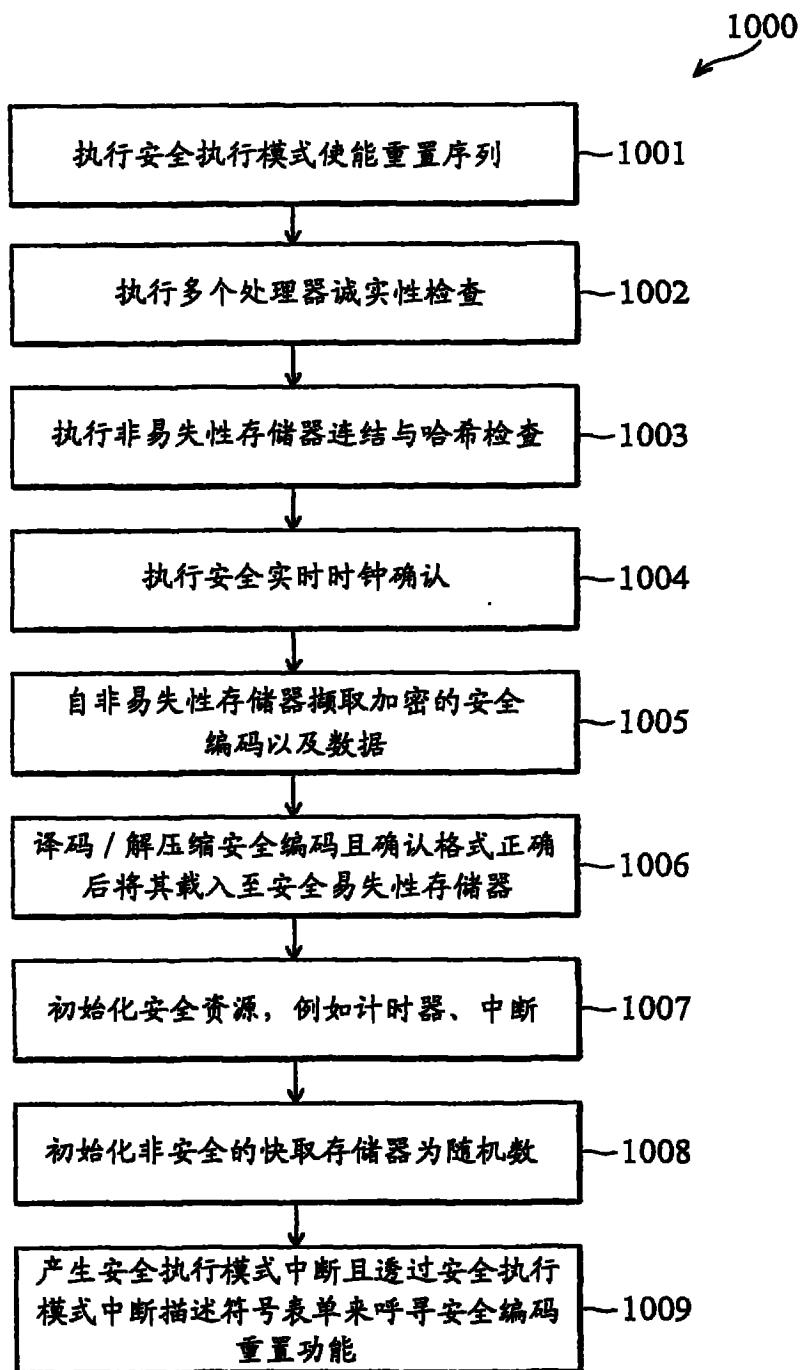


图 10

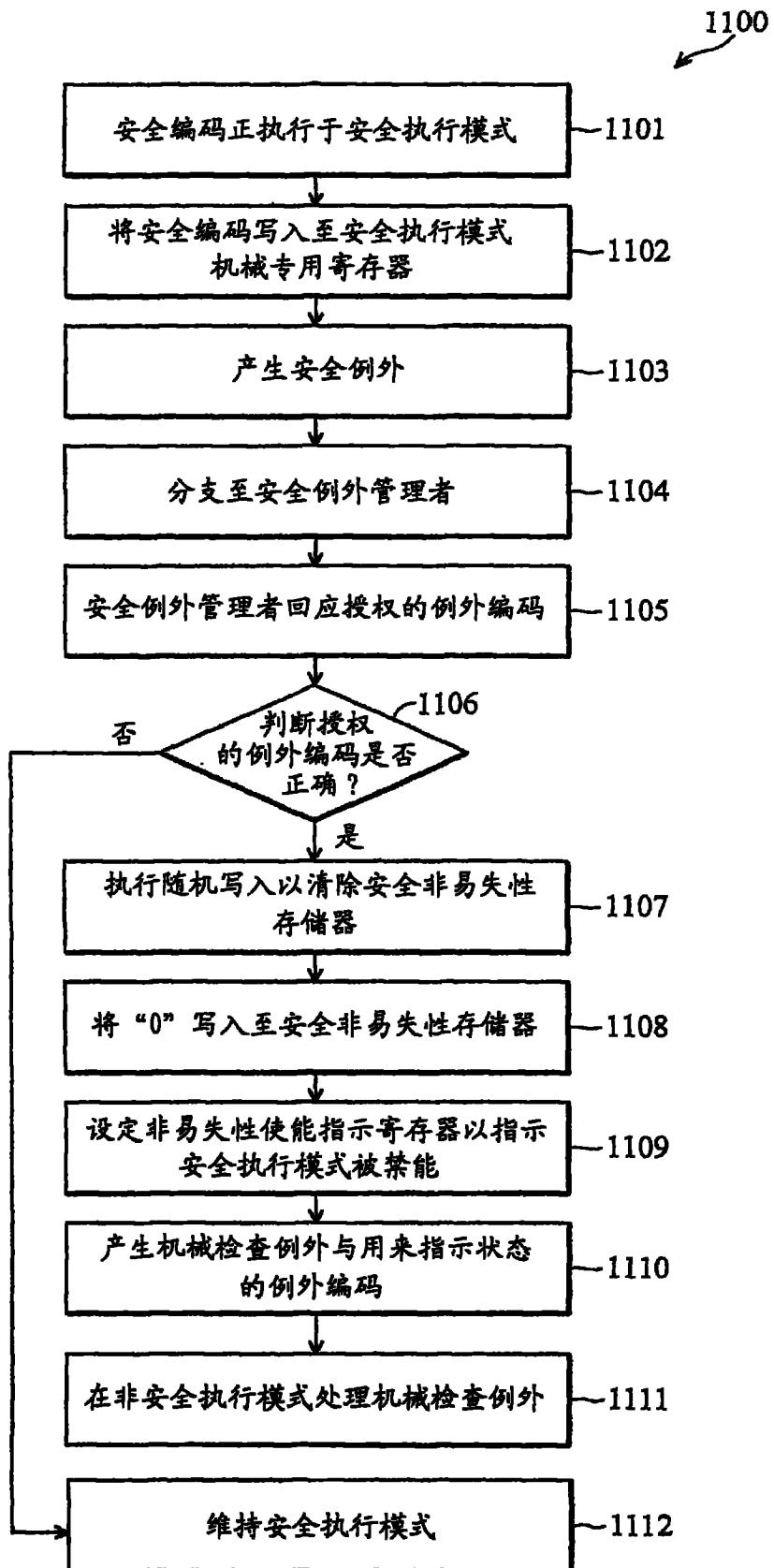


图 11

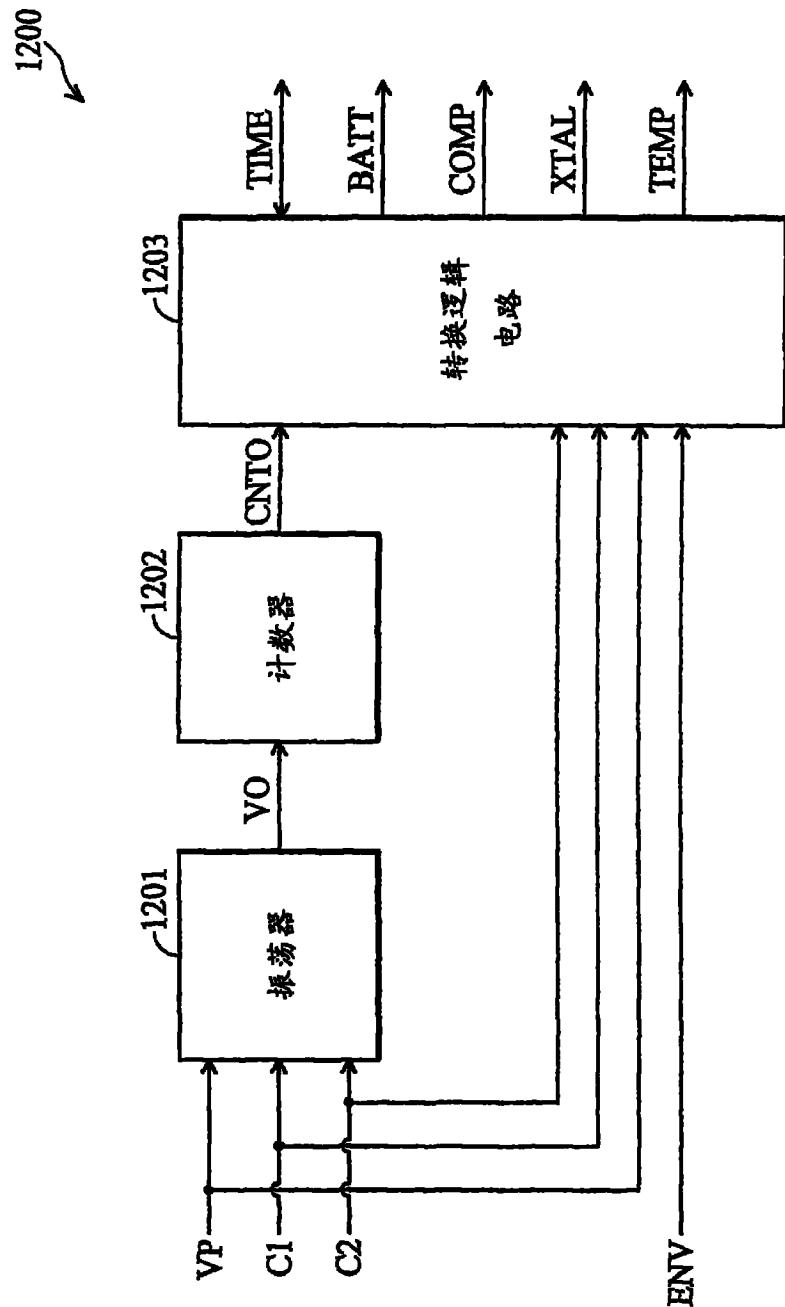


图 12