(19)中华人民共和国国家知识产权局

(12)发明专利

(10)授权公告号 CN 101427222 B
(45)授权公告日 2012.11.21

(21)申请号 200780014735.8
(22)申请日 2007.04.24
(30)优先权数据
118881/2006 2006.04.24 JP
(85)PCT申请进入国家阶段日
2008.10.24
(86)PCT申请的申请数据
PCT/JP2007/058838 2007.04.24
(87)PCT申请的公布数据
WO2007/125911 JA 2007.11.08
(73)专利权人 松下电器产业株式会社
地址 日本大阪府
(72)发明人 前田学 松岛秀树 井藤好克
(74)专利代理机构 永新专利商标代理有限公司 72002
代理人 徐殿军
(51)Int.Cl.
G06F 11/28(2006.01)
G06F 21/22(2006.01)
(56)对比文件
CN 1516836 A,2004.07.28, 全文.
WO 2005091143 A1,2005.09.29, 第[0022]段至[0045]段.
审查员 魏峰

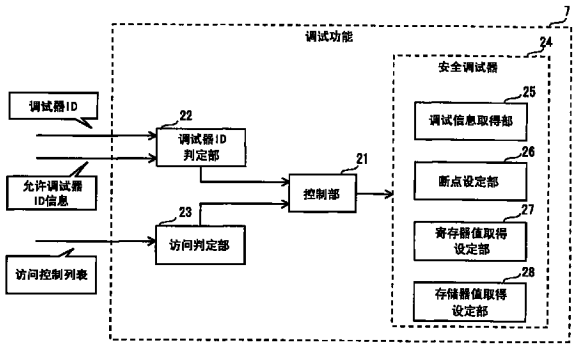
权利要求书 5 页 说明书 31 页 附图 15 页

(54)发明名称

数据处理装置、方法、程序生成装置、方法

(57)摘要

数据处理装置控制调试器对程序执行调试处理。程序中包含用于判定可否调试处理的验证值、和对程序的每个部分示出可否访问程序的访问控制列表。数据处理装置取得调试器的调试器ID、与程序中包含的验证值和访问控制列表。对应于调试器ID与验证值的比较结果,判定可否调试处理。另外,构成调试对象的程序的部分若在访问控制列表中示为可访问,则允许访问,若为不可访问,则不允许访问。



CN 101427222 B

1. 一种数据处理装置，控制调试处理部的调试处理的执行，其特征在于：

具备：

第 1 取得部件，取得识别所述调试处理部的识别符；

第 2 取得部件，取得处于被保护不受非法访问的状态下的调试对象程序的规定部分中包含的验证值；

判定部件，对从所述调试对象程序取得的所述验证值与由所述第 1 取得部件取得的所述识别符进行比较，按照该比较结果，判定是否允许对所述调试对象程序的调试处理；和

控制部件，当判定为不允许时，禁止对所述调试对象程序执行调试处理，

所述调试对象程序的所述规定部分中，包含表示对于构成所述调试对象程序的各部分允许或不允许访问的访问控制列表，

所述第 2 取得部件包含访问控制列表取得部，该访问控制列表取得部取得所述调试对象程序的所述规定部分中包含的所述访问控制列表，

所述数据处理装置还具备访问判定部件，该访问判定部件根据所述取得的所述访问控制列表，判定是否允许访问所述调试对象程序的一部分，

所述控制部件，在所述判定部件判定为允许、且所述访问判定部件判定为不允许的第 1 情况下，禁止执行所述一部分的调试处理，在所述判定部件判定为允许、且所述访问判定部件判定为允许的第 2 情况下，使所述调试处理部执行所述一部分的调试处理。

2. 根据权利要求 1 所述的数据处理装置，其特征在于：

所述访问控制列表所示的所述部分的每一个，表示与将所述调试对象程序加载到存储器时的加载目的地存储器地址的所述部分分别对应的地址范围，

在所述访问控制列表中，将与所述部分分别对应的所述地址范围的每一个，与可否访问相对应，

所述访问判定部件对于与将所述调试对象程序加载到存储器时的加载目的地存储器地址的所述一部分相对应的地址范围，通过参照在所述访问控制列表中相对应的可否访问，执行所述判定。

3. 根据权利要求 1 所述的数据处理装置，其特征在于：

所述访问控制列表中所示的所述部分的每一个，表示所述调试对象程序中包含的符号，

所述访问控制列表中，将所述符号的每一个与可否访问相对应，

所述访问判定部件对于所述调试对象程序的所述一部分中包含的符号，通过参照在所述访问控制列表中相对应的可否访问，执行所述判定。

4. 根据权利要求 1 所述的数据处理装置，其特征在于：

所述调试对象程序的所述规定部分中包含多个所述验证值，

所述规定部分中包含一个以上所述访问控制列表，所述访问控制列表的每一个，与所述验证值的至少一个相对应，

所述判定部件对于所述验证值的每一个，与所述取得的识别符进行比较，执行所述判定，

所述访问判定部件根据与所述判定部件判定为允许的所述验证值相对应的所述访问控制列表，执行所述判定。

5. 根据权利要求 1 所述的数据处理装置，其特征在于：

所述数据处理装置还具备显示部，

所述控制部件包含显示控制部，在所述第 1 情况下，该显示控制部使所述显示部执行表示禁止所述一部分的调试处理之意的显示，在所述第 2 情况下，该显示控制部使所述显示部显示所述调试处理的结果。

6. 根据权利要求 1 所述的数据处理装置，其特征在于：

所述判定部件对所述验证值与所述识别符进行比较，在所述验证值与所述识别符一致的情况下，判定为允许所述调试处理。

7. 根据权利要求 1 所述的数据处理装置，其特征在于：

所述判定部件，

包含比较值保持部，该比较值保持部以保护其不被非法访问的状态存储比较值，并且

执行将所述验证值与所述识别符用作运算符的规定运算，在该运算结果与所述存储的所述比较值一致的情况下，判定为允许所述调试处理。

8. 根据权利要求 1 所述的数据处理装置，其特征在于：

所述数据处理装置具有具备防止从外部非法访问的机构的安全域，

所述数据处理装置具备通常模式与安全模式作为动作模式，切换所述通常模式与所述安全模式而动作，仅在所述安全模式时使用所述安全域而动作，

所述数据处理装置还具备切换部，该切换部切换所述通常模式与所述安全模式，

在所述通常模式下动作的程序，通过经由所述切换部将规定的处理请求通知给在所述安全模式下动作的程序，从而能访问在所述安全模式下动作的程序，

所述调试对象程序存储在所述安全域中，

所述第 2 取得部件在所述安全域中，从所述调试对象程序执行所述验证值的所述取得，

所述判定部件在所述安全域中执行所述判定。

9. 根据权利要求 8 所述的数据处理装置，其特征在于：

所述调试处理部在所述安全域之外在所述通常模式下动作，

所述数据处理装置还具备在所述安全模式下执行调试处理的安全调试器，所述安全调试器包含于所述安全域中，

所述调试处理部输出所述调试对象程序的调试处理请求，

所述控制部件，在所述调试处理部输出所述调试处理请求时，使所述判定部件执行所述判定，当判定为不允许时，禁止所述安全调试器对涉及所述调试处理请求的所述调试对象程序进行调试处理。

10. 根据权利要求 9 所述的数据处理装置，其特征在于：

所述调试对象程序的所述规定部分中，包含表示对于构成所述调试对象程序的各部分允许或不允许访问的访问控制列表，

所述第 2 取得部件包含访问控制列表取得部，该访问控制列表取得部取得所述调试对象程序的所述规定部分中包含的所述访问控制列表，

所述数据处理装置还具备访问判定部件，该访问判定部件根据所述取得的所述访问控制列表，判定是否允许访问所述调试对象程序的所述一部分，

所述访问判定部件在所述安全域中执行所述判定，

所述控制部件，对于所述调试处理部输出的所述调试处理请求所涉及的所述调试对象程序，在所述判定部件判定为允许、且所述访问判定部件判定为不允许的情况下，禁止所述安全调试器执行所述一部分的所述调试处理，在所述判定部件判定为允许、且所述访问判定部件判定为允许的情况下，使所述安全调试器执行所述一部分的调试处理。

11. 根据权利要求 9 所述的数据处理装置，其特征在于：

所述调试处理部具有对所述通常程序和与所述通常程序协同的所述调试对象程序执行调试处理的功能，

所述调试处理部进行的所述调试对象程序的调试处理，通过所述调试处理部输出调试处理请求、并且所述调试处理部经所述切换部受理由所述安全调试器对所输出的所述调试处理请求执行的调试处理的结果来执行。

12. 根据权利要求 11 所述的数据处理装置，其特征在于：

所述调试处理部输出识别该调试处理部所附着的通常程序的处理识别符，

所述安全调试器，将位于与从所述调试处理部输出的所述处理识别符所示的通常程序协同动作的调试对象程序的入口点的命令，变更为断开命令。

13. 根据权利要求 11 所述的数据处理装置，其特征在于：

所述数据处理装置，当在所述安全模式下在执行所述调试对象程序过程中检测出调试例外时，经所述切换部向所述调试处理部通知调试例外的发生，

所述调试处理部，当从所述切换部受理所述调试例外的发生通知时，输出表示调试处理的执行结果的调试信息的取得请求，

所述安全调试器，当所述判定部件对于所述调试对象程序做出肯定的判定时，若受理所述调试信息的取得请求，则执行所述调试对象程序的调试处理，取得调试信息，经所述切换部将取得的调试信息输出到所述调试处理部。

14. 根据权利要求 11 所述的数据处理装置，其特征在于：

所述数据处理装置还具备：

第 1 结果显示部，将所述通常程序的调试处理的结果显示于第 1 显示区域；和

第 2 结果显示部，将与所述通常程序的协同所涉及的调试对象程序的调试处理的结果，显示于与所述第 1 显示区域不同的第 2 显示区域，

所述第 1 和第 2 结果显示部，在所述协同所涉及的调试对象程序与所述通常程序协同动作时，在所述第 1 显示区域和所述第 2 显示区域中，显示所述调试对象程序与所述通常程序的调试处理的结果。

15. 根据权利要求 11 所述的数据处理装置，其特征在于：

在所述通常模式下，通常 OS 动作，

在所述安全模式下，保护 OS 动作，

所述通常程序作为所述通常 OS 生成的处理，在所述通常模式下动作，

所述调试处理部作为在所述通常 OS 中动作的调试器，在所述通常模式下动作，

所述调试对象程序作为所述保护 OS 生成的处理，在所述安全模式下动作，

所述安全调试器作为所述保护 OS 具有的功能被安装。

16. 根据权利要求 9 所述的数据处理装置，其特征在于：

所述控制部件，当所述调试处理部输出所述调试处理请求时，使所述判定部件执行所述判定，当判定为不允许时，向所述调试处理部输出表示禁止执行所述调试处理的不可调试处理通知。

17. 一种程序生成装置，其特征在于：

具备：

程序取得部件，取得包含应保密的保护信息的程序；

验证值生成部件，生成用于按照调试处理部的识别符判定是否允许对所取得的所述程序的调试处理的验证值；和

保护程序生成部件，向所述程序附加所述验证值生成部件对所述程序生成的验证值，生成保护程序，

由所述程序取得部件取得的、作为源代码的所述程序中，对应保密的所述保护信息所在的部位预先附加标记，

作为编译器的所述程序生成装置，根据所述标记的有无，将记述有应保密的所述保护信息的区域写入到秘密信息存储文件，

所述程序生成装置还包含访问控制列表取得部，该控制列表取得部取得表示对于构成所述程序的各部分允许或不允许访问的访问控制列表，

所述保护程序生成部件包含访问控制列表附加部，该访问控制列表附加部将所述取得的访问控制列表附加于所述程序。

18. 一种数据处理方法，控制调试处理部的调试处理的执行，其特征在于：

包含：

第 1 取得步骤，取得识别所述调试处理部的识别符；

第 2 取得步骤，取得处于被保护不受非法访问的状态下的调试对象程序的规定部分中包含的验证值；

判定步骤，对从所述调试对象程序取得的所述验证值与在所述第 1 取得步骤中取得的所述识别符进行比较，按照该比较结果，判定是否允许对所述调试对象程序的调试处理；和

控制步骤，当判定为不允许时，禁止对所述调试对象程序执行调试处理，

所述调试对象程序的所述规定部分中，包含表示对于构成所述调试对象程序的各部分允许或不允许访问的访问控制列表，

所述第 2 取得步骤包含访问控制列表取得步骤，该访问控制列表取得步骤取得所述调试对象程序的所述规定部分中包含的所述访问控制列表，

所述数据处理方法还具备访问判定步骤，该访问判定步骤根据所述取得的所述访问控制列表，判定是否允许访问所述调试对象程序的一部分，

所述控制步骤中，在所述判定步骤判定为允许、且所述访问判定步骤判定为不允许的第 1 情况下，禁止执行所述一部分的调试处理，在所述判定步骤判定为允许、且所述访问判定步骤判定为允许的第 2 情况下，使所述调试处理部执行所述一部分的调试处理。

19. 一种程序生成方法，其特征在于：

包含：

程序取得步骤，取得包含应保密的保护信息的程序；

验证值生成步骤，生成用于按照调试处理部的识别符判定是否允许对所取得的所述程

序的调试处理的验证值；和

保护程序生成步骤，向所述程序附加所述验证值生成步骤中对所述程序生成的验证值，生成保护程序，

由所述程序取得步骤取得的、作为源代码的所述程序中，对应保密的所述保护信息所在的部位预先附加标记，

所述程序生成方法，根据所述标记的有无，将记述有应保密的所述保护信息的区域写入到秘密信息存储文件，

所述程序生成方法还包含访问控制列表取得步骤，该访问控制列表取得步骤取得表示对于构成所述程序的各部分允许或不允许访问的访问控制列表，

所述保护程序生成步骤包含访问控制列表附加步骤，该访问控制列表附加步骤将所述取得的访问控制列表附加于所述程序。

数据处理装置、方法、程序生成装置、方法

技术领域

[0001] 本发明涉及程序的保护，尤其涉及一种控制程序的调试处理执行的技术。

背景技术

[0002] 保护用于著作权管理的程序等、不期望执行非法解析等非法行为的程序（下面称为‘保护程序’）的技术被广泛使用。这是因为若保护不充分，则不限于程序的权利者，在各个方面均会发生损害。

[0003] 例如，若非法者能非法解析对加密后的数字内容执行解密处理后进行再现的程序并篡改该程序，则担心非法利用数字内容。即，非法者非法再现数字内容，或即便限制数字内容的复制次数或再现次数，该限制也被无效化。

[0004] 作为保护程序等数据不被非法者非法解析等的技术，在下述非专利文献 1 中公开了一种 LSI(Large Scale Integration：大规模集成电路）技术，即构筑具有防止从外部非法访问的机构的安全域，具备在安全域中执行处理的安全模式、与通过不使用安全域来执行处理的通常模式，切换通常模式与安全模式来进行动作。根据该技术，通过仅在安全模式时使保护程序动作，可保护该保护程序不被非法解析等。

[0005] 但是，在开发在安全域中动作的程序的情况下，为了执行动作的验证或差错的修正，必需调试 (debug) 上述程序。而且，若无论是谁均能调试上述程序，则非法者会进行调试并插足非法解析等，所以必需将可调试者仅限定为上述程序的开发者等有关人员。

[0006] 因此，以往，使用仅知道规定认证代码的人才能调试在安全域中动作的程序的技术（参照下述的专利文献 1）。具体而言，专利文献 1 的技术中，只有在使用了认证代码的认证中成功的人，才能执行调试处理。

[0007] 由此，由于不知道认证代码的人认证失败，所以可防止非法人员调试安全域中动作的程序。

[0008] 专利文献 1：日本特开 2004-171565 号公报

[0009] 非专利文献 1：TrustZone-Integrated Hardware and Software Security(http://www.arm.com/pdfs/TZ Whitepaper.pdf)

[0010] 根据上述专利文献 1 的技术，在将程序的开发有关人员仅限于例如本公司内的特定组的情况下，只要适当管理认证代码，则可实现程序的保护。

[0011] 另一方面，近年来，为了缓和程序开发的负担，有时多个程序分别由不同的权利者开发，这些程序协同动作。例如在以 SD-Audio 标准将以 OMADRM(Open Mobile Alliance Digital Rights Management：开放移动联盟数字版权管理）标准分发的音乐内容输出到 SD 存储卡 (Secure Digital Memory Card：安全数字存储卡）的情况下，执行基于 OMA DRM 标准的处理的程序与执行基于 SD-Audio 标准的处理的程序分别由不同的权利者开发，这些程序协同动作。

[0012] 开发这些协同动作的多个程序时发生差错，且涉及协同动作的程序的至少一部分是保护程序的情况下，开发者必需同时调试保护程序与其它程序。此时，在这些程序的权利

者就各个程序不同的情况下，存在某个权利者的保护程序由其他权利者一方的开发者来进行调试的需要。

[0013] 但是，上述专利文献 1 的技术使用认证代码来控制是否调试在安全域中动作的程序。知道认证代码的人可调试在安全域中动作的全部程序。

[0014] 此时，上述其它权利者一方的开发者若知道认证代码，则不限于涉及协同动作的程序，所有的保护程序都可以调试。结果，对于上述某个权利者不期望调试的保护程序，若上述其它权利者一方的开发者能获得该保护程序，则也可进行调试。因此，上述某个权利者在共同开发协同动作的程序群的情况下，会承担不期望被调试的其它保护程序被解析等、机密信息也许泄漏到外部等巨大危险。伴随这种危险，在保护程序的同时分别由不同的权利者共同开发协同动作的程序群是困难的。

发明内容

[0015] 因此，本发明的目的在于提供一种在保护程序的同时、使不同的权利者共同开发程序变得容易的数据处理装置、数据处理方法、集成电路、控制调试处理执行的程序、程序生成装置。

[0016] 为了解决上述课题，本发明是一种数据处理装置，控制调试处理部的调试处理的执行，其特征在于：具备：第 1 取得部件，取得识别所述调试处理部的识别符；第 2 取得部件，取得处于被保护不受非法访问的状态下的调试对象程序的规定部分中包含的验证值；判定部件，对从所述调试对象程序取得的所述验证值与由所述第 1 取得部件取得的所述识别符进行比较，按照该比较结果，判定是否允许对所述调试对象程序的调试处理；和控制部件，当判定为不允许时，禁止对所述调试对象程序执行调试处理。

[0017] 发明效果

[0018] 本发明的数据处理装置比较所取得的验证值与调试处理部的识别符，按照比较结果，控制调试处理部可否执行调试处理。即，若程序的权利者使某个验证值包含于该程序中，则允许该程序的调试处理的调试处理部的识别符被决定。

[0019] 因此，该程序的权利者可利用自身包含于该程序中的验证值来指定可执行该程序的调试处理的调试处理部的识别符。

[0020] 即，由于该程序的权利者可如自身希望的那样限定具有可执行该程序的调试处理的识别符的调试处理部，所以可避免让保持不具有该识别符的调试处理部的利害关系人员等不必要地调试该程序。

[0021] 但是，作为程序的权利者，即便允许让其它权利者等调试该程序，也想限制该程序的调试。例如，在该程序中包含想保密的信息的情况下，由于若该秘密部分暴露，则在各个方面发生损害，所以对该秘密部分不希望被解析。

[0022] 因此，也可以是，所述调试对象程序的所述规定部分中，包含表示对于构成所述调试对象程序的各部分允许或不允许访问的访问控制列表，所述第 2 取得部件包含访问控制列表取得部，该访问控制列表取得部取得所述调试对象程序的所述规定部分中包含的所述访问控制列表，所述数据处理装置还具备访问判定部件，该访问判定部件根据所述取得的所述访问控制列表，判定是否允许访问所述调试对象程序的一部分，所述控制部件，在所述判定部件判定为允许、且所述访问判定部件判定为不允许的第 1 情况下，禁止执行所述一

部分的调试处理，在所述判定部件判定为允许、且所述访问判定部件判定为允许的第 2 情况下，使所述调试处理部执行所述一部分的调试处理。

[0023] 由此，程序的权利者可使得该程序中包含想保密的信息的部分不被调试。

[0024] 具体而言，所述访问控制列表所示的所述部分的每一个，表示与将所述调试对象程序加载到存储器时的加载目的地存储器地址的所述部分分别对应的地址范围，在所述访问控制列表中，将与所述部分分别对应的所述地址范围的每一个，与可否访问相对应，所述访问判定部件对于与将所述调试对象程序加载到存储器时的加载目的地存储器地址的所述一部分相对应的地址范围，通过参照在所述访问控制列表中相对应的可否访问，执行所述判定。

[0025] 由此，可限制对配置在特定存储器地址的秘密信息的调试。

[0026] 另外，也可以是，所述访问控制列表中所示的所述部分的每一个，表示所述调试对象程序中包含的符号，所述访问控制列表中，将所述符号的每一个与可否访问相对应，所述访问判定部件对于所述调试对象程序的所述一部分中包含的符号，通过参照在所述访问控制列表中相对应的可否访问，执行所述判定。

[0027] 由此，可限制对特定符号的调试，可通过指定符号这样的简单处理，在符号处理秘密信息的情况下保护秘密信息。

[0028] 但是，有时利害关系人员想灵活地设定程序的保护强度。例如，是想对关联公司具有的调试处理部宽松地允许调试处理的执行，对无关人员的调试处理部，包含秘密信息地限制调试处理的执行的情况等。

[0029] 因此，也可以是，所述调试对象程序的所述规定部分中包含多个所述验证值，所述规定部分中包含一个以上所述访问控制列表，所述访问控制列表的每一个，与所述验证值的至少一个相对应，所述判定部件对于所述验证值的每一个，与所述取得的识别符进行比较，执行所述判定，所述访问判定部件根据与所述判定部件判定为允许的所述验证值相对应的所述访问控制列表，执行所述判定。

[0030] 由此，可对具有分别对应于验证值的识别符的每个调试处理部，指定调试对象程序的可调试部分。即，程序的权利者可按照调试处理部的识别符来设定程序保护的部分。

[0031] 另外，优选的是，所述数据处理装置还具备显示部，所述控制部件包含显示控制部，在所述第 1 情况下，该显示控制部使所述显示部执行表示禁止所述一部分的调试处理之意的显示，在所述第 2 情况下，该显示控制部使所述显示部显示所述调试处理的结果。

[0032] 由此，执行调试的人可知道是否允许调试。

[0033] 在上述数据处理装置中，判定部件的判定方式也可为所述判定部件对所述验证值与所述识别符进行比较，在所述验证值与所述识别符一致的情况下，判定为允许所述调试处理。

[0034] 由此，程序的权利者可利用所述验证值来指定允许执行调试处理的调试处理部。另外，由于通过所述验证值与所述识别符是否一致这样的简单的运算来执行判定，所以可缩短判定所需的时间。

[0035] 另外，作为判定的方式，也可以是，所述判定部件，包含比较值保持部，该比较值保持部以保护其不被非法访问的状态存储比较值，并且执行将所述验证值与所述识别符用作运算符的规定运算，在该运算结果与所述存储的所述比较值一致的情况下，判定为允许所

述调试处理。

[0036] 这样，即便假设所述验证值因某非法部件而被暴露，也由于使用规定的运算与比较值执行所述判定，所以难以从所述验证值得到允许调试处理的调试处理部的识别符。即，根据上述构成，可提高程序的保护强度。

[0037] 该数据处理装置具体地也可通过下述构成实现，即，所述数据处理装置具有具备防止从外部非法访问的机构的安全域，所述数据处理装置具备通常模式与安全模式作为动作模式，切换所述通常模式与所述安全模式而动作，仅在所述安全模式时使用所述安全域而动作，所述数据处理装置还具备切换部，该切换部切换所述通常模式与所述安全模式，在所述通常模式下动作的程序，通过经由所述切换部将规定的处理请求通知给在所述安全模式下动作的程序，从而能访问在所述安全模式下动作的程序，所述调试对象程序存储在所述安全域中，所述第 2 取得部件在所述安全域中，从所述调试对象程序执行所述验证值的所述取得，所述判定部件在所述安全域中执行所述判定。

[0038] 根据上述构成，所述调试对象程序存储在安全域中，另外，所述验证值由所述第 2 取得部件在安全域中取得。

[0039] 因此，所述验证值在数据处理装置取得其的过程中，难以被利害关系人员或非法者等任何人知道。因此，非法者等即便获得程序，也难以确定具有可执行程序的调试处理的识别符的调试处理部，所以可减小非法调试程序的可能性。另外，由于程序中包含所述验证值，所以也不必将用于控制可否调试的信息事先存储在数据处理装置中。

[0040] 在该构成中，也可以是，所述调试处理部在所述安全域之外在所述通常模式下动作，所述数据处理装置还具备在所述安全模式下执行调试处理的安全调试器，所述安全调试器包含于所述安全域中，所述调试处理部输出所述调试对象程序的调试处理请求，所述控制部件，在所述调试处理部输出所述调试处理请求时，使所述判定部件执行所述判定，当判定为不允许时，禁止所述安全调试器对涉及所述调试处理请求的所述调试对象程序进行调试处理。

[0041] 根据上述构成，由于具备在通常模式下动作的调试处理部、和在安全模式下动作的安全调试器，所以即便调试处理部被非法地篡改等，也不影响安全调试器，可防止对在安全模式下动作的程序的非法解析。

[0042] 另外，该构成中，也可以是，所述调试对象程序的所述规定部分中，包含表示对于构成所述调试对象程序的各部分允许或不允许访问的访问控制列表，所述第 2 取得部件包含访问控制列表取得部，该访问控制列表取得部取得所述调试对象程序的所述规定部分中包含的所述访问控制列表，所述数据处理装置还具备访问判定部件，该访问判定部件根据所述取得的所述访问控制列表，判定是否允许访问所述调试对象程序的所述一部分，所述访问判定部件在所述安全域中执行所述判定，所述控制部件，对于所述调试处理部输出的所述调试处理请求所涉及的所述调试对象程序，在所述判定部件判定为允许、且所述访问判定部件判定为不允许的情况下，禁止所述安全调试器执行所述一部分的所述调试处理，在所述判定部件判定为允许、且所述访问判定部件判定为允许的情况下，使所述安全调试器执行所述一部分的调试处理。

[0043] 由此，程序的权利者可使得该程序中包含想保密的信息的部分不被调试。

[0044] 另外，在程序协同动作的情况下，也可如下构成。

[0045] 即，所述调试处理部具有对所述通常程序和与所述通常程序协同的所述调试对象程序执行调试处理的功能，所述调试处理部进行的所述调试对象程序的调试处理，通过所述调试处理部输出调试处理请求、并且所述调试处理部经所述切换部受理由所述安全调试器对所输出的所述调试处理请求执行的调试处理的结果来执行。

[0046] 由此，无论协同动作的程序是在通常模式下动作的程序还是在安全模式下动作的程序，均可由在通常模式下动作的调试处理部同时执行调试处理，所以可高效地开发协同动作的程序。

[0047] 但是，由于不能从在通常模式下动作的程序直接访问在安全模式下动作的程序，所以在安全模式下以通常模式动作的调试处理部不知从何时起在安全模式下调试对象程序动作。因此，对程序的开发者来说，难以调试在安全模式下动作的调试对象程序。

[0048] 因此，也可以是，所述调试处理部输出识别该调试处理部所附着的通常程序的处理识别符，所述安全调试器，将位于与从所述调试处理部输出的所述处理识别符所示的通常程序协同动作的调试对象程序的入口点的命令，变更为断开命令。

[0049] 由此，若调试对象程序开始动作，则利用中断命令暂时中止动作，所以可对程序的开发者执行调试处理执行用的设定等。

[0050] 另外，也可以是，所述数据处理装置，当在所述安全模式下在执行所述调试对象程序过程中检测出调试例外时，经所述切换部向所述调试处理部通知调试例外的发生，所述调试处理部，当从所述切换部受理所述调试例外的发生通知时，输出表示调试处理的执行结果的调试信息的取得请求，所述安全调试器，当所述判定部件对于所述调试对象程序做出肯定的判定时，若受理所述调试信息的取得请求，则执行所述调试对象程序的调试处理，取得调试信息，经所述切换部将取得的调试信息输出到所述调试处理部。

[0051] 由此，程序的开发者若允许调试处理，则可知道对安全模式下动作的程序的调试处理的结果。

[0052] 另外，也可以是，所述数据处理装置还具备：第 1 结果显示部，将所述通常程序的调试处理的结果显示于第 1 显示区域；和第 2 结果显示部，将与所述通常程序的协同所涉及的调试对象程序的调试处理的结果，显示于与所述第 1 显示区域不同的第 2 显示区域，所述第 1 和第 2 结果显示部，在所述协同所涉及的调试对象程序与所述通常程序协同动作时，在所述第 1 显示区域和所述第 2 显示区域中，显示所述调试对象程序与所述通常程序的调试处理的结果。

[0053] 由此，程序的开发者可在确认通常程序与调试对象程序双方的动作的同时，进行调试。

[0054] 另外，也可以是，在所述通常模式下，通常 OS 动作，在所述安全模式下，保护 OS 动作，所述通常程序作为所述通常 OS 生成的处理，在所述通常模式下动作，所述调试处理部作为在所述通常 OS 中动作的调试器，在所述通常模式下动作，所述调试对象程序作为所述保护 OS 生成的处理，在所述安全模式下动作，所述安全调试器被实现为所述保护 OS 具有的功能。

[0055] 另外，也可以是，所述控制部件，当所述调试处理部输出所述调试处理请求时，使所述判定部件执行所述判定，当判定为不允许时，向所述调试处理部输出表示禁止执行所述调试处理的不可调试处理通知。

[0056] 由此，使用调试处理部进行调试的用户，在不允许调试对象程序的调试的情况下，可知道该情况。

[0057] 另外，上述程序可如下生成。

[0058] 即，一种程序生成装置，其特征在于：具备：程序取得部件，取得包含应保密的保护信息的程序；验证值生成部件，生成用于按照调试处理部的识别符判定是否允许对所取得的所述程序的调试处理的验证值；和保护程序生成部件，向所述程序附加所述验证值生成部件对所述程序生成的验证值，生成保护程序。

[0059] 这里，也可以是，所述程序生成装置还包含访问控制列表取得部，该控制列表取得部取得表示对于构成所述程序的各部分允许或不允许访问的访问控制列表，所述保护程序生成部件包含访问控制列表附加部，该访问控制列表附加部将所述取得的访问控制列表附加于所述程序。

[0060] 根据该构成，由于程序的开发者自身使所述验证值包含于程序中，所以可如开发者希望的那样，控制程序的调试处理的执行。

[0061] 另外，本发明是一种数据处理方法，控制调试处理部的调试处理的执行，其特征在于：包含：第 1 取得步骤，取得识别所述调试处理部的识别符；第 2 取得步骤，取得处于被保护不受非法访问的状态下的调试对象程序的规定部分中包含的验证值；判定步骤，对从所述调试对象程序取得的所述验证值与在所述第 1 取得步骤中取得的所述识别符进行比较，按照该比较结果，判定是否允许对所述调试对象程序的调试处理；和控制步骤，当判定为不允许时，禁止对所述调试对象程序执行调试处理。

[0062] 另外，本发明是一种计算机可读取的控制程序，在数据处理装置中执行，该数据处理装置控制调试处理部的调试处理的执行，其特征在于：包含：第 1 取得步骤，取得识别所述调试处理部的识别符；第 2 取得步骤，取得处于被保护不受非法访问的状态下的调试对象程序的规定部分中包含的验证值；判定步骤，对从所述调试对象程序取得的所述验证值与在所述第 1 取得步骤中取得的所述识别符进行比较，按照该比较结果，判定是否允许对所述调试对象程序的调试处理；和控制步骤，当判定为不允许时，禁止对所述调试对象程序执行调试处理。

[0063] 另外，本发明是一种集成电路，用于数据处理装置，该数据处理装置控制调试处理部的调试处理的执行，其特征在于：具备：第 1 取得部，取得识别所述调试处理部的识别符；第 2 取得部，取得处于被保护不受非法访问的状态下的调试对象程序的规定部分中包含的验证值；判定部，对从所述调试对象程序取得的所述验证值与由所述第 1 取得部取得的所述识别符进行比较，按照该比较结果，判定是否允许对所述调试对象程序的调试处理；和控制部，当判定为不允许时，禁止对所述调试对象程序执行调试处理。

[0064] 另外，本发明是一种程序生成方法，其特征在于：包含：程序取得步骤，取得包含应保密的保护信息的程序；验证值生成步骤，生成用于按照调试处理部的识别符判定是否允许对所取得的所述程序的调试处理的验证值；和保护程序生成步骤，向所述程序附加所述验证值生成步骤中对所述程序生成的验证值，生成保护程序。

[0065] 另外，本发明是一种计算机可读取的控制程序，用于使程序生成装置执行生成程序的处理，其特征在于：包含：程序取得步骤，取得包含应保密的保护信息的程序；验证值生成步骤，生成用于按照调试处理部的识别符判定是否允许对所取得的所述程序的调试处

理的验证值；和保护程序生成步骤，向所述程序附加所述验证值生成步骤中对所述程序生成的验证值，生成保护程序。

[0066] 另外，本发明是一种集成电路，用于生成程序的程序生成装置，其特征在于：包含：程序取得部，取得包含应保密的保护信息的程序；验证值生成部，生成用于按照调试处理部的识别符判定是否允许对所取得的所述程序的调试处理的验证值；和保护程序生成部，向所述程序附加所述验证值生成部对所述程序生成的验证值，生成保护程序。

附图说明

[0067] 图 1 是本发明实施方式 1 的数据处理装置 1 的示意图。

[0068] 图 2 是调试功能 7 的详细框图。

[0069] 图 3 是调试器 ID 判定部 22 的详细框图。

[0070] 图 4 是切换设备驱动器 13 的框图。

[0071] 图 5 是表示加密后的保护程序 73 的图。

[0072] 图 6 是表示访问判定部 23 取得的访问控制列表 53 的数据构造一例的图。

[0073] 图 7 是表示调试功能 7 的动作图。

[0074] 图 8 是本发明实施方式 1 的不执行调试时的通常程序 12 和保护程序 8 的执行的流程图。

[0075] 图 9 是表示通常程序 12 需要执行保护程序 8 的功能时的动作的流程图。

[0076] 图 10 是表示通常程序 12a 结束保护程序 8a 的使用时的动作的流程图。

[0077] 图 11 是表示调试器 14 执行对通常程序 12 与保护程序 8 的调试处理用的预处理的流程图。

[0078] 图 12 是在保护程序 8a 执行中发生基于断点的调试例外、并使用调试器 14 对保护程序 8a 执行调试处理时的流程图。

[0079] 图 13 是说明本发明的保护程序 8 的生成方法用的图。

[0080] 图 14 是保护程序生成装置 72 的构成图。

[0081] 图 15 是表示保护程序生成装置 72 生成加密化的保护程序 73 的处理的流程图。

[0082] 图 16 是说明调试器 ID 管理服务器执行的调试器 ID 的管理方法的图。

[0083] 图 17 是表示调试器 ID 管理服务器 81 用于调试器 ID 管理的调试器 ID 管理文件 90 的数据构造的图。

[0084] 图 18 是表示用于显示程序的动作信息的图形用户界面 (GUI) 的图。

[0085] 图 19 是本发明实施方式 5 的基于字符的用户界面 (CUI) 的显示方法的说明图。

[0086] 符号说明

[0087] 1 数据处理装置

[0088] 2 LSI

[0089] 3 切换机构

[0090] 6 保护 OS

[0091] 7 调试功能

[0092] 8 保护程序

[0093] 11 通常 OS

[0094] 12 通常程序

[0095] 13 切换设备驱动器

[0096] 14 调试器

[0097] 15 调试器用切换设备驱动器

[0098] 21 控制部

[0099] 22 调试器 ID 判定部

[0100] 23 访问判定部

[0101] 24 安全调试器

[0102] 25 调试信息取得部

[0103] 26 断点设定部

[0104] 27 寄存器值取得设定部

[0105] 28 存储器值取得设定部

[0106] 31 调试器 ID 比较部

[0107] 32 调试器 ID 运算部

[0108] 33 比较值保持部

[0109] 41 切换操作部

[0110] 42 请求分配部

[0111] 43 通常请求受理部

[0112] 44 调试请求受理部

[0113] 51 保护程序主体

[0114] 52 允许调试器 ID 信息

[0115] 53 访问控制列表

[0116] 54 解密用头信息

[0117] 60 访问控制列表

[0118] 61 开始地址

[0119] 62 结束地址

[0120] 63 访问允许信息

[0121] 64 符号名

[0122] 65 访问允许信息

[0123] 71 保护程序源代码

[0124] 72 保护程序生成装置

[0125] 73 加密的保护程序

[0126] 74 允许调试器 ID 存储文件

[0127] 75 访问控制列表存储文件

[0128] 76 秘密信息区域存储文件

[0129] 77 编译器

[0130] 78 连接程序

[0131] 79 保护程序化工具

[0132] 81 调试器 ID 管理服务器

[0133] 82 保护程序开发装置

[0134] 83 保护程序解析装置

[0135] 90 调试器 ID 管理文件

[0136] 91 管理序号

[0137] 92 调试器 ID

[0138] 93 保护程序的开发者名

[0139] 94 联络地址

[0140] 150 GUI

[0141] 151 代码显示部

[0142] 152 寄存器显示部

[0143] 153 存储器显示部

[0144] 154 符号显示部

[0145] 155 观察点显示部

[0146] 156 调用堆栈显示部

[0147] 157 窗口标题显示部

[0148] 158 菜单显示部

[0149] 159 模式显示部

[0150] 160 通常程序用调试窗口

[0151] 161 保护程序用调试窗口

[0152] 170 CUI

[0153] 171 调试处理结果的显示例

具体实施方式

[0154] 下面，参照附图来说明本发明的一实施方式。

[0155] 1 实施方式 1

[0156] 图 1 是本发明实施方式 1 的数据处理装置 1 的示意图。数据处理装置 1 由具有保护机构的 LSI2 与切换机构 3、保护 OS6、调试功能 7、保护程序 8(8a、8b、..)、通常 OS11、通常程序 12(12a、12b、..)、切换设备驱动器 13(下面称为‘切换设备驱动器 13’)、调试器 14、调试器用切换设备驱动器 15(下面称为‘调试器用切换设备驱动器 15’)构成。

[0157] 1.1 实施方式 1 中的各功能块的说明

[0158] 1.1.1 LSI2

[0159] 图 1 中，LSI2 搭载用于保护程序不被非法解析或篡改的机构、即保护机构。保护机构具有防止从外部非法访问的硬件机构。作为保护机构的具体实例，例如暂时截断从外部访问等。

[0160] LSI2 具备保护模式(或也可称为‘安全模式’)与通常模式，作为动作模式，切换保护模式与通常模式来动作。动作模式的切换使用后述的切换机构 3 来执行。

[0161] 所谓保护模式是由保护机构保护程序不被非法解析或篡改的特殊模式，保护 OS6 或保护程序 8 动作。另一方面，所谓通常模式是不由保护机构保护程序的一般模式，通常 OS11 或通常程序 12 动作。

[0162] 保护 OS6 使用切换机构 3 来执行从保护模式到通常模式的切换。位于通常 OS11 中的切换设备驱动器 13 等使用切换机构 3 来执行从通常模式到保护模式的切换。

[0163] 1.1.2 切换机构 3

[0164] 切换机构 3 具有受理来自保护 OS6 或通常 OS11 的动作模式切换指示、执行动作模式切换所需的处理用的硬件机构。动作模式的切换处理例如可适用非专利文献 1 中记载的技术。

[0165] 此时，切换机构 3 无论在通常模式还是在保护模式下均动作，具有通常模式和保护模式均可访问的存储区域。在通常模式下 LSI2 动作的情况下，使从通常模式下动作的通常程序 12 向保护模式下动作的保护程序 8 的请求等暂时存储在上述存储区域中。在切换了动作模式之后，保护 OS6 或保护程序 8 读出所述存储的信息，由此实现通常模式下动作的程序与保护模式下动作的程序间的通信。

[0166] 1.1.3 保护 OS6

[0167] 保护 OS6 是 LSI2 为保护模式时的、控制数据处理装置 1 的动作的 OS。

[0168] 保护 OS6 执行在保护模式下动作的保护程序 8 的管理（处理管理）或资源管理、使用存储器管理单元 (MMU) 的保护程序间的访问控制、中断处理、使用切换机构 3 的向通常模式的切换处理、为了调试保护程序而使用调试功能 7 的调试处理等。

[0169] 1.1.4 调试功能 7

[0170] 调试功能 7 在调试器 14 执行对保护程序 8 的调试处理时，控制该调试处理的执行。

[0171] 即，当调试器 4 调试保护程序 8 时，调试功能 7 判定是否允许调试器 14 调试保护程序 8。在判定的结果为允许的情况下，根据来自调试器 14 的请求，对保护程序 8 执行调试信息的取得或断点 (breakpoint) 的设定、寄存器值或存储器值的取得、设定等处理。所谓调试信息是程序调试用的信息，表示目标文件中的程序代码与源代码的对应关系等。另外，调试功能 7 使用停止标志 (flag)，执行保护程序 8 的调试处理的预处理。调试功能 7 的细节如后所述。

[0172] 1.1.5 保护程序 8

[0173] 保护程序 8 是包含必需保护不被非法解析或篡改的信息（下面称为秘密信息）的应用程序。

[0174] 作为秘密信息的实例，有用于对加密的数字内容进行解密的解密密钥或解密算法、存储涉及再现或拷贝的权利的权利信息等。另外，为了防止非法解析，保护程序 8 在开始执行之前以加密的状态被保持，在开始执行时由保护 OS6 解密。

[0175] 1.1.5.1 保护程序 8 的补充

[0176] 保护程序 8 在开始执行之前，如图 5 所示的加密的保护程序 73 那样，以加密的状态被保持。

[0177] 如图 5 所示，加密的保护程序 73 由保护程序主体 51、允许调试器 ID 信息 52、访问控制列表 53、解密用头信息 54 构成。当使用解密用头信息 54 对加密的保护程序 73 进行解密时，得到保护程序 8。保护程序 8 由保护程序主体 51、允许调试器 ID 信息 52、和访问控制列表 53 构成。

[0178] 1.1.5.2 保护程序主体 51

[0179] 保护程序主体 51 是程序的执行代码。

[0180] 1.1.5.3 允许调试器 ID 信息 52

[0181] 允许调试器 ID 信息 52 是用于判定是否允许对保护程序 8 的调试处理的验证值。在本实施方式中，设所谓允许调试器 ID 信息 52 表示允许对保护程序 8 的调试处理的调试器的识别符（调试器 ID）。即，具有与允许调试器 ID 信息 52 所示的值相同的调试器 ID 的调试器，能够执行对保护程序 8 的调试处理。当调试功能 7 执行上述判定时，使用该允许调试器 ID 信息 52。

[0182] 1.1.5.4 访问控制列表 53

[0183] 访问控制列表 53 是表示是否允许对保护程序 8 的规定区域进行访问的列表。所谓访问控制列表 53 主要是将构成保护程序 8 的各部分分别与是否允许访问相对应的列表。访问控制列表 53 的细节如后所述。

[0184] 1.1.5.5 解密用头信息 54

[0185] 解密用头信息 54 表示加密的保护程序 73 的解密所需的信息。例如，解密用头信息 54 中包含加密中所用的算法、或将保护程序 8 加载到存储器的地址等。在加密的算法中，附加解密所需的信息并执行程序解密的技术以往已被公知，不是本发明的主要构成要件，所以省略详细说明。

[0186] 1.1.5.6 各数据的配置

[0187] 保护程序 8 中、允许调试 ID 信息 52 或访问控制列表 53 或保护程序主体 51 可任意配置。具体而言，也可将如何配置允许调试器 ID 信息 52 等信息，作为头信息等信息，附加于程序。

[0188] 另外，也可事先定义包含于保护程序的哪个部分中，根据该定义，数据处理装置 1 的调试功能 7 读出允许调试器 ID 信息 52。例如，事先定义表示允许调试 ID 信息 52 的比特数（或字节数）、与表示访问控制列表 53 的比特数，将从保护程序 8 的开头起、到规定比特为止，设为允许调试 ID 信息 52，将从规定比特其到后续的规定比特为止，设为访问控制列表 53。

[0189] 在存在多个由允许调试器 ID 信息 52 与访问控制列表 53 构成的组的情况下，也可在保护程序的开头包含表示有几个这些组的信息。

[0190] 另外，也可在解密用头信息 54 中包含在保护程序 8 内如何配置这些允许调试器 ID 信息 52 等。调试功能 7 通过读取表示这些允许调试器 ID 信息 52 等在保护程序 8 中所占位置的信息等，可取得允许调试器 ID 信息 52 或访问控制列表 53。

[0191] 保护程序 8 的细节在实施方式 2 中与其生成方法一起详细说明。

[0192] 1.1.6 通常 OS11

[0193] 通常 OS11 是 LSI2 为通常模式时的、控制数据处理装置 1 的动作的 OS。

[0194] 即，通常 OS11 在通常模式下动作时，执行在通常模式下动作的通常程序 12 的管理（处理管理）或资源管理、中断处理等。

[0195] 1.1.7 切换设备驱动器 13

[0196] 切换设备驱动器 13 作为通常 OS11 的设备驱动器动作，当通常程序 12 执行与保护程序 8 的通信时使用。虽然如后所述，但在调试器 14 执行与调试功能 7 的通信时，利用调试器用切换设备驱动器 15。

[0197] 具体而言，切换设备驱动器 13 执行通常程序 12 与保护程序 8 之间的通信数据的传递处理、和从通常模式向保护模式的切换处理。所谓通信数据的传递处理主要是受理通常程序 12 输出的数据，经由切换机构 3，输出到保护程序 8，以及，经由切换机构 3，取得从保护程序 8 输出的数据，将取得的数据输出到通常程序 12 的处理。

[0198] 切换设备驱动器 13 的细节如后所述。

[0199] 1.1.8 调试器用切换设备驱动器 15

[0200] 调试器用切换设备驱动器 15 作为通常 OS11 的设备驱动器动作，当调试器 14 执行与调试功能 7 的通信时使用。

[0201] 调试器用切换设备驱动器 15 执行调试器 14 与调试功能 7 的通信数据的传递处理、和从通常模式向保护模式的切换处理。

[0202] 1.1.9 通常程序 12

[0203] 通常程序 12(12a、12b、..) 是在通常 OS11 上动作的应用程序。

[0204] 通常程序 12 利用切换设备驱动器 13 与在保护模式下动作的保护程序 8 执行通信，与保护程序 8 协同动作。

[0205] 1.1.10 调试器 14

[0206] 调试器 14 具有对通常程序 12 执行调试处理的功能、和对保护程序 8 执行调试处理的功能。调试器 14 具有作为用于识别自身的识别符的调试器 ID。当调试功能 7 判定是否允许调试时，使用该调试器 ID。另外，调试器 14 的调试器 ID 的管理等在实施方式 3 中详细说明。

[0207] 调试器 14 调试通常程序 12 的功能例如通过与 Linux（注册商标）中使用的 GDB 等应用程序调试器一样的功能来实现。

[0208] 另外，所谓对保护程序 8 执行调试处理的功能是如下功能，即，调试器 14 经调试器用切换设备驱动器 15 与保护 OS6 的调试功能 7 通信，调试功能 7 对保护程序 8 执行调试信息的取得或断点的设定、寄存器值或存储器值的取得、设定等调试处理，受理该调试处理的结果。

[0209] 在下面的说明中，调试器 14 与通常模式下动作的通常程序 12 连接 (attach)，对所连接的通常程序（例如通常程序 12a) 和与该通常程序协同动作的保护程序（例如保护程序 8a) 执行调试处理。

[0210] 本实施方式 1 的调试器 14 为在通常 OS11 上动作的应用程序调试器，但不限于此，例如也可以是在 Linux（注册商标）中使用的 KGDB 等内核模式调试器，可执行通常模式或保护模式下动作的设备驱动器的调试。

[0211] 1.2 调试功能 7 的详细说明

[0212] 下面，说明‘1.1.4 调试功能 7’中说明的调试功能 7 的细节。

[0213] 图 2 是调试功能 7 的详细框图。调试功能 7 具有控制部 21、调试器 ID 判定部 22、访问判定部 23、安全调试器 24。安全调试器 24 具有调试信息取得部 25、断点设定部 26、寄存器值取得设定部 27、存储器值取得设定部 28。

[0214] 如‘1.1.5 保护程序 8’中所述，保护程序 8 中包含允许调试器 ID 信息 52 与访问控制列表 53，如‘1.1.10 调试器 14’中所述，调试器 14 具有调试器 ID。

[0215] 在下面的调试功能 7 的说明中，对于调试对象的保护程序，不将哪个程序特定为

调试对象，统称为调试对象的保护程序 8 来进行说明。

[0216] 1.2.1 调试器 ID 判定部 22

[0217] 图 2 中，调试器 ID 判定部 22 判定是否允许调试器 14 对调试对象的保护程序 8 执行调试处理。

[0218] 即，调试器 ID 判定部 22 取得调试器 14 具有的调试器 ID、和调试对象的保护程序 8 中包含的允许调试器 ID 信息 52。比较取得的调试器 ID 与允许调试器 ID 信息 52。按照该比较结果，执行是否允许调试器 14 执行调试对象的保护程序 8 的调试（即由调试器 14 对调试对象的保护程序执行调试处理）的判定。

[0219] 1.2.1.1 调试器 ID 判定部 22 的详细说明

[0220] 图 3 是调试器 ID 判定部 22 的详细框图。

[0221] 如图 3 所示，调试器 ID 判定部 22 具有调试器 ID 比较部 31、与调试器 ID 运算部 32、比较值保持部 33。调试器 ID 判定部 22 判定调试器 14 的调试器 ID 与调试对象的保护程序 8 中包含的允许调试器 ID 信息 52 所示的值是否一致。

[0222] 具体而言，如图 3 所示，调试器 ID 运算部 32 受理调试器 14 的调试器 ID 与调试对象的保护程序 8 中包含的允许调试器 ID 信息 52。从调试器 ID 中减去允许调试器 ID 信息 52 所示的值。将减法的结果作为运算结果，输出到调试器 ID 比较部 31。

[0223] 调试器 ID 比较部 31 比较调试器 ID 运算部 32 的运算结果与比较值保持部 33 保持的比较值，在一致的情况下，将‘可调试’通知给控制部 21，在不一致的情况下，将‘不可调试’通知给控制部 21。比较值保持部 33 保持“0”，作为用于与调试器 ID 运算部 32 的运算结果比较的比较值。

[0224] 1.2.1.2 调试器 ID 判定部 22 的补充说明

[0225] 本实施方式 1 的调试器 ID 判定部 22 判定调试器 14 的调试器 ID 与调试对象的保护程序 8 中包含的允许调试器 ID 信息 52 所示的值是否一致，但不限于判定调试器 ID 的一致。即，调试器 ID 运算部 32 除减法之外，也可执行乘法或加密解密运算。另外，比较值保持部 33 也可保持“0”以外的值。

[0226] 主要是，调试器 ID 判定部 22 将调试器 14 的识别符与调试对象的保护程序 8 中包含的验证值作为运算符，执行规定的运算，若其结果与比较值保持部 33 保持的比较值一致，则判定为‘可调试’，若不一致，则判定为‘不可调试’。

[0227] 另外，也可以是在未图示的调试器 ID 保持部中事先保持保护程序 8 的允许调试器 ID 信息 52 所示的值，调试器 ID 判定部 22 比较从调试器 14 受理到的调试器 14 的调试器 ID 与调试器 ID 保持部中保持的值。此时，调试器 ID 运算部 32 不特别执行运算。

[0228] 1.2.2 访问判定部 23

[0229] 返回图 2 继续说明。

[0230] 访问判定部 23 在调试器 14 对调试对象的保护程序 8 的规定区域请求访问的情况下，判定是否允许该访问。

[0231] 即，访问判定部 23 从调试对象的保护程序 8 取得访问控制列表 53，根据取得的访问控制列表 53，判定是否允许调试器 14 访问要访问的区域。

[0232] 1.2.2.1 访问控制列表 53 的详细说明

[0233] 这里，说明访问控制列表 53 的细节。

[0234] 图 6 是表示访问判定部 23 取得的访问控制列表 53 的数据构造一例的图。

[0235] 访问控制列表 53 由执行访问控制的区域、和涉及该区域的访问允许信息两个部分构成。在下面的说明中，说明以存储器地址执行访问控制时的访问控制列表 53a、与利用符号执行访问控制时的访问控制列表 53b。所谓符号 (symbol) 是识别程序中包含的变量或函数等的识别符。

[0236] 1.2.2.2 由存储器地址执行访问控制的情况

[0237] 图 6(a) 是由存储器地址指定执行访问控制的区域时的访问控制列表 53a 的数据构造，执行访问控制的区域由开始地址与结束地址指定。

[0238] 如图 6(a) 所示，访问控制列表 53a 的 1 条记录包含开始地址 61a、结束地址 62a 与访问允许信息 63a。

[0239] 开始地址 61a 与结束地址 62a 表示执行访问控制的存储器区域的开始地址与结束地址。

[0240] 访问允许信息 63a 表示是否允许对访问开始地址 61a 与结束地址 62a 所示的存储器区域进行访问。在允许的情况下，例如以 1 比特的信息表示‘可访问’，在不允许的情况下，例如以 1 比特的信息表示‘不可访问’。访问控制列表 53a 中包含多个由开始地址 61a 和结束地址 62a 所表示的存储器区域的组，这些组的每一个中对应存储是否允许访问。

[0241] 在列表的开头，将未由开始地址 61a 与结束地址 62a 所示的区域定义为‘default’(缺省)。在本实施方式中，设对‘default’区域的访问为‘不可访问’。

[0242] 另外，开始地址 61a 等所示的地址在本实施方式中是相对地址。即，保护程序 8 的解密用头信息 54 中示出将保护程序 8 加载到存储器时的存储器地址，开始地址 61a 等示出将该存储器地址的开头设为 0 的相对地址。当然，开始地址 61a 等所示的地址也可以是存储器的绝对地址。

[0243] 1.2.2.3 由存储器地址执行访问控制时的动作

[0244] 访问判定部 23 取得访问控制列表 53a 与保护程序 8a 的调试信息。另外，使用保护程序 8a 的调试信息，将调试器 14 请求访问的符号变换为地址。

[0245] 访问判定部 23 从列表的上方开始，顺序判定变换后的地址是否包含于访问控制列表 53a 中的开始地址 61a 和结束地址 62a 所示的存储器区域的每个中。若判定为包含，则取得与该区域对应的访问允许信息 63a，将访问允许信息 63a 所示的信息、即‘可访问’或‘不可访问’通知给控制部 21。另外，在不包含于列表中的存储器区域中的情况下，根据与列表开头的‘default’对应的访问允许信息 63a，将‘可访问’或‘不可访问’通知给控制部 21。

[0246] 1.2.2.4 由符号执行访问控制的情况

[0247] 图 6(b) 是由符号名指定执行访问控制的区域时的访问控制列表 53b 的数据构造，执行访问控制的区域由符号名指定。

[0248] 如图 6(b) 所示，访问控制列表 53b 的 1 条记录包含符号名 64b 与访问允许信息 65b。

[0249] 符号名 64b 表示构成访问控制对象的符号的名称。

[0250] 访问允许信息 65b 表示是否允许访问符号名 64b 所示的符号。

[0251] 如图 6(b) 所示，在访问控制列表 53b 中，对每个符号示出是否允许访问。

[0252] 在列表的开头，将未由符号名 64b 所示的符号定义为‘default’。在本实施方式中，设对‘default’符号的访问为‘不可访问’。

[0253] 1.2.2.5 由符号执行访问控制时的动作

[0254] 访问判定部 23 取得访问控制列表 53b。从列表的上方开始顺序判定调试器 14 请求访问的符号名称是否与访问控制列表 53b 中的符号名 64b 所示的符号名一致。在与列表中的符号名一致的情况下，取得该符号名所对应的访问允许信息 65b，将访问允许信息 65b 所示的信息、即‘可访问’或‘不可访问’通知给控制部 21。在与列表中的符号名不一致的情况下，根据与列表开头的‘default’对应的访问允许信息 65b，将‘可访问’或‘不可访问’通知给控制部 21。

[0255] 1.2.2.6 补充说明

[0256] 本实施方式 1 的访问控制列表 53a、53b 在其列表的开头，将未由访问控制列表 53a、53b 所示的符号作为‘default’，对应于‘default’存储访问允许信息 63a、63b，但不限于此。

[0257] 即，在访问判定部 23 中，也可在访问控制列表 53a、53b 中未包含的存储器区域或符号名的情况下，始终判定为‘可访问’，或相反始终判定为‘不可访问’。

[0258] 另外，调试器 14 利用符号来请求访问，但不限于符号。例如，调试器 14 也可利用存储器的地址来指定访问的区域。

[0259] 此时，在上述实例中使用了访问控制列表 53a 的访问允许判定中，执行将从调试器 14 指定的符号暂时变换为地址的处理，但也可利用从调试器 14 指定的地址直接判定。

[0260] 1.2.3 安全调试器 24

[0261] 安全调试器 24 对应于来自调试器 14 的委托，执行各种调试处理。

[0262] 安全调试器 24 包含调试信息取得部 25、断点设定部 26、寄存器值取得设定部 27、存储器值取得设定部 28。

[0263] 作为调试处理，由调试信息取得部 25 执行从调试对象的保护程序 8 取得符号信息等调试信息的处理。

[0264] 另外，由断点设定部 26 执行在调试对象的保护程序 8 中设定断点的处理。

[0265] 由寄存器值取得设定部 27 取得调试对象的保护程序 8 正使用的寄存器值，或设定调试对象的保护程序 8 使用的寄存器值。

[0266] 由存储器值取得设定部 28 取得调试对象的保护程序 8 正使用的存储器值，或设定调试对象的保护程序 8 使用的存储器值。

[0267] 1.2.4 控制部 21

[0268] 控制部 21 根据调试器 ID 判定部 22 与访问判定部 23 的判定结果，确认是否允许调试器 14 对调试对象的保护程序 8 执行调试处理。

[0269] 若确认的结果是允许执行调试处理，则控制部 21 对应于来自调试器 14 的委托，调用安全调试器 24 中包含的各处理（调试信息取得部 25、断点设定部 26、寄存器值取得设定部 27、存储器值取得设定部 28）。

[0270] 控制部 21 在调试器 ID 判定部 22 判定为不允许调试器 14 调试处理调试对象的保护程序 8 的情况下、或由访问判定部 23 判定为对禁止调试器 14 访问的区域进行访问的情况下，不处理从调试器 14 委托的请求。即，不调用安全调试器 24 中包含的各处理。此时，

为了将不调用安全调试器 24 中包含的各处理这一情况通知给调试器 14，也可向调试器 14 输出表示不允许执行调试处理的不可调试处理通知。由此，调试器 14 可执行向调试器 14 的用户示出不允许执行调试处理等的处理。

[0271] 如上所述，说明了使用允许调试器 ID 信息 52 与访问控制列表 53 来进行调试处理的执行的控制，但保护程序 8 中包含的允许调试器 ID 信息 52 所示的调试器 ID 可以是一个，也可以是多个。通过包含多个调试器 ID，可允许多个开发者进行调试处理。例如，考虑多个开发者共同开发程序的情况等。

[0272] 另外，保护程序 8 中也可包含将允许调试器 ID 信息 52 与访问控制列表 53 对应的多个组。

[0273] 不用说，也可包含多个含有 1 个调试器 ID 的允许调试器 ID 信息 52，使访问控制列表 53 对应于各个允许调试器 ID 信息 52。此时，访问控制列表 53 表示对各个允许调试器 ID 信息 52 的每个不同的访问限制即可。由此，可对每个调试器 ID，施加不同的访问限制。

[0274] 控制部 21 使调试器 ID 判定部 22 对所保持的多个允许调试器 ID 信息 52 所示的调试器 ID 的每个执行判定，若存在判定为调试允许的调试器 ID，则委托访问判定部 23 进行基于该调试器 ID 所对应的访问控制列表的访问判定。在将全部的调试器 ID 判定为不允许调试的情况下，不处理从调试器 14 委托的请求。

[0275] 1.3 切换设备驱动器 13 的详细说明

[0276] 下面，说明‘1.1.7 切换设备驱动器 13’中说明的切换设备驱动器 13 的细节。

[0277] 图 4 是切换设备驱动器 13 的框图。切换设备驱动器 13 具有切换操作部 41、请求分配部 42、通常请求受理部 43、调试请求受理部 44。

[0278] 1.3.1 切换操作部 41

[0279] 切换操作部 41 通过保存通常模式下使用的寄存器值等，保存通常模式下的数据处理装置的状态，之后，使用切换机构 3，执行从通常模式切换到保护模式的处理。

[0280] 并且，当从保护模式切换到通常模式时，执行所保存的状态的恢复处理，将来自切换时发生的保护模式的请求通知给请求分配部 42。作为该请求，有基于调试例外的调试请求、与对通常程序 12 的请求。

[0281] 1.3.2 请求分配部 42

[0282] 请求分配部 42 判断来自保护模式的请求是基于保护模式动作中发生的调试例外的调试请求、还是对通常程序 12 的请求。

[0283] 在判断的结果是调试请求的情况下，向调试请求受理部 44 通知调试请求，在是对通常程序 12 的请求的情况下，向通常请求受理部 43 通知来自保护模式的请求。

[0284] 1.3.3 通常请求受理部 43

[0285] 通常请求受理部 43 对保护 OS6 或保护程序 8 等保护模式下动作的程序与通常程序 12 之间的通信，进行中介。

[0286] 即，将来自保护模式下动作的程序的请求通知给通常程序 12，或相反，将来自通常程序 12 的请求通知给保护模式下动作的程序。

[0287] 1.3.4 调试请求受理部 44

[0288] 调试请求受理部 44 将因保护程序 8 中设定的断点而发生的调试例外，通知给正在调试与保护程序 8 协同动作中的通常程序 12 的调试器 14。

[0289] 通过如上所述构成切换设备驱动器13，可在对通常程序12与保护程序8之间的通信进行中介的同时，将保护程序8发生的调试例外通知给适当的调试器14。由此，可防止将保护程序8中包含的秘密信息通知给无关的调试器而泄漏秘密信息。

[0290] 1.3.5 切换设备驱动器13的补充

[0291] 本实施方式1的切换设备驱动器13，除切换操作部41以外，由请求分配部42及通常请求受理部43、调试请求受理部44构成，但不限于这种构成。例如，也可以是切换设备驱动器13仅由切换操作部41构成，将请求分配部42及通常请求受理部43、调试请求受理部44作为库(library)包含在通常程序12中。此时，请求分配部42及通常请求受理部43、调试请求受理部44的动作变为执行在通常程序12的执行中调用的库。

[0292] 1.4 动作

[0293] 下面，说明数据处理装置1的动作。

[0294] 在下面的说明中，首先，在‘1.4.1 调试功能7的动作’中说明也是本发明特征的调试功能7的动作。

[0295] 接着，作为整体动作，在‘1.4.2 不执行调试处理时的动作’中说明不执行调试处理时的动作。即，说明通常程序12与保护程序8的协同动作。

[0296] 之后，在‘1.4.3 调试处理的预处理’‘1.4.4 调试处理’中，将调试器14对通常程序12与保护程序8执行调试处理时的动作、与其预处理一起加以说明。

[0297] 由于保护程序8是在保护模式下动作的程序，所以在通常模式下动作的程序和调试器对保护程序8不能直接进行访问。因此，安全调试器24执行对保护程序8的调试处理，调试器14受理其结果。

[0298] 1.4.1 调试功能7的动作

[0299] 图7是表示调试功能7的动作图。

[0300] 调试器ID判定部22根据调试器14的调试器ID与保护程序8的允许调试器ID信息52，判定是否允许调试器14调试保护程序8(S101)。

[0301] 控制部21根据调试器ID判定部22的判定结果，切换处理(S102)。即，在调试器ID判定部22的判定结果为‘不可调试’的情况下(S102：否)，中止调试处理。

[0302] 在调试器ID判定部22的判定结果为‘可调试’的情况下(S102：是)，访问判定部23根据保护程序8的访问控制列表53，判定是否允许调试器14访问调试器14要访问的保护程序8的区域(S103)。

[0303] 控制部21根据访问判定部23的判定结果，切换处理(S104)。即，在访问判定部23的判定结果为‘不可访问’的情况下(S104：否)，中止调试处理。

[0304] 在访问判定部23的判定结果为‘可访问’的情况下(S104：是)，调试功能7调用安全调试器24的各处理部(调试信息取得部25、断点设定部26、寄存器值取得设定部27、存储器值取得设定部28)，执行处理(S105)。

[0305] 1.4.2 不执行调试处理时的动作

[0306] 图8是本发明实施方式1的不执行调试时的通常程序12和保护程序8的执行的流程图。

[0307] 1.4.2.1 保护程序8的加载处理

[0308] 首先，说明保护程序8的加载处理。下面，设通常程序12与保护程序8协同动作。

通常程序 12 经由作为通常 OS11 的设备驱动器的切换设备驱动器 13 与保护 OS3，调用保护程序 8。在下面的说明中，以通常程序 12a 与保护程序 8a 协同动作的情况为例进行说明。通常程序 12b 动作的情况或保护程序 8b 协同动作的情况也一样。

[0309] 如图 8 所示，首先，启动通常程序 12a。启动的通常程序 12a 作为用于将动作模式切换到保护模式的预处理，对切换设备驱动器 13 进行 open（打开）（步骤 S201）。所谓 open 是指切换设备驱动器 13 处于可与保护 OS6 等保护模式下动作的处理进行通信的状态。

[0310] 通常程序 12a 为了使保护程序 8a 动作，指定加密的保护程序，经由切换设备驱动器 13，将使涉及该指定的加密后的保护程序加载到存储器的请求，通知给保护 OS6（步骤 S202）。这里，若对加密过的保护程序进行解密，则生成保护程序 8a。

[0311] 保护 OS6 受理加密的保护程序的加载请求，从加密的保护程序的解密用头信息中，取得加载所需的信息（步骤 S203）。在加载所需的信息中，包含加密的保护程序中包含的保护程序主体的加载目的地地址等、加密的保护程序解密所需的信息等。

[0312] 保护 OS6 根据涉及取得的、加载所需的信息，对加密的保护程序进行解密。将利用解密得到的保护程序 8a 加载到以保护模式管理着的存储器区域（步骤 S204），变为可执行保护程序 8a 的状态。

[0313] 若变为可执行保护程序 8a 的状态，则经由切换设备驱动器 13，从保护 OS6 返回到通常程序 12a（步骤 S205）。即，向通常程序 12a 移交处理的执行权，重新开始执行通常程序 12a。

[0314] 1.4.2.2 保护程序 8 的执行

[0315] 接着，说明当通常 OS11 中执行通常程序 12a 时，通常程序 12a 需要执行保护程序 8a 具有的功能时的处理。

[0316] 图 9 是表示通常程序 12 需要执行保护程序 8 的功能时的动作的流程图。

[0317] 在图 9 所示的处理说明中，设已执行上述图 8 所示的处理，处于可执行保护程序 8 的状态。

[0318] 通常程序 12a 经由切换设备驱动器 13，将保护程序 8a 的执行的请求通知给保护 OS6（步骤 S206）。上述执行的请求中示出保护程序 8a 应执行的命令或处理。

[0319] 保护 OS6 受理保护程序 8a 的执行的请求，执行保护程序 8a，执行对应于所述执行请求的处理（步骤 S207）。

[0320] 若保护程序 8a 结束处理，则经由保护 OS6、切换设备驱动器 13，从保护程序 8a 返回到通常程序 12a（步骤 S208）。若通常程序 12a 利用保护程序 8a 的处理结果，则经由切换机构 3 或切换设备驱动器 13，在保护程序 8a 与通常程序 12a 之间，执行处理结果的传递。

[0321] 每当需要执行保护程序 8a 具有的功能时，就执行从上述步骤 S206 至步骤 S208 的处理。

[0322] 1.4.2.3 保护程序 8 的使用的结束

[0323] 下面，说明通常程序 12a 结束保护程序 8a 的使用时的动作。

[0324] 图 10 是表示通常程序 12a 结束保护程序 8a 的使用时的动作的流程图。

[0325] 如图 10 所示，若通常程序 12a 结束保护程序 8a 的使用，则通常程序 12a 经由切换设备驱动器 13，将保护程序 8a 的删除请求输出到保护 OS6（步骤 S209）。删除请求中示出构成删除对象的保护程序 8。在本实施方式中，利用删除请求删除保护程序 8a。

[0326] 保护 OS6 受理删除请求，删除保护程序 8a（步骤 S210）。之后，经由切换设备驱动器 13，从保护 OS6 返回到通常程序 12a。通过该删除操作，在再次加载保护程序 8a 之前，不能使用保护程序 8a 的功能。另外，由于作为明文状态的保护程序 8a 从存储器中消失，所以使之难以受到非法解析。

[0327] 之后，通常程序 12a 中，若不必切换到保护模式，则对切换设备驱动器 13 进行 close（关闭）(S211)。所谓 close 是切换设备驱动器 13 不执行与保护 OS6 等的通信的状态。

[0328] 1.4.3 调试处理的预处理

[0329] 下面，说明调试器 14 对通常程序 12 与保护程序 8 执行调试处理时的动作。首先，说明调试处理的预处理。

[0330] 1.4.3.1 预处理

[0331] 图 11 是表示调试器 14 执行对通常程序 12 与保护程序 8 的调试处理用的预处理的流程图。

[0332] 在下面的说明中，设通常程序 12a 与保护程序 8a 协同动作，调试器 14 对通常程序 12a 与保护程序 8a 执行调试处理。

[0333] 调试器 14 受理程序开发者的附着操作，为了对通常程序 12a 执行调试处理，附着于通常程序 12a 上 (S301)。

[0334] 调试器 14 为了与调试处理执行时以保护模式动作的调试功能 7 通信，对调试器用切换设备驱动器 15 进行 open(S302)。

[0335] 调试器 14 为了向调试功能 7 通知通常程序 12a 的处理 ID，经由调试器用切换设备驱动器 15，将通常程序 12a 的处理 ID 通知给以保护模式动作的调试功能 7(S303)。

[0336] 调试功能 7 保存所通知的处理 ID，当执行保护程序 8a 时，将表示执行刚刚开始之后是否停止保护程序的停止标志设为有效 (S304)。执行将停止标志设为有效的处理的理由在后述的‘1.4.3.2 预处理的补充’中说明。停止标志是 1 比特的信息，存储在保护机构内寄存器或存储器等存储区域中。

[0337] 调试器 14 从程序开发者受理涉及调试处理的操作，执行对通常程序 12a 的调试处理。若结束必要的处理，则调试器 14 从程序开发者受理程序重新开始执行的操作，重新开始执行作为调试对象的通常程序 12a(S305)。

[0338] 重新开始执行的通常程序 12a 对切换设备驱动器 13 进行 open，经由切换设备驱动器 13，向保护 OS6 请求保护程序 8a 的加载处理 (S306) 与保护程序 8a 的执行处理 (S307)。

[0339] 步骤 S306 的加载处理与图 7 所示的步骤 S202、S203、S204 基本一样。另外，步骤 S307 的执行处理与图 8 的步骤 S206、S207、S208 基本一样。不同的是当调试器 14 动作时，调试功能 7 为了判断通常程序 12a 是否被调试，将通常程序 12a 的处理 ID 与各处理中传递的数据一起通知 (S303)。

[0340] 另外，当保护 OS6 受理保护程序 8a 的执行请求时，保护 OS6 向调试功能 7 请求执行预处理 (S308)。调试功能 7 受理请求，执行预处理。这里，所谓预处理是指调试功能 7 判断对应于处理 ID 的停止标志是否有效 (S309)，在有效的情况下 (S309 :是)，将位于保护程序 8a 的入口点的命令变更为断开 (break) 命令 (S310)。若停止标志不是有效 (S309 :否)，则调试功能 7 不变更入口点的命令。

[0341] 保护 OS6 执行保护程序 8a(S311)。

[0342] 1.4.3.2 预处理的补充

[0343] 在上述动作中，在步骤 S304 中，使停止标志有效，但执行该处理主要是为了程序开发者能容易调试。

[0344] 下面，示出执行这种处理的理由，在本实施方式 1 中，如不执行调试时的处理的说明中所述的那样，保护程序 8a 在被通常程序 12a 调用时被加载，不需要时，则从存储器上删除。这里，由于本实施方式 1 的程序从执行通常程序 12a 起开始，所以程序开发者难以确认执行已被切换到保护程序 8a 这一情况。

[0345] 另外，本实施方式 1 的调试器 14 附着于通常程序，所以程序开发者也不能对保护程序 8a 直接设定断点，难以调试保护程序 8a。

[0346] 因此，为了让程序开发者知道处理已移动到保护程序 8a，对保护程序 8a 提供可设定断点的机会等，设当读出保护程序 8a 时，停止处理。

[0347] 1.4.4 调试处理

[0348] 图 12 是在保护程序 8a 执行中发生基于断点的调试例外、并使用调试器 14 对保护程序 8a 执行调试处理时的流程图。

[0349] 1.4.4.1 调试处理

[0350] 如图 12 所示，在保护程序 8a 执行中，若发生基于保护程序 8a 中设定的断点的调试例外，则向保护 OS6 通知调试例外 (S401)。

[0351] 保护 OS6 受理调试例外的通知，向切换设备驱动器 13 通知调试例外的发生 (S402)。

[0352] 切换设备驱动器 13 若受理调试例外的发生通知，则为了让调试器 14 执行调试处理，由调试请求受理部 44 向调试器 14 请求调试处理执行 (S403)。

[0353] 调试器 14 若受理调试处理执行的请求，则为了向程序开发者提供调试信息，经由调试器用切换设备驱动器 15，向保护 OS6 请求取得调试信息 (S404)。此时，还向保护 OS6 请求调试器 14 的调试器 ID、或未图示的调试用通信区域的通知。这里，调试用的通信区域是通常模式与保护模式双方能访问的区域，在从保护模式向通常模式传递调试信息时使用。

[0354] 保护 OS6 向调试功能 7 请求取得调试信息 (S405)。

[0355] 调试功能 7 利用调试器 ID 判定部 22 和访问判定部 23，判定是否允许对保护程序 8a 的调试处理、和对保护程序 8a 的涉及调试处理的规定部分的访问 (S406)。

[0356] 在步骤 S406 中，在判定为允许调试处理和访问的情况下 (S406 :是)，则调试信息取得部 25 取得保护程序 8a 的调试信息，将调试信息拷贝到调试用通信用区域 (S407)。拷贝后，调试功能 7 经由保护 OS6、调试器用切换设备驱动器 15，将调试信息取得完成通知给调试器 14，处理从调试功能 7 回到调试器 14(S408)。

[0357] 调试器 14 取得拷贝到调试用通信用区域的调试信息，通过显示于未图示的显示部中，向程序开发者示出调试信息 (S409)。

[0358] 之后，当程序开发者参照调试信息结束必要的处理时，调试器 14 从程序开发者受理规定的操作，经由切换设备驱动器 13，向保护 OS6 请求重新开始执行作为调试对象的保护程序 8a(S410)。

[0359] 保护 OS6 受理重新开始执行的请求，重新开始执行保护程序 8a(S411)。

[0360] 之后，在再次发生调试例外的情况下，以同样的流程执行处理。

[0361] 1.4.4.2 调试处理的补充

[0362] 在以上的实例中，说明了保护程序 8a 执行中发生基于断点的调试例外时的处理。此外，作为使用调试功能 7 的调试处理的模式 (pattern)，还可考虑各种方式。具体而言，由程序开发者请求断点的设定处理、或寄存器值、存储器值的设定或取得处理等其它调试处理，对应于该请求，调试功能 7 执行调试处理。在这些情况下，也以与上述一样的流程，由调试功能 7（严格地讲，为安全调试器 24 具有的各功能部）执行处理，但由于基本的处理是一样的，所以省略详细说明。

[0363] 在本实施方式 1 中，设当发生调试例外时，能进行对涉及该例外发生的保护程序 8a 的调试处理，但这不限于保护程序 8a。也可进行对调试器 14 调试的通常程序 12a 等通常程序的调试处理。

[0364] 2 实施方式 2

[0365] 下面说明实施方式 2。在实施方式 2 中，具体说明保护程序 8 的生成方法和生成保护程序 8 的程序生成装置。

[0366] 2.1 程序的生成方法的概要

[0367] 图 13 是说明本发明的保护程序 8 的生成方法用的图。

[0368] 保护程序 8 被加密生成。为了生成加密的保护程序 73，使用保护程序源代码 71、保护程序生成装置 72、作为附加于保护程序的信息的允许调试器 ID 存储文件 74 与访问控制列表存储文件 75、以及秘密信息区域存储文件 76。

[0369] 图 13 所示的保护程序源代码 71 是记述了保护程序 8 的动作的源代码。

[0370] 保护程序生成装置 72 执行保护程序源代码 71 的编译与链接。向生成的执行文件附加允许调试器 ID 信息与访问控制列表，执行加密。并且，通过将解密所需的信息作为解密用头信息来附加，从而生成加密的保护程序 73。具体如后所述。

[0371] 加密的保护程序 73 是由保护程序生成装置 72 生成的程序。

[0372] 允许调试器 ID 存储文件 74 与访问控制列表存储文件 75，分别是包含对保护程序 8 执行调试时调试功能 7 或调试器 14 使用的允许调试器 ID 信息、访问控制列表的数据。

[0373] 秘密信息区域存储文件 76 由程序中的各信息的区域、和表示涉及该区域的信息是否是秘密信息的秘密信息区分构成。

[0374] 保护程序开发者制作保护程序源代码 71。另外，将源代码中允许从调试器访问的区域与不允许的区域制作为访问控制列表，记录在访问控制列表存储文件 75 中。并且，将秘密信息的区域与非秘密信息的区域记录在秘密信息区域存储文件 76 中。允许调试器 ID 存储文件 74 另外获得。将获得的允许调试器 ID 存储文件 74 与保护程序源代码 71、访问控制列表存储文件 75、秘密信息区域存储文件 76 作为输入，使保护程序生成装置 72 动作。结果，生成加密的保护程序 73。

[0375] 2.2 保护程序生成装置 72 的构成

[0376] 图 14 是保护程序生成装置 72 的构成图。

[0377] 保护程序生成装置 72 由编译器 77、连接程序 78、保护程序化工具 79 构成。

[0378] 2.2.1 编译器 77

[0379] 图 14 所示的编译器 77 编译所输入的保护程序源代码 71，生成目标文件 (object

file)。制作表示变量或函数配置的符号信息、或目标文件中的程序代码与源代码的对应关系等，作为调试信息，附加于目标文件中。

[0380] 2.2.2 连接程序 78

[0381] 连接程序 78 将由编译器 77 生成的目标文件与库相链接，生成可执行的文件。并且，连接程序 78 生成表示在所生成的可执行文件中的何处配置何变量或函数的符号文件。

[0382] 2.2.3 保护程序化工具 79

[0383] 保护程序化工具 79 向连接程序 78 制作的可执行文件的头，附加输入到保护程序生成装置 72 的允许调试器 ID 存储文件 74 中存储的允许调试器 ID 信息、与访问控制列表存储文件 75 中存储的访问控制列表。并且，将秘密信息区域存储文件 76 中记载的各区域是否是秘密信息的信息附加于调试信息，生成保护程序。

[0384] 保护程序化工具 79 利用由保护 OS6 与保护程序生成装置 72 共同保持的密钥，对生成的保护程序执行加密，附加加载保护程序的地址等，作为解密用头信息。

[0385] 加密使用保护 OS6 与保护程序生成装置 72 共同保持的密钥（所谓公用密钥加密方式），但不用说，也可使用彼此保持不同密钥的公开密钥加密方式等。

[0386] 在访问控制列表以符号名与访问允许信息的组的形式来提供的情况下，也可使用连接程序 78 输出的符号文件，从符号名求出配置该符号的区域，将访问控制列表的符号名变更到该区域（即表示可执行文件中的何处的信息）。

[0387] 加密的保护程序 73 的构成如在实施方式 1 中使用图 5 说明的那样。加密的保护程序 73 由保护程序主体 51 与允许调试器 ID 信息 52、访问控制列表 53、解密用头信息 54 构成，将允许调试器 ID 信息 52 与访问控制列表 53 附加于保护程序主体 51 的头，形成加密构造。解密用头信息 54 由于存储着解密所需的数据，因此也可不加密。

[0388] 2.3 保护程序生成装置 72 的动作

[0389] 下面，说明保护程序生成装置 72 生成加密的保护程序 73 的处理。

[0390] 图 15 是表示保护程序生成装置 72 生成加密化的保护程序 73 的处理的流程图。

[0391] 设保护程序源代码 71、访问控制列表存储文件 75、秘密信息区域存储文件 76 已由程序开发者制作。即，程序开发者记述保护程序源代码 71。另外，决定源代码中不可执行调试器访问的区域与可执行调试器访问的区域，以制作访问控制列表，记述为访问控制列表存储文件 75。将记载秘密信息的区域记述在秘密信息区域存储文件中。

[0392] 设程序开发者另外获得允许调试器 ID 存储文件 74。

[0393] 说明保护程序生成装置 72 的动作，保护程序生成装置 72 受理保护程序源代码 71、允许调试器 ID 存储文件 74、访问控制列表存储文件 75、和秘密信息区域存储文件 76，作为输入 (S501)。

[0394] 保护程序生成装置 72 使用编译器 77 与连接程序 78，执行输入的保护程序源代码 71 的编译与链接 (S502)。

[0395] 保护程序化工具 79 向通过保护程序源代码 71 的编译与链接而生成的保护程序主体 51，附加访问控制列表存储文件 75 中存储的访问控制列表 53、与调试器 ID 存储文件 74 中存储的允许调试器 ID 信息 52(S503)，并且，向调试信息追加秘密信息区域存储文件 76 中记载的各区域是否是秘密信息的信息，执行加密 (S504)。

[0396] 保护程序化工具 79 向加密的程序附加保护程序主体 51 的加载目的地地址等解密

所需的信息，作为解密用头信息 54，作为加密的保护程序 73 输出 (S505)。

[0397] 2.4 实施方式 2 的补充说明

[0398] 在上述说明中，设程序开发者执行向秘密信息区域存储文件 76 的写入，但不限于此。例如，也可由编译器等计算机程序自动执行这些访问控制列表或秘密信息存储文件的制作等。具体而言，也可对源代码，在秘密信息所在的部位事先赋予某个标记，编译器根据该标记的有无，执行向秘密信息存储文件的写入。

[0399] 在上述说明中，保护程序生成装置 72 执行从保护程序源代码的编译至保护程序制作的全部处理，但不限于此。例如，也可由不同的装置来执行保护程序的生成与访问控制列表等的附加。此时，保护程序生成装置 72 构成为 :编译和链接保护程序源代码 71 后生成保护程序主体 51 的装置、与取得生成的保护程序主体 51 后附加访问控制列表等的装置的组。在上述构成中，生成保护程序主体 51 的装置，作为输入只要有保护程序源代码 71 即可。

[0400] 由于向附加访问控制列表等的装置提供保护程序主体 51，所以不必输入保护程序源代码 71。根据该构成，由于分别执行保护程序主体 51 的制作与访问控制列表等的附加，所以可通过将各个作业委托给不同的公司等，实现程序开发效率的提高。

[0401] 3 实施方式 3

[0402] 下面，说明实施方式 3。在实施方式 3 中，具体说明如何管理调试器 ID。在实施方式 3 中，设调试器 ID 由调试器 ID 管理服务器管理，来进行说明。

[0403] 图 16 是说明调试器 ID 管理服务器执行的调试器 ID 的管理方法的图。

[0404] 3.1 构成的说明

[0405] 3.1.1 调试器 ID 管理服务器 81

[0406] 图 16 所示的调试器 ID 管理服务器 81 管理调试器 ID。调试器 ID 的管理使用后述的调试器 ID 管理文件 90 来进行。

[0407] 为了按照保护程序开发者是谁（或按照保护程序开发者使用的保护程序开发装置 82）来控制调试处理可否执行，期望调试器 ID 为对每个保护程序开发者（或每个保护程序开发装置 82）分别不同的值。因此，调试器 ID 管理服务器 81 需要以不向多个保护程序开发者（或保护程序开发装置 82）赋予相同的调试器 ID 的方式，进行管理。

[0408] 调试器 ID 管理服务器 81 是管理调试器 ID 的公司（调试器 ID 管理公司）具有的服务器（设为‘公司’，但不限于此，只要是管理调试器 ID 的管理者，则即便是公司以外的组织或个人也无妨）。

[0409] 调试器 ID 管理服务器 81 根据来自保护程序开发装置 82 的请求，发行与过去发行的调试器 ID 不同的调试器 ID，将存储了所发行的调试器 ID 的调试器 ID 存储文件，提供给保护程序开发装置 82。另外，调试器 ID 管理公司制作具有对应于调试器 ID 的 ID 的调试器，提供给保护程序开发装置 82。

[0410] 3.1.2 保护程序开发装置 82

[0411] 保护程序开发装置 82 使用保护程序生成装置 72，制作保护程序。

[0412] 具体而言，保护程序开发装置 82 从保护程序解析装置 83 受理调试器 ID 存储文件。取得所受理的调试器 ID 存储文件所示的调试器 ID，作为允许调试器 ID 存储文件 74，输入到实施方式 2 说明的保护程序生成装置 72，由此，生成允许对所取得的调试器 ID 所示

的调试器的调试的保护程序 8。

[0413] 保护程序开发装置 82 由开发保护程序的程序开发者（个人或组织）所有。

[0414] 3.1.3 保护程序解析装置 83

[0415] 保护程序解析装置 83 是用于执行保护程序中包含的差错解析的装置。保护程序解析装置 83 由个人或组织所有。

[0416] 具体而言，保护程序解析装置 83 从调试器 ID 管理服务器 81 获得发行调试器 ID，将获得的调试器 ID 存储文件提供给开发解析对象的保护程序的保护程序开发装置 82。

[0417] 若提供了调试器 ID 存储文件的保护程序开发装置 82 根据调试器 ID 存储文件生成保护程序，则从保护程序开发装置 82 获得可调试的保护程序，解析该保护程序。

[0418] 3.2 调试器 ID 管理文件 90 的数据构造

[0419] 图 17 是表示调试器 ID 管理服务器 81 用于调试器 ID 管理中的调试器 ID 管理文件 90 的数据构造的图。

[0420] 调试器 ID 管理文件 90 的 1 条记录由管理序号 91、调试器 ID92、保护程序的开发者名 93 与联络地址 94 构成。

[0421] 管理序号 91 中存储管理调试器 ID 管理服务器 81 已发行的调试器 ID 用的序号。

[0422] 调试器 ID92 中存储由管理序号 91 管理的已发行的调试器 ID 的值。为了能识别各程序开发者（或每个保护程序开发装置 82），并且防止模仿调试器的调试 ID 执行调试处理的模仿攻击，期望调试器 ID92 为足够长的数值串。

[0423] 保护程序的开发者名 93 中存储可申请发行调试器 ID 的程序开发者的名称。

[0424] 联络地址 94 存储可申请发行调试器 ID 的程序开发者的联络地址。

[0425] 3.3 实施方式 3 的补充说明

[0426] 在实施方式 3 中，调试器 ID 管理服务器 81 向保护程序开发装置 82 提供调试器，但调试器的提供目的地不限于保护程序开发装置 82。例如，也可提供给保护程序解析装置 83。

[0427] 另外，设调试器 ID 管理文件 90 为管理已发行的调试器 ID 用的文件，但不限于已发行的调试器 ID，也可与未发行的调试器 ID 一起管理。

[0428] 并且，调试器 ID 管理文件 90 还管理已发行调试器 ID 的发行目的地的程序开发者名称或其联络地址，但也可仅管理已发行的调试器 ID。

[0429] 调试器未必从调试器 ID 管理公司提供，也可从调试管理公司委托开发调试器的其他人的服务器等提供。此时，其他人从调试器 ID 管理服务器 81 取得调试器 ID 的信息。

[0430] 4 实施方式 4

[0431] 在下面的实施方式中，说明如何显示上述调试器 14 等执行的调试处理的结果。

[0432] 实施方式 4 是向实施方式 1 的调试器附加了将程序的动作信息显示于图形用户界面上的显示部件的实施方式。调试器主体的功能与实施方式 1 一样，所以省略说明。

[0433] 这里，所谓程序的动作信息是指为了调试构成调试对象的程序而参照的信息、即与程序动作关联的信息。具体而言，在以下的实例中，动作信息指程序的代码、或执行程序的处理器的各寄存器的值、局部变量的符号名及其值、由符号指定的变量及其值、存储器的使用量等值、函数的调用层和以保护模式与通常模式哪个模式来动作的信息等。以实施方式 1 为例，保护程序 8 的动作信息根据调试器 14 经调试功能 7 取得的信息获得，通常程序

12 的动作信息根据调试器 14 直接取得的信息获得。

[0434] 4.1GUI150a 的说明

[0435] 图 18 表示用于显示程序的动作信息的图形用户界面 (GUI)。

[0436] 图 18(a) 中，表示调试器 14 附着于通常程序 12，调试通常程序 12 时的显示部件的画面构成，即 GUI150a。另外，图 18(b) 中表示通常程序 12 执行保护程序 8，调试通常程序 12 与保护程序 8 时的显示部件的画面构成，即 GUI150b。

[0437] 如图 18(a) 所示，GUI150a 由代码显示部 151、寄存器显示部 152、存储器显示部 153、符号显示部 154、观察点 (watch point) 显示部 155、调用堆栈 (call stack) 显示部 156、窗口标题 (window title) 显示部 157、菜单显示部 158 构成。

[0438] 代码显示部 151 是用于显示调试对象的程序的代码的显示部，显示源代码或汇编码、机器语言。

[0439] 寄存器显示部 152 中显示执行程序的处理器的各寄存器的值。

[0440] 存储器显示部 153 中显示存储器的值。

[0441] 符号显示部 154 显示调试对象的程序中、在停止的函数内使用的局部变量的符号名及其值。

[0442] 观察点显示部 155 显示由符号指定的变量及其值。

[0443] 调用堆栈显示部 156 显示调试对象的程序中、调用停止的函数前的调用层。

[0444] 窗口标题显示部 157 是用于显示窗口的标题的显示部，显示调试器或调试对象的程序的程序名或程序的状态。

[0445] 菜单显示部 158 显示调试器的菜单。作为菜单，有调试对象的程序的打开或附着、调试器的结束、设定调试器的动作或程序的动作信息显示方法的设定画面的显示、调试对象程序的执行或中断、重新开始、分步执行等。

[0446] 模式显示部 159 是用于显示程序因断点等停止时、执行程序的模式的显示部，在通常程序执行中停止的情况下为了表示是通常模式，显示为‘通常模式’，在保护程序执行中停止的情况下为了表示是保护模式，显示为‘保护模式’。

[0447] 4.2GUI150a 的说明补充

[0448] 在实施方式 4 中，为了显示通常程序 12 与保护程序 8 的哪个中发生调试例外，在模式显示部中显示为‘通常模式’或‘保护模式’，但不限于此。例如，模式的显示不限于‘通常模式’或‘保护模式’的字符串，只要是可明白是不同模式的显示均可。举具体实例，可以是图标等。并且，显示部位也不限于模式显示部，可由窗口整体显示，也可由窗口的颜色等区别。另外，也可另外设置模式显示用的窗口。

[0449] 在这些显示部中，显示调试器 14 刚刚附着于通常程序之后、或通常程序被断点命令停止且可由调试器 14 调试的状态时、停止执行的通常程序的停止时刻的各种状态、或调试器 14 执行中调试器 14 的执行状态。并且，也可利用来自用户的输入等变更对寄存器或存储器、变量等设定的值。

[0450] 4.3GUI150b 的说明

[0451] 如图 18(b) 所示，GUI150b 由通常程序用调试窗口 160 与保护程序用调试窗口 161 构成。

[0452] 通常程序用调试窗口 160 是用于显示通常程序 12 的各种信息的窗口。

[0453] 保护程序用调试窗口 161 是用于显示保护程序的各种信息的窗口。不管哪个窗口（通常程序用调试窗口 160、保护程序用调试窗口 161）都构成为在窗口内显示图 18(a) 的各显示部（代码显示部 151、寄存器显示部 152、存储器显示部 153、符号显示部 154、观察点显示部 155、调用堆栈显示部 156）。由于图面复杂，所以图 18(b) 中，未记载这些各显示部。

[0454] 使用具有这种显示部件的调试器，当调试器的用户调试通常程序 12 与保护程序 8 时，在调试器刚刚启动之后或刚刚附着于通常程序之后的状态下，为图 18(a) 所示的画面构成。

[0455] 之后，通常程序 12 执行保护程序 8，在保护程序 8 的执行中发生基于断点的调试例外的情况下，为了显示保护程序的各种信息，将图 18(a) 的画面分割为两个，变更为图 18(b) 所示的画面构成。

[0456] 此时，为了让用户知道是保护程序 8 的调试，执行弹出显示，并且响起嘀声等特定声音或由用户设定的声音，在模式显示部 159 中显示为‘保护模式’，由此让用户意识到是保护模式。

[0457] 4.4 实施方式 4 的补充

[0458] 在实施方式 4 中，设在保护程序 8 调试时将调试器的窗口分割成两个，但不限于分割。例如，也可为了调试保护程序而重新生成窗口，利用制表 (tab) 或菜单，切换通常程序用调试窗口 160 与保护程序用调试窗口 161。

[0459] 在保护程序 8 的执行中发生调试例外，并变为可调试的状态时，可将焦点移动到保护程序用调试窗口 161，对保护程序实施断点的设定或存储器、寄存器值的变更等。并且，也可对通常程序 12 执行同样的设定或变更操作。

[0460] 在实施方式 4 中，在保护程序 8 的执行中发生调试例外时，调试器的用户同时执行保护程序 8 的调试与通常程序 12 的调试，但不限于同时执行调试。例如，也可仅将可执行调试的对象限于保护程序，或限制成不能访问通常程序的调试窗口 160。

[0461] 如实施方式 1 所述，使用调试器 14 与调试功能 7 的保护程序 8 的调试，根据调试器 14 而受到限制。因此，存在不允许对保护程序 8 的调试处理的情况、或要显示不能用访问控制列表允许访问的区域的情况等、在各显示部中不能显示信息的情况。

[0462] 这样，在请求了不能显示的信息的情况下，必需让使用调试器的用户知道不能显示这一情况。作为通知存在这种不能显示的信息的方法，执行表示不能显示的弹出显示，并且响起嘀声等特定声音或由用户设定的声音，用‘*’等特定的字符串显示不能显示的部分。

[0463] 上述中，用‘*’等特定的字符串显示不能显示的部分，但不限于用特定的字符串显示，也可显示图标或不做任何显示。另外，也可使不能显示的部分的背景色与可显示的部分的背景色不同。

[0464] 当调试保护程序 8 时，存在保护程序 8 中的秘密信息、或通常程序 12 与保护程序 8 的通信用共有领域等、程序的开发或解析时重要的信息或区域。因此，调试器 14 中，通过在这种必需注意的信息或区域的显示方法方面做出努力，督促使用调试器的用户注意，是优选的。

[0465] 作为秘密信息的显示方法，当调试器 14 执行对保护程序 8 的调试处理时，取得涉及保护程序 8 的调试信息中包含的秘密信息区域的信息，判定各显示部中显示信息时要显示的代码或数据是否包含于秘密信息区域中。

[0466] 在判定的结果是秘密信息的情况下，当通过用户的设定设定为不显示秘密信息时，不显示秘密信息，以该部分空白的状态来显示。

[0467] 另一方面，在设定为显示秘密信息的情况下，通过将显示该秘密信息时的字符颜色设为由用户设定的颜色（例如红色），强调是秘密信息。并且，显示信息时，执行表示显示秘密信息的弹出显示，响起嘀声等特定声音或由用户设定的声音。通过如此动作，督促用户注意是秘密信息。

[0468] 在实施方式 4 中，设在设定为不显示秘密信息的情况下以该部分为空白的状态来显示，但不限于以空白的状态来显示，也可用与背景色不同的颜色来涂布显示秘密信息的部分，或使用图标。即，只要是用户看不到秘密信息内容的状态，则可以是任何显示方法。

[0469] 作为是通常程序 12 与保护程序 8 共有的信息的共有信息的显示方法，在调试器 14 执行对保护程序 8 的调试处理时，从保护 OS6 取得涉及通常模式与保护模式下共有信息的区域的信息，检查各显示部中显示信息时要显示的代码或数据的区域是否包含于共有信息的区域中。

[0470] 在是共有信息的区域的情况下，通过将显示该区域的信息时的字符颜色设为由用户设定的颜色（例如黄色），强调是共有信息。并且，显示信息时执行表示显示共有信息的弹出显示，响起嘀声等特定声音或由用户设定的声音。通过如此动作，督促用户注意是共有信息。

[0471] 共有信息也可与秘密信息一样适用多种显示方法，但也可使用与秘密信息不同的显示方法以便能区别共有信息与秘密信息。

[0472] 在本实施方式 4 中，设显示秘密信息或共有信息时的字符颜色为由用户设定的颜色，但不限于变更字符颜色，也可变更背景颜色，或变更粗字或斜体等字符的风格，或附加下划线或底纹等字符修饰，也可包围秘密信息或共有信息整体。

[0473] 5 实施方式 5

[0474] 实施方式 5 与实施方式 4 不同，对实施方式 1 的调试器，附加在基于字符的用户界面中显示程序的动作信息的显示部件。实施方式 5 的调试器主体的功能也与实施方式 1 一样，所以省略说明。

[0475] 5.1CUI 的说明

[0476] 图 19 是本发明实施方式 5 的基于字符的用户界面 (CUI170) 的显示方法的说明图。

[0477] 若调试器 14 启动，则显示与控制台 (console) 不同的、例如‘(dbg)’这样的提示符，表示调试器 14 处于可使用的状态。该状态表示通常程序 12 可调试（调试处理结果的显示例 171a)。

[0478] 若执行通常程序 12，由通常程序 12 执行保护程序 8，则根据实施方式 1 所述的动作，首先在最初执行时保护程序的入口点设定的断点发生调试例外。

[0479] 此时，调试器 14 为了表示保护程序 8 可调试，通过显示与通常程序 12 的调试时不同的、例如‘(dbg-sec)’这样的提示符，使得用户能区别。所谓提示符是使用户能区别地显示来自调试器 14 的消息的显示。例如如图所示，也可使用括号来表示提示符。

[0480] 调试器 14 在保护程序 8 的执行中停止的情况下，通过将提示符或来自调试器 14 的消息、从用户输入的字符的风格变更为斜体，显示在保护程序 8 的执行中停止这一情况，

使用户可区别（调试处理结果的显示例 171b）。

[0481] 在上述状态下，可执行以停止执行的程序为对象的调试，但由于实施方式 5 中执行以通常程序 12 与保护程序 8 协同动作的状态为对象的调试，所以期望在通常程序 12 的执行中调试保护程序 8，或相反，在保护程序 8 的执行中可调试通常程序 12。

[0482] 下面，说明此时使用的界面。

[0483] 首先，在通常程序 12 的执行中发生调试例外，通常程序 12 为可调试状态时，有时调试器 14 的用户想执行保护程序 8 的调试而不是通常程序 12 的调试。此时，在通常程序 12 为可调试的状态下，通过输入‘secure’等命令，数据处理装置 1 转移到保护模式，保护程序 8 变为可调试状态。

[0484] 此时，提示符从‘(dbg)’变更为‘(dbg-sec)’，但由于通常程序 12 执行中停止的状态不变，所以字体不变为斜体等（调试处理结果的显示例 171c）。

[0485] 相反，在保护程序 8 的执行中发生调试例外，保护程序为可调试状态时，在想执行通常程序的调试而不是保护程序的调试的情况下，通过输入‘normal’等命令，通常程序变为可调试状态。

[0486] 此时，提示符从‘(dbg-sec)’变更为‘(dbg)’，但由于保护程序 8 执行中停止的状态不变，所以字体仍以斜体显示，不恢复成通常的字体（调试处理结果的显示例 171d）。

[0487] 调试器 14 的用户在要调试保护程序 8 的情况下，不允许调试、或要显示禁止访问的区域信息的情况下，当输入命令时，显示‘Access invalid’等，由此通知用户不允许调试或禁止访问。

[0488] 5.2 实施方式 5 的补充

[0489] 实施方式 5 中，变更为了区别通常程序 12 可调试还是保护程序 8 可调试而显示的提示符，但不限于变更提示符。

[0490] 例如，也可在提示符之外，每当从用户请求的处理结束时，显示表示当前状态的字符。另外，也可将提示符或来自调试器的消息、从用户输入的字符的风格变更为粗字或斜体，也可变更字符的颜色或背景色，或附加下划线或底纹等修饰。

[0491] 并且，为了区别在通常程序 12 的执行中停止还是在保护程序执行中停止，将字符的风格变更为斜体，但不限于将字符的风格变为斜体。

[0492] 例如，也可变更为粗字等其它风格，或变更字符的颜色或背景色，或附加下划线或底纹等字符修饰。并且，也可在行头显示区分是在通常程序 12 的执行中停止还是在保护程序 8 的执行中停止的字符串。

[0493] 另外，每当通常程序 12 或保护程序 8 停止，用户可输入命令时，或通常程序 12 的调试与保护程序 8 的调试切换时，也可消除全部以前的输出。

[0494] 在本实施方式中，命令或消息的一例如下所示。

[0495] 消息‘Change to Secure mode.’表示数据处理装置 1 移动到保护模式。消息‘Change to Normal mode.’表示数据处理装置 1 移动到通常模式。消息是数据处理装置 1 执行对应于用户输入的命令的处理之后，通知该处理结果的消息。

[0496] 命令‘secure’是用于移动到保护模式的命令，命令‘normal’是用于移动到通常模式的命令。

[0497] 可输入调试器的命令或对应于命令而输出的消息，按照各调试器而不同，所以省

略详细说明。

[0498] 6 补充

[0499] 根据上述实施方式说明了本发明，但本发明当然不限于上述实施方式。下面的情况也包含于本发明中。

[0500] (1) 在上述实施方式中，允许调试器 ID 信息 52 或访问控制列表 53 包含于保护程序 8 中，但不限于此，也可从数据处理装置的外部取得允许调试器 ID 信息 52 或访问控制列表 53。此时，最好与表示与哪个保护程序 8 对应的信息一起取得涉及取得的允许调试器 ID 信息 52 等。

[0501] 由此，可容易地判定可否执行对保护程序 8 的调试处理。

[0502] 由于允许调试器 ID 信息 52 等表示可调试的调试器 ID 或可访问的区域，所以当被解析时，由于不期望保护保护程序 8，所以需要存储在数据处理装置中保护机构等安全区域中，以便不被非法访问暴露。

[0503] 另外，允许调试器 ID 信息 52 等的取得当然也需要安全地执行，以不被非法者的窃听等而非法取得。

[0504] (2) 在上述实施方式中，设保护程序 8 或通常程序 12 在保护 OS6 或通常 OS11 上动作，但不限于此。例如，也可不经 OS 而直接动作。

[0505] 此时，将调试功能 7 或切换设备驱动器 13 和调试用切换设备驱动器 15 设置为 LSI2 的功能等，也执行各 OS 执行的中断监视等处理。另外，考虑调试器 14 也直接由 LSI2 上的 CPU 等可执行的语言形成等。

[0506] (3) 在上述实施方式中，设将调试器 14、切换设备驱动器 13、调试器用切换设备驱动器 15、调试功能 7 等安装为在 LSI2 上动作的软件，但不限于此。例如，也可作为 LSI2 的功能实现，或以与 LSI2 相互通信的硬件等的形式实现。另外，也可仅将各构成要素的一部分实现为 LSI 功能，或实现为硬件。

[0507] (4) 在上述实施方式 3 中，为了拥有多个保护程序开发装置 82 的组织等，也可按照经保护程序开发装置 82 发送的请求，向多个保护程序开发装置 82 赋予同一调试器 ID。

[0508] (5) 在上述实施方式 4 和实施方式 5 中，参照图 18 和图 19 介绍了用户界面的实例，但用户界面的显示不限于图 18 和图 19 所示。

[0509] 即，也可不显示构成界面的部分构成要素，当然也可适当替换各构成要素的显示位置，或将字符等的风格变为其它形式。

[0510] (6) 在上述实施方式中，调试功能 7 通过调试器 ID 的确认来确认允许调试这一情况，并仅在通过访问控制列表的确认允许对所请求的区域的访问的情况下，允许进行调试，但也可仅通过任一确认来允许调试。

[0511] 另外，确认的顺序也不限于在调试器 ID 的确认后执行访问控制列表的确认的顺序，也可先确认任一个。

[0512] (7) 在上述实施方式中，调试器 14 附着于通常程序 12，执行对与该通常程序 12 协同动作的保护程序 8 的调试处理，但不限于此。例如，也可以是可直接附着于保护程序 8 的调试器。

[0513] 此时，由于调试器 14 不能附着于未加载的程序，所以将保护程序 8 事先加载到存储器上，经由调试器用切换设备驱动器 15 将保护程序 8 中发生的调试例外通知给调试器

14。

[0514] 并且，在数据处理装置具有可在保护模式下使用的输入输出装置的情况下，调试器也可以保护模式而非通常模式动作。此时，调试功能 7 与调试器 14 不经由调试器用切换设备驱动器 15 而直接通信。

[0515] (8) 在上述实施方式中，通过在预处理中将停止标志设为有效，程序在保护程序 8 等的执行开始时必然停止，但不限于此。

[0516] 例如，在如变形例 (7) 所示调试器 14 直接附着于保护程序 8 并可设定断点的情况、或希望仅调试通常程序 12 的情况下，若程序在保护程序 8 的执行开始时必然停止，则不便。因此，也可不在预处理时始终将停止标志设为有效，可通过用户的选择来决定是否变为有效。

[0517] (9) 在上述实施方式中，未特别言及调试器 ID 的生成方式，但可考虑如下。

[0518] 例如，也可以是调试器 ID 管理服务器 81 生成随机数，作为调试器 ID，分配给调试器。

[0519] 另外，除此之外例如也可将保护程序 8 的一部分或全部的散列值用作调试器 ID。

[0520] 此时，调试器 ID 判定部 22 在比较值保持部 33 中保持调试器 14 的调试器 ID。若从调试器 14 请求调试，则调试器 ID 运算部 32 计算构成调试请求对象的保护程序 8 的散列值，调试器 ID 比较部 31 比较计算结果与比较值保持部 33 保持的值，由此进行判定。

[0521] 由此，在保护程序 8 的内容变化的情况下，散列值与调试器 ID 的比较不成功，因此一旦保护程序 8 升级，则可自动禁止由旧的调试器进行的调试。

[0522] 此时，实施方式 3 的保护程序开发装置 82 在保护程序 8 中取得散列值的部位完成的时刻向调试器 ID 管理服务器 81 发送该保护程序 8。调试器 ID 管理服务器 81 计算保护程序 8 的散列值，将该散列值作为调试器 ID，返回到保护程序开发装置 82。返回的调试器 ID 是仅保护程序 8 的开发者能知道的信息。因此，调试器的制造者为了制成对应于该保护程序 8 的调试的调试器，需要从保护程序 8 的开发者接受通知相应的调试器 ID，将通过通知取得的调试器 ID 分配给调试器 14。但是，即便在上述情况下，在调试器 ID 管理服务器 81 的所有者与调试器的制造者相同的情况下，调试器的制造者尽管未接收调试器 ID 的通知，也可制成调试器。这是因为调试器的制造者能够从自身所具有的调试器 ID 管理服务器 81 知道调试器 ID。

[0523] 另外，如上所述在使用散列值的情况下，通过计算散列值，算出可执行对保护程序 8 的调试处理的调试器的调试器 ID，所以不必使允许调试器 ID 信息 52 包含于保护程序 8 中。

[0524] 并且，调试器 ID 中也可包含每个程序开发者的值与每个程序的值。此时，通过每个程序开发者的值的检查，来确认程序开发者单位下的调试允许 / 不允许，仅在存在程序开发者单位下的调试允许的情况下，才可通过检查程序单位的值，确认程序单位下的调试允许等，实现更细致的调试控制。

[0525] (10) 在上述实施方式中，说明检测出在保护程序 8a 执行中发生基于断点的调试例外后执行对保护程序 8a 的调试处理的情况。不限于此，也可在保护程序 8 的执行中仅发生了一般的错误中断的情况下，通过检测该中断，由调试器 14 执行调试处理。所谓一般的错误中断，例如有在保护程序 8 执行中发生基于 0 的除法的情况、或发生溢出的情况等。在

发生一般的错误中断的情况下，通过执行 OS6 将调试例外的发生通知给切换设备驱动器 13 等的处理 (S402 等)，调试器 14 可执行对保护程序 8 的调试处理。

[0526] (11) 在上述实施方式中，控制可否执行对保护程序 8 的调试处理，但不限于此，即便对于通常程序 12，也可分配允许调试处理的调试器的调试器 ID，控制可否执行对通常程序 12 的调试处理。

[0527] (12) 也可将上述构成要素的一部分或全部实现为 LSI 等集成电路。此时的集成电路可以是与 LSI2 相同的集成电路，也可以是不同的集成电路。

[0528] LSI 也可根据集成度的不同，被称为 IC、系统 LSI、超 (super)LSI、甚 (ultra)LSI，但不用说，以上述任一集成度实现系统 LSI2 的情况都包含于本发明中。另外，也可在 LSI 制造后，利用可编程的 FPGA(Field Programmable Gate Array :现场可编程门阵列) 或可再构成 LSI 内部的电路单元连接或设定的可重构处理器。

[0529] 并且，如果因半导体技术的进步或派生的其它技术而出现置换 LSI 的集成电路化技术，则当然也可使用该技术来进行构成要素的集成化。例如可适用生物技术等。

[0530] 13) 本发明也可以是上述所示的方法。另外，也可以是将这些方法实现为 CPU 的处理的程序，或是由上述计算机程序构成的数字信号。

[0531] (14) 本发明也可将上述计算机程序或上述数字信号记录在计算机可读取的记录媒体中，例如软盘、硬盘、CD-ROM、MO、DVD、DVD-ROM、DVD-RAM、BD(Blu-ray Disc :蓝光光盘)、半导体存储器等中。另外，也可以是记录在这些记录媒体中的上述数字信号。

[0532] (15) 也可以是这些实施方式和变形例的组合。

[0533] 产业上的可利用性

[0534] 本发明的数据处理装置可通过控制可否执行调试处理来实现程序的保护，尤其是作为在保护程序的同时支援开发的装置来说是有用的。

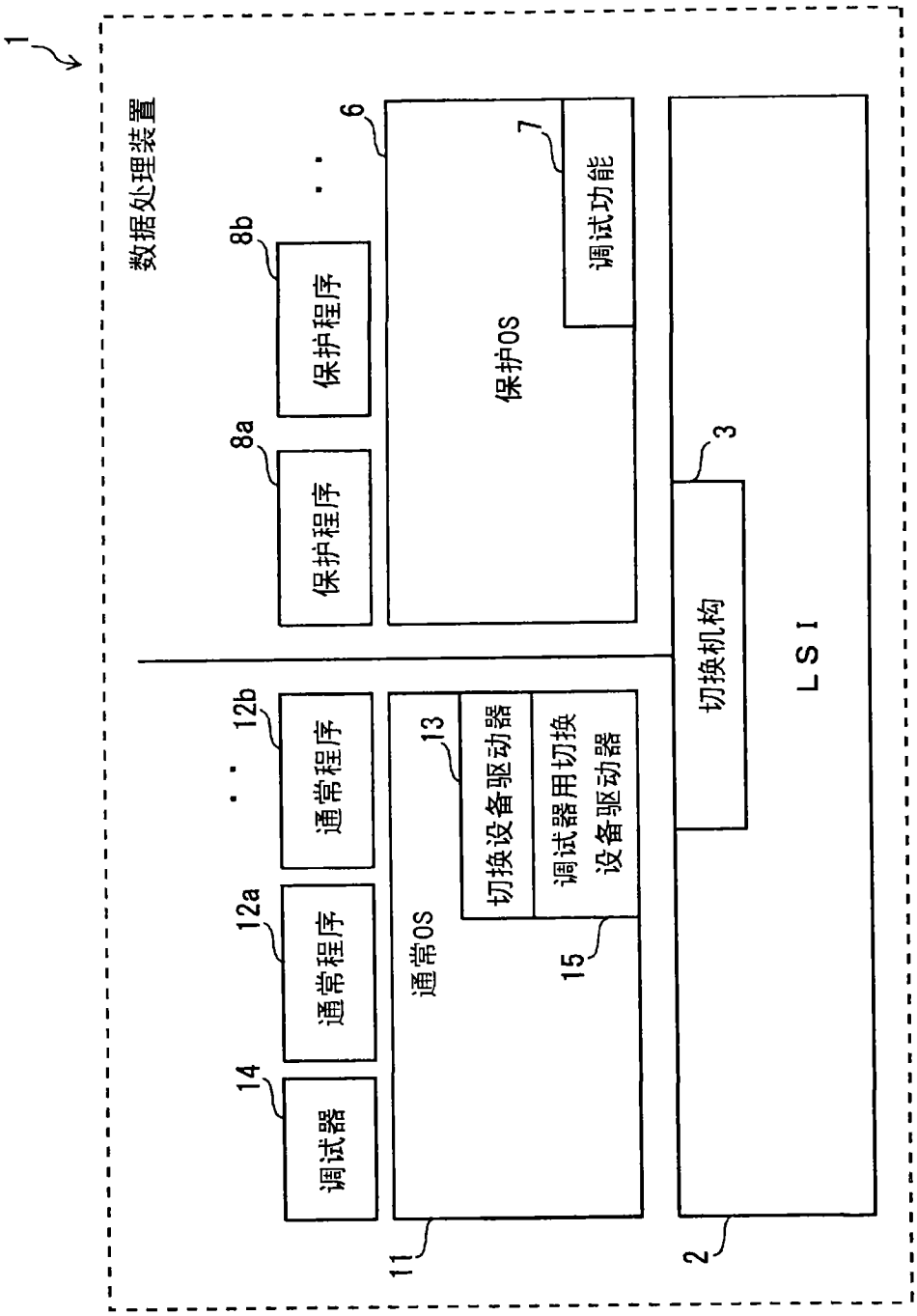

1
数据处理装置
8b
保护程序
8a
保护程序
6
保护OS
7
调试功能
3
切换机构
LSI
12b
通常程序
12a
通常程序
14
调试器
通常OS
13
切换设备驱动器
15
调试器用切换设备驱动器
11
2

图1

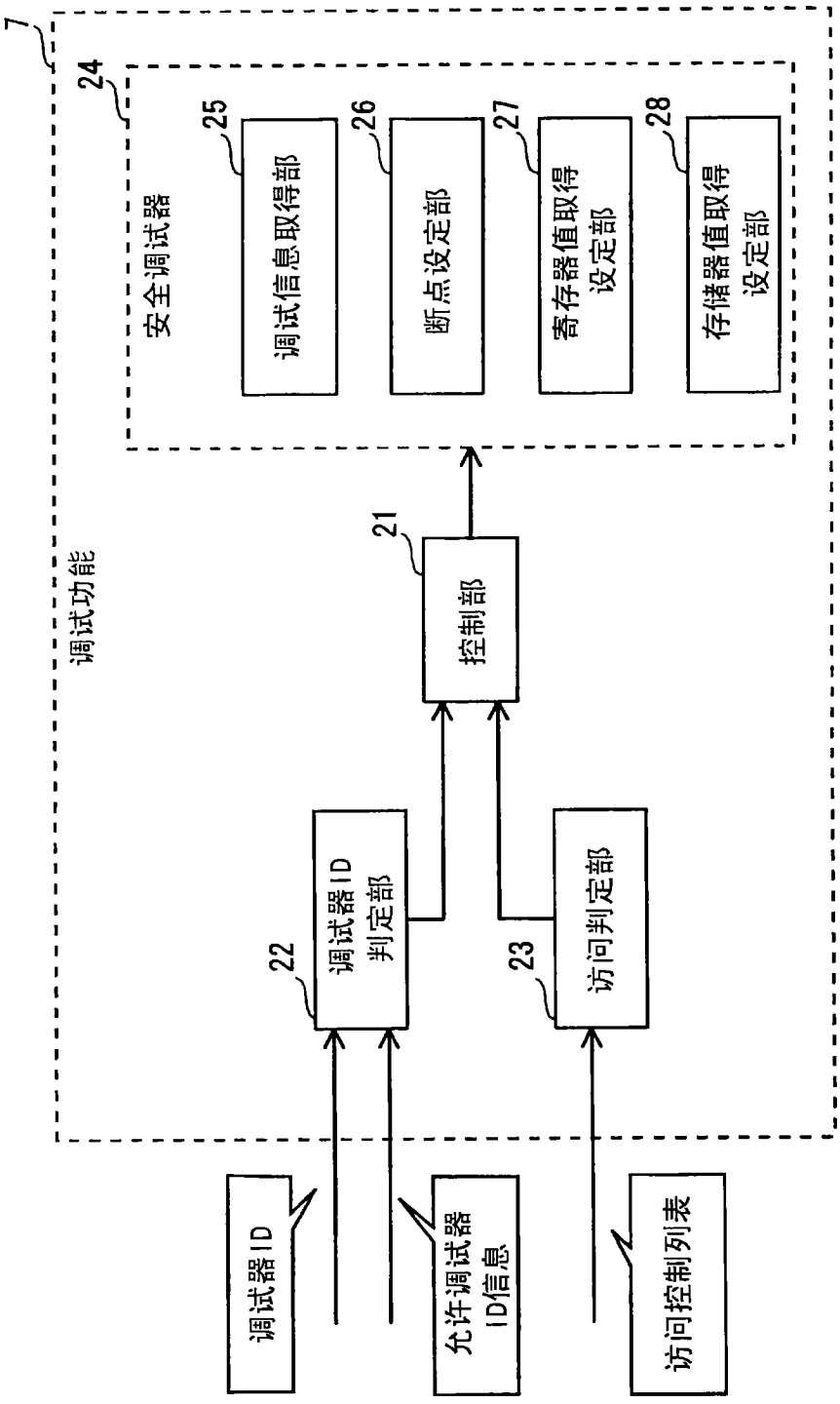

7
24
安全调试器
25
调试信息取得部
26
断点设定部
27
寄存器值取得设定部
28
存储器值取得设定部
调试功能
21
控制部
22
调试器ID判定部
23
访问判定部
调试器ID
允许调试器ID信息
访问控制列表

图2

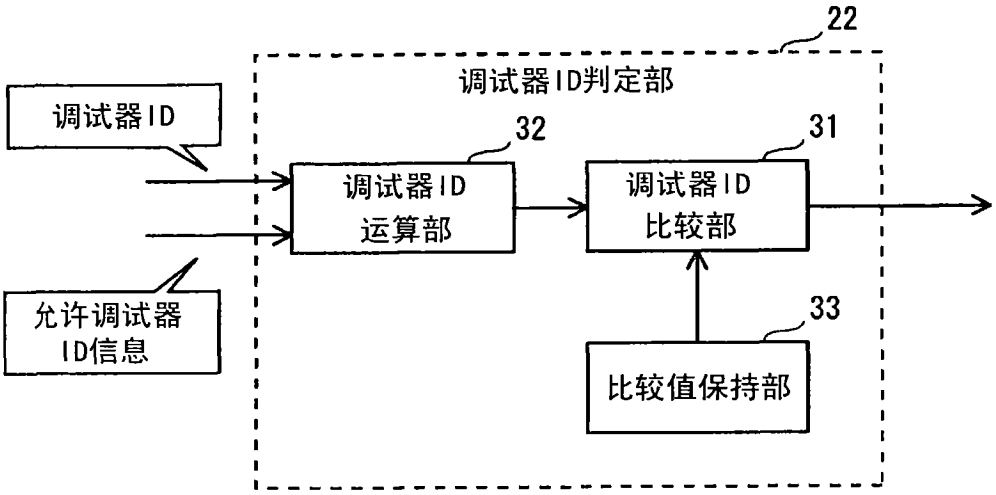

22
调试器ID判定部
调试器ID
32
调试器ID
运算部
31
调试器ID
比较部
允许调试器
ID信息
33
比较值保持部

图 3

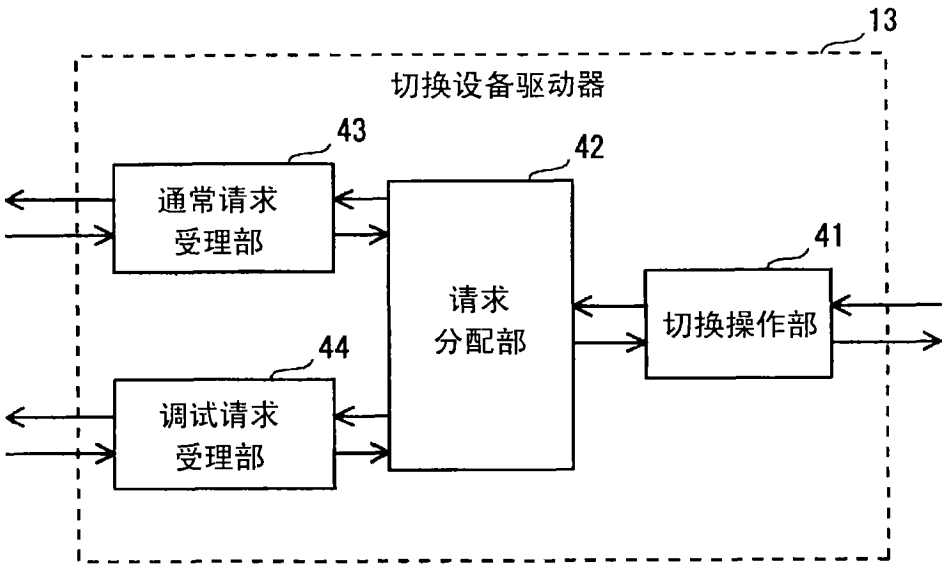

13
切换设备驱动器
43
通常请求
受理部
42
请求
分配部
41
切换操作部
44
调试请求
受理部

图 4

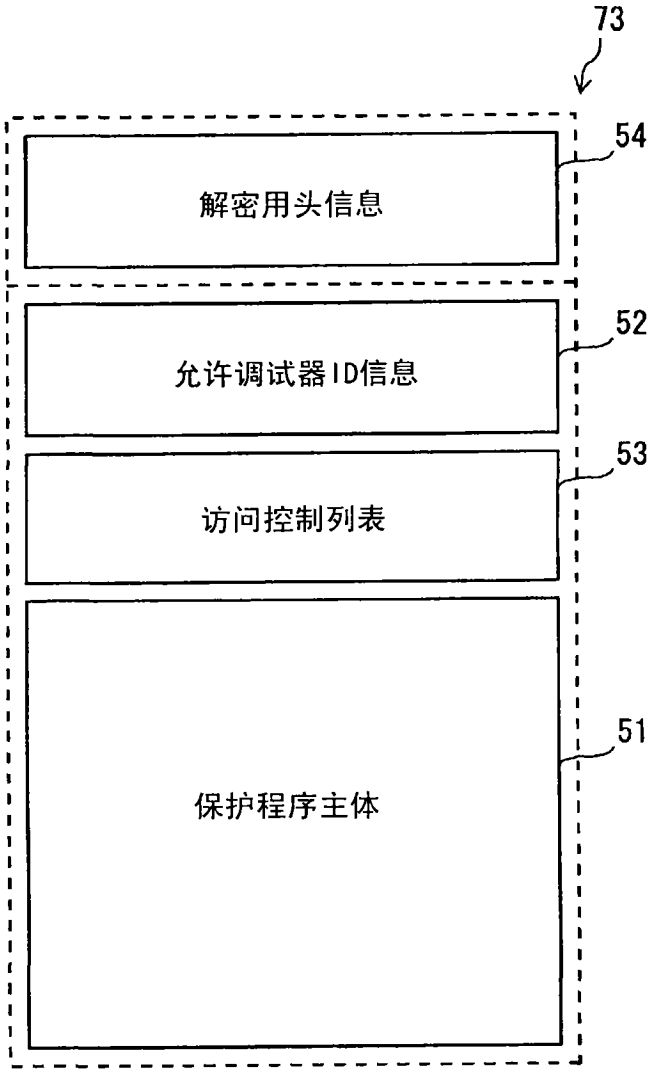
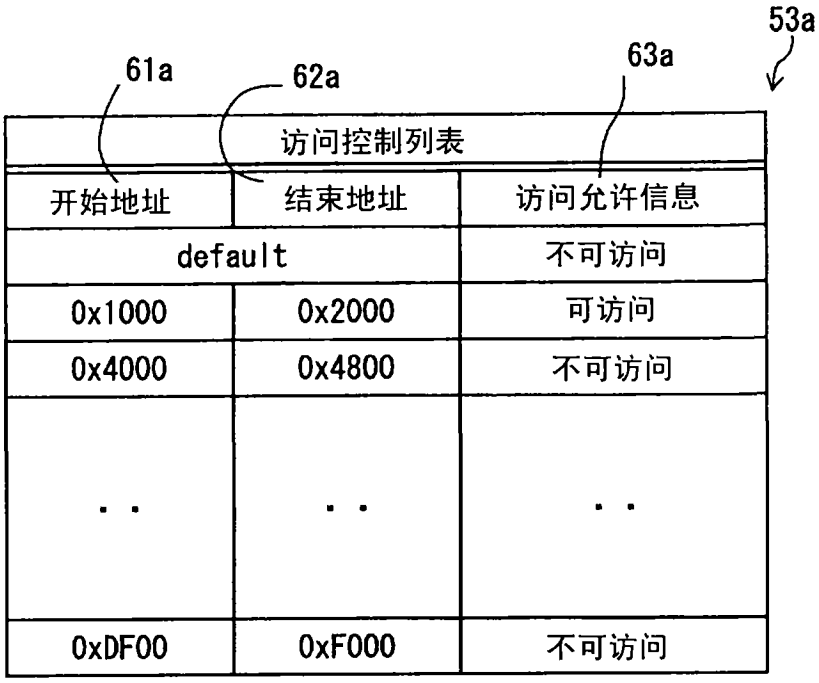


图 5

(a)


53a
61a
62a
63a
访问控制列表
开始地址
结束地址
访问允许信息
default
不可访问
0x1000
0x2000
可访问
0x4000
0x4800
不可访问
0xDF00
0xF000
不可访问

(b)

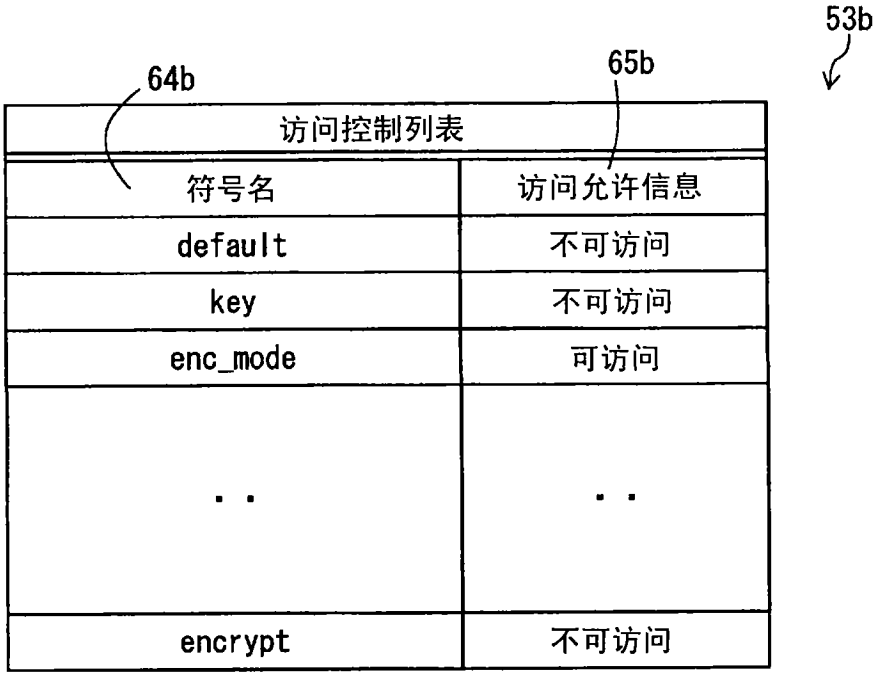

53b
64b
65b
访问控制列表
符号名
访问允许信息
default
不可访问
key
不可访问
enc_mode
可访问
encrypt
不可访问

图 6

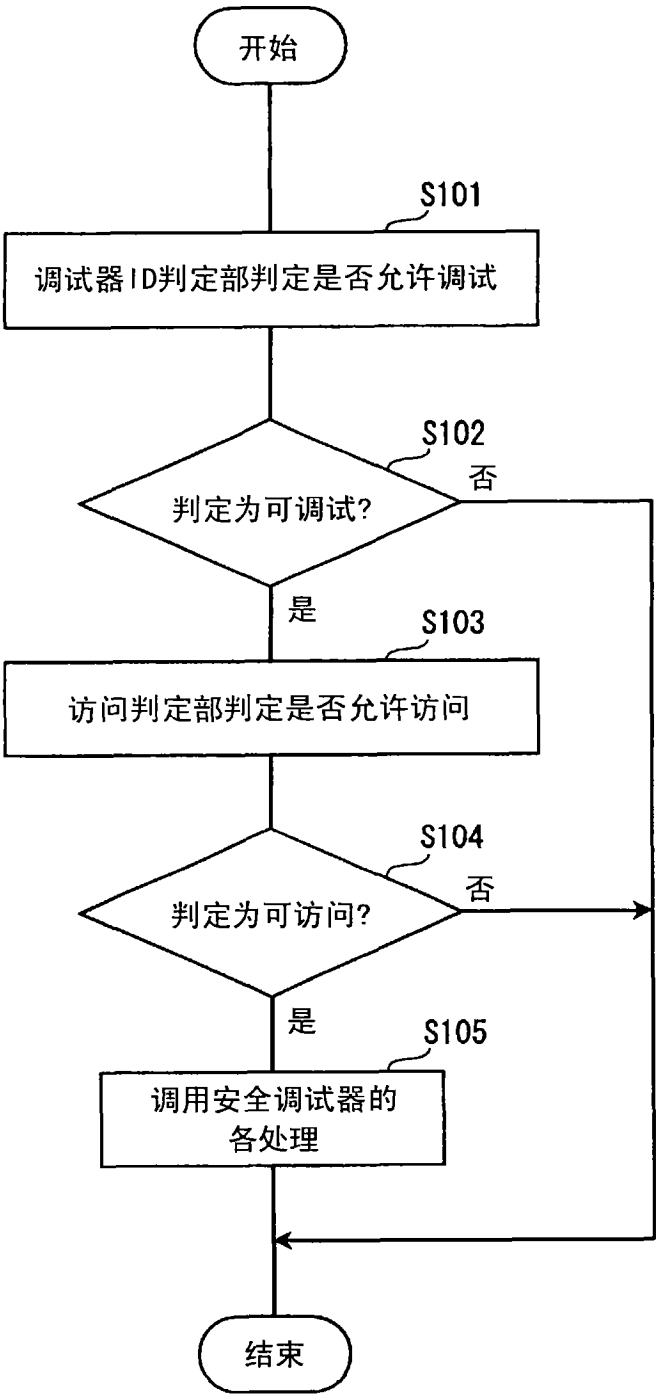

开始
S101
调试器ID判定部判定是否允许调试
S102
判定为可调试?
否
是
S103
访问判定部判定是否允许访问
S104
判定为可访问?
否
是
S105
调用安全调试器的
各处理
结束

图 7

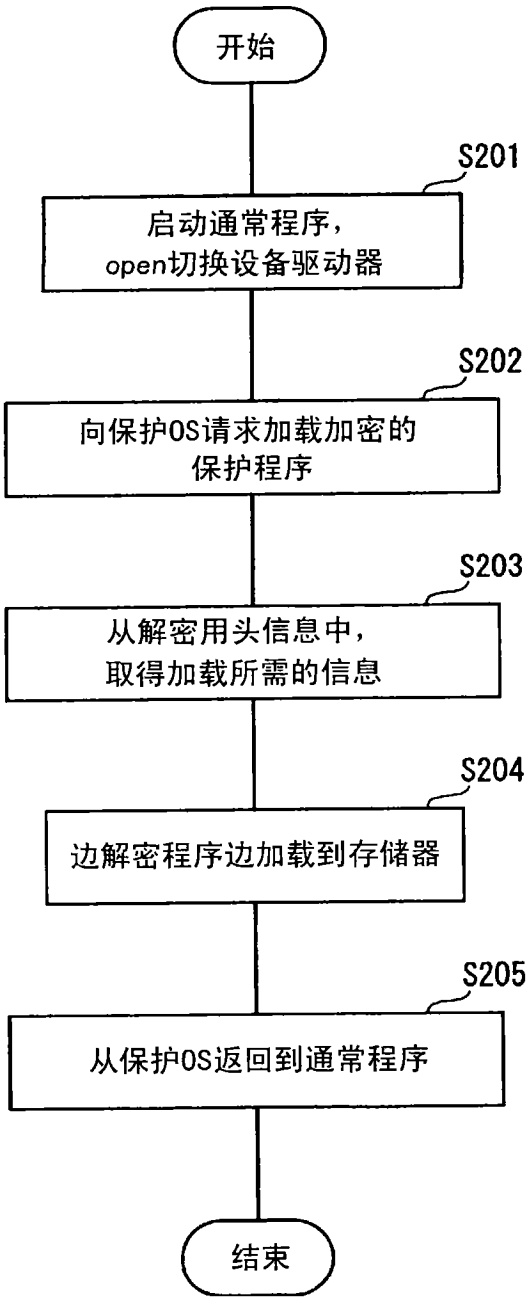

开始
S201
启动通常程序，
open切换设备驱动器
S202
向保护OS请求加载加密的
保护程序
S203
从解密用头信息中，
取得加载所需的信息
S204
边解密程序边加载到存储器
S205
从保护OS返回到通常程序
结束

图 8

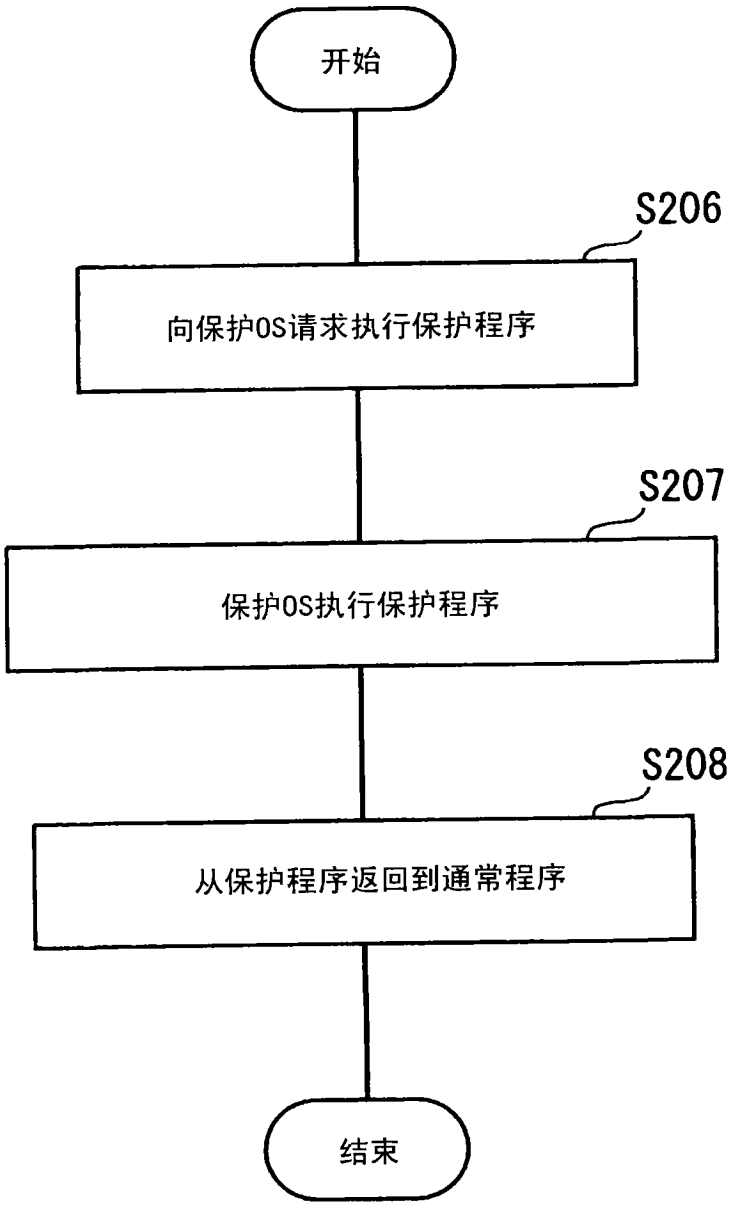

开始
S206
向保护OS请求执行保护程序
S207
保护OS执行保护程序
S208
从保护程序返回到通常程序
结束

图 9

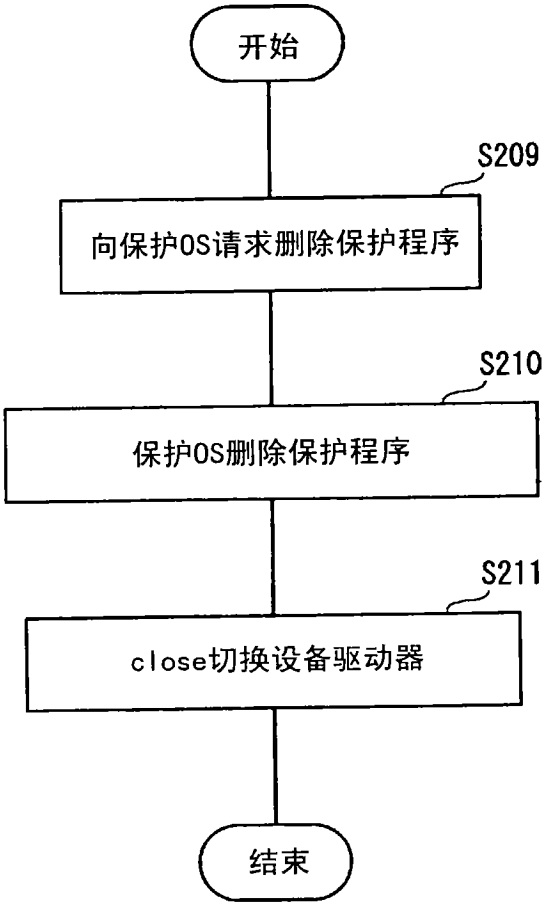

开始
S209
向保护OS请求删除保护程序
S210
保护OS删除保护程序
S211
close切换设备驱动器
结束

图 10

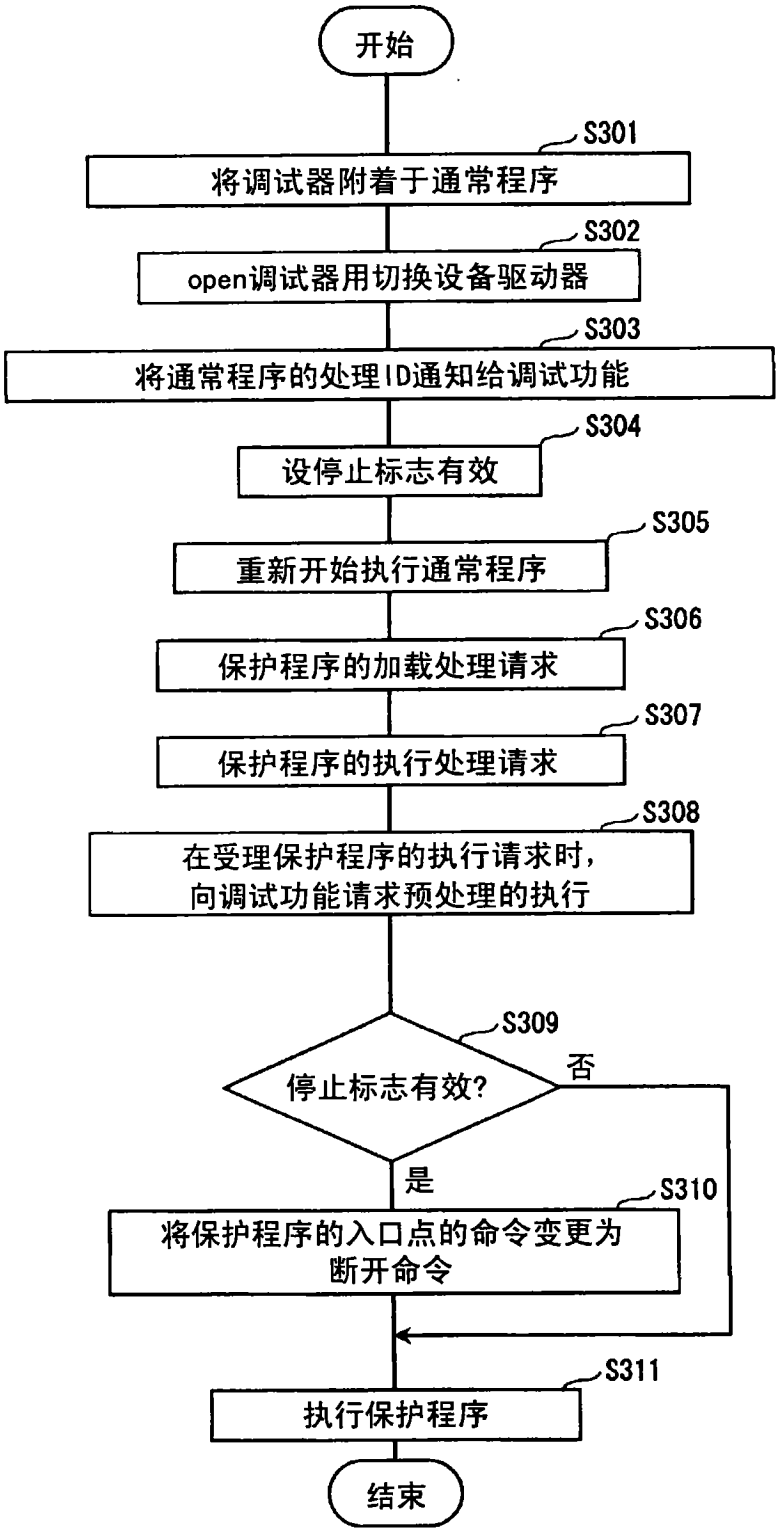

开始
S301
将调试器附着于通常程序
S302
open调试器用切换设备驱动器
S303
将通常程序的处理ID通知给调试功能
S304
设停止标志有效
S305
重新开始执行通常程序
S306
保护程序的加载处理请求
S307
保护程序的执行处理请求
S308
在受理保护程序的执行请求时，
向调试功能请求预处理的执行
S309
停止标志有效?
否
是
S310
将保护程序的入口点的命令变更为
断开命令
S311
执行保护程序
结束

图 11

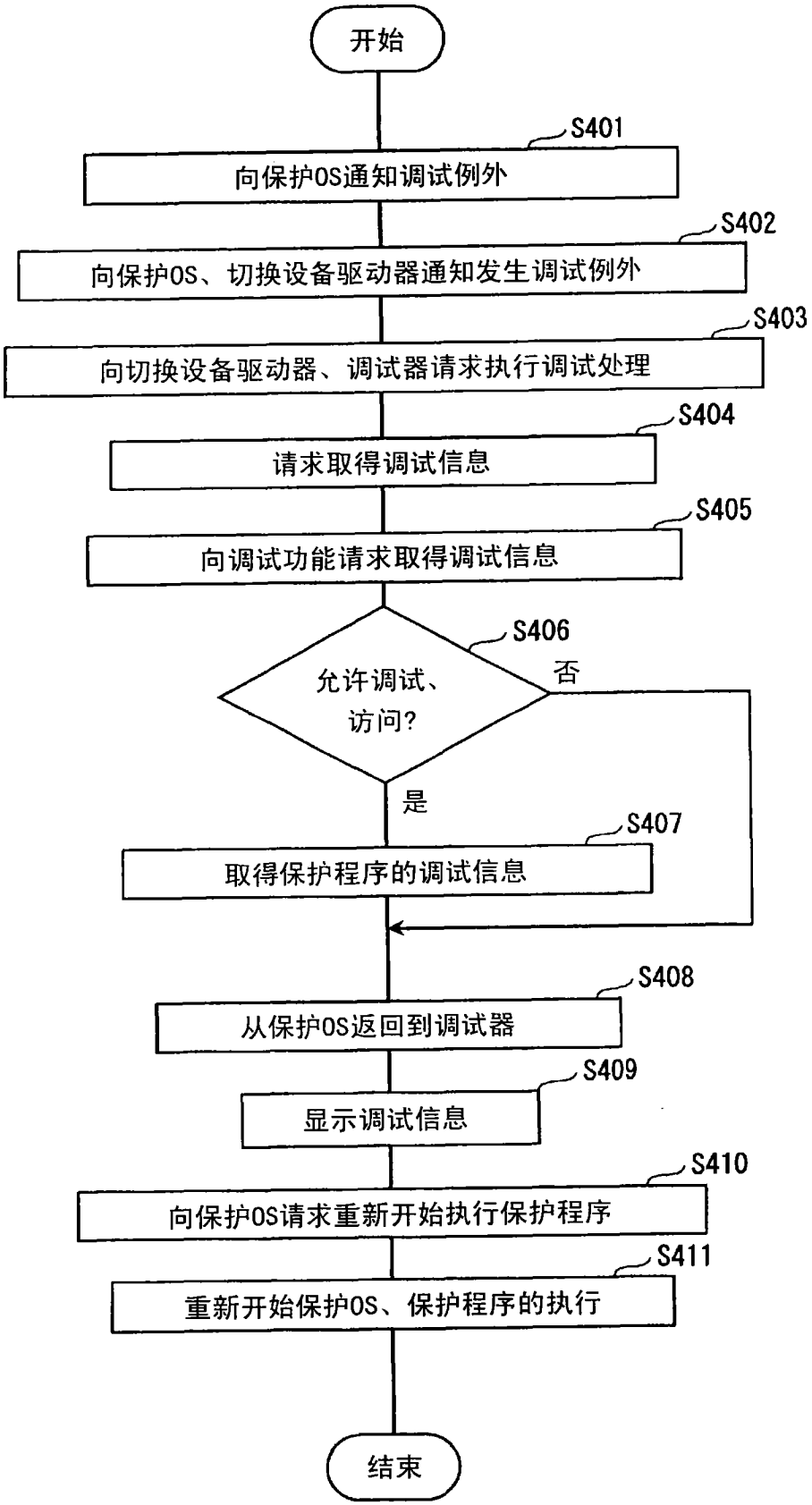

开始
S401
向保护OS通知调试例外
S402
向保护OS、切换设备驱动器通知发生调试例外
S403
向切换设备驱动器、调试器请求执行调试处理
S404
请求取得调试信息
S405
向调试功能请求取得调试信息
S406
允许调试、访问?
否
是
S407
取得保护程序的调试信息
S408
从保护OS返回到调试器
S409
显示调试信息
S410
向保护OS请求重新开始执行保护程序
S411
重新开始保护OS、保护程序的执行
结束

图 12

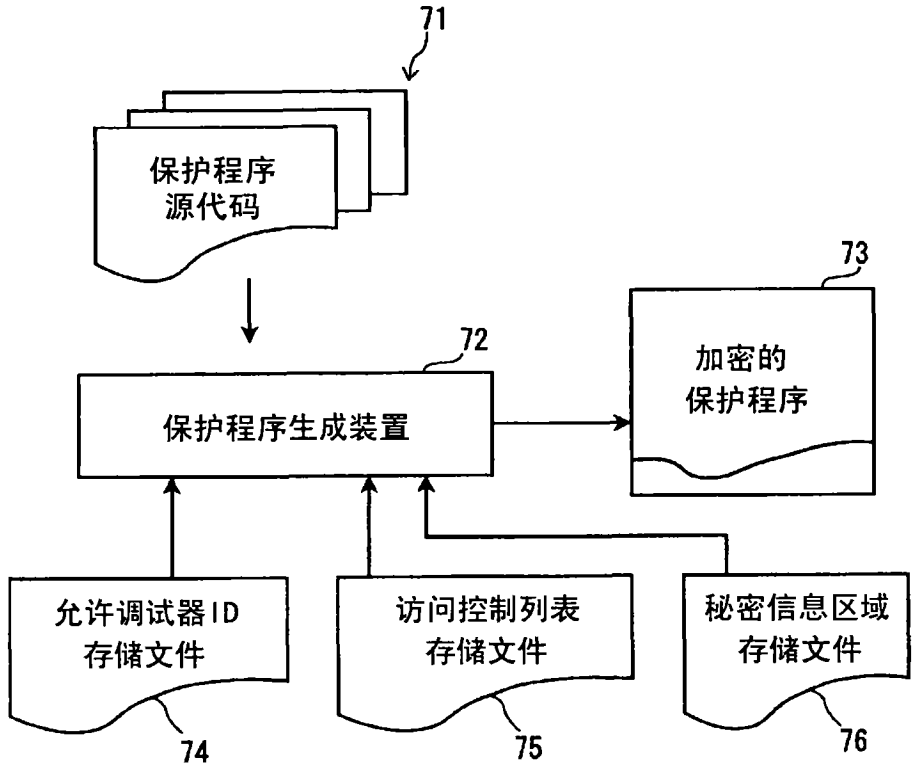

71
保护程序
源代码
73
加密的
保护程序
72
保护程序生成装置
允许调试器ID
存储文件
74
访问控制列表
存储文件
75
秘密信息区域
存储文件
76

图 13

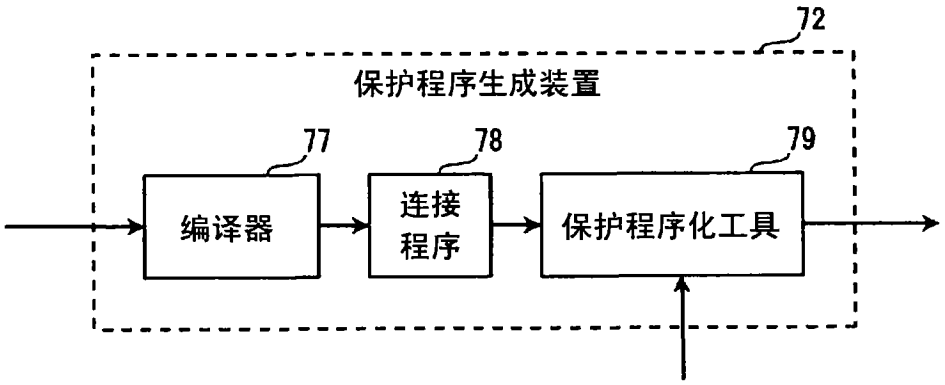

72
保护程序生成装置
77
编译器
78
连接
程序
79
保护程序化工具

图 14

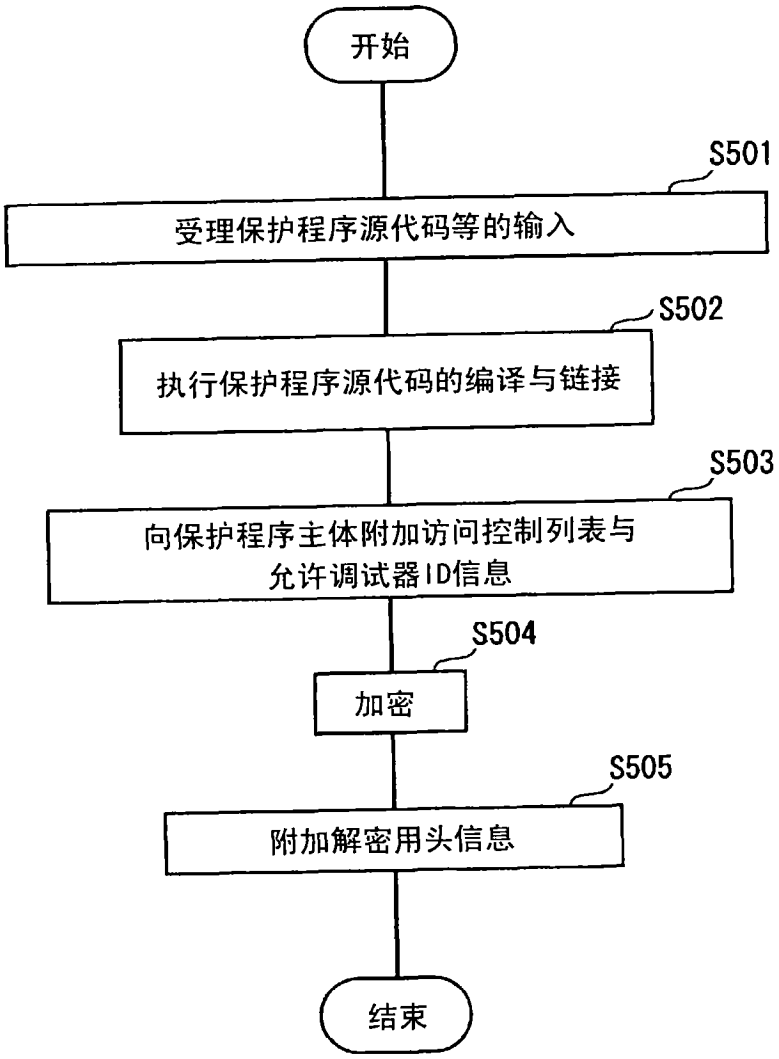

开始
S501
受理保护程序源代码等的输入
S502
执行保护程序源代码的编译与链接
S503
向保护程序主体附加访问控制列表与
允许调试器ID信息
S504
加密
S505
附加解密用头信息
结束

图 15

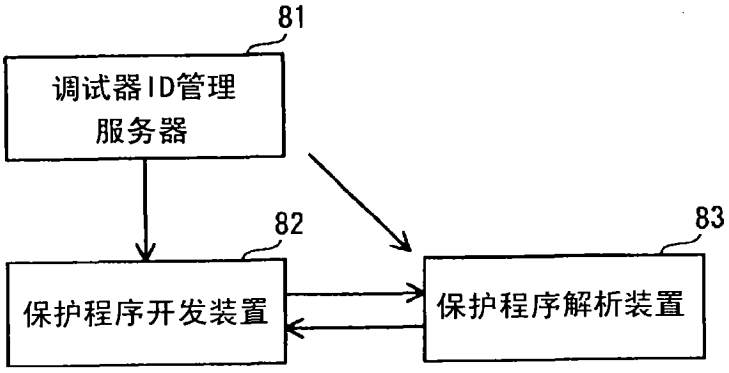

81
调试器ID管理
服务器
82
保护程序开发装置
83
保护程序解析装置

图 16

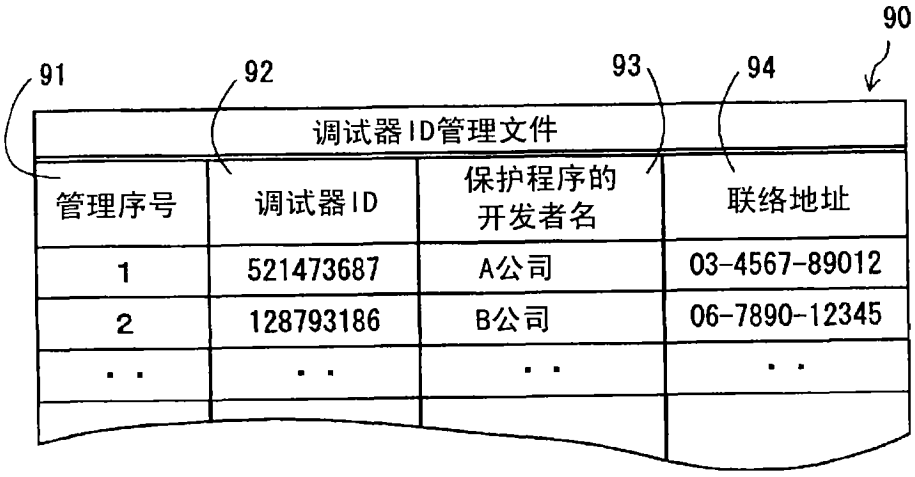

90
91
92
93
94
调试器ID管理文件
管理序号
调试器ID
保护程序的开发者名
联络地址
1
521473687
A公司
03-4567-89012
2
128793186
B公司
06-7890-12345

图 17

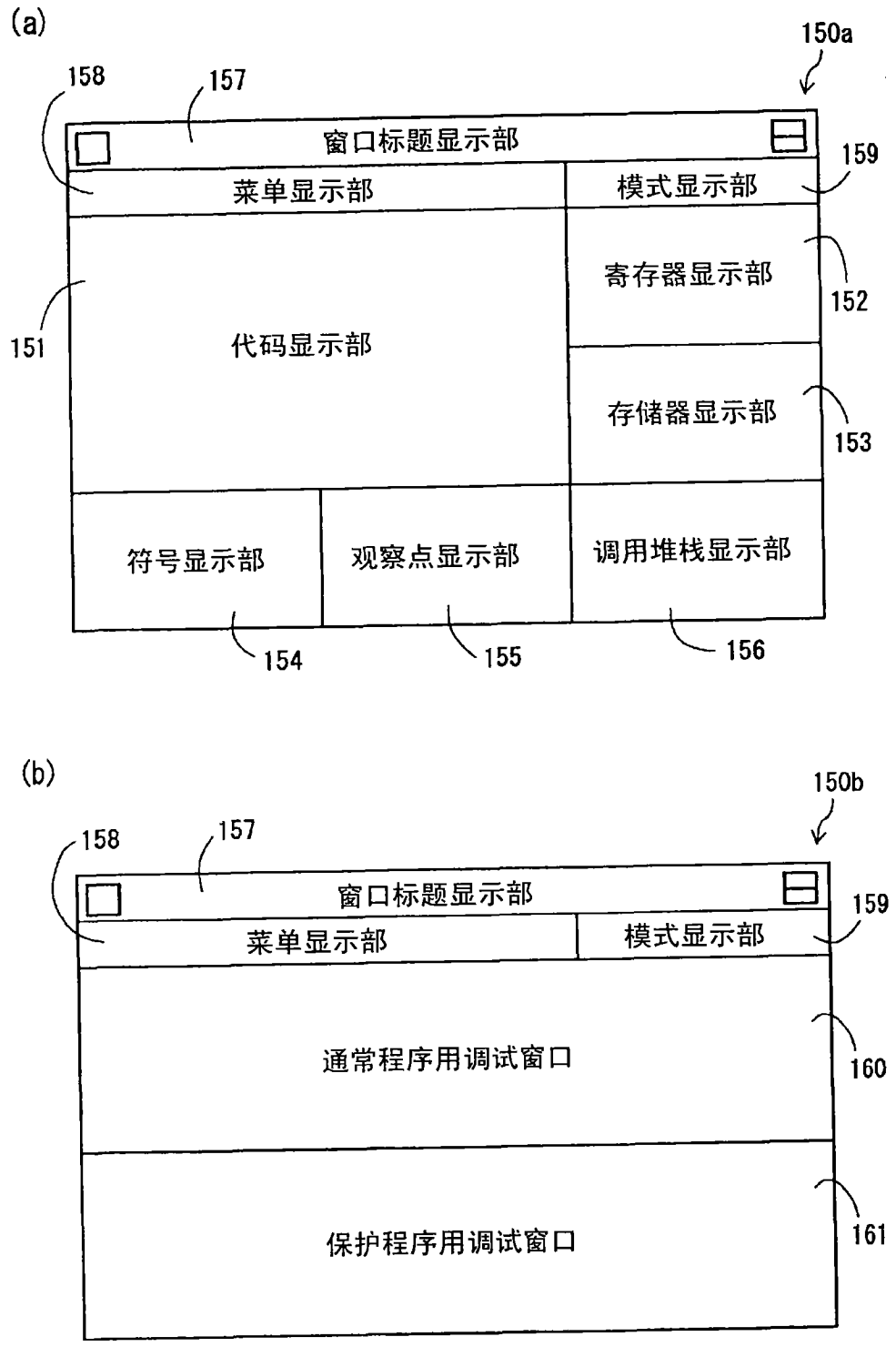

(a)
150a
158
157
窗口标题显示部
159
菜单显示部
模式显示部
寄存器显示部
152
151
代码显示部
存储器显示部
153
符号显示部
观察点显示部
调用堆栈显示部
154
155
156
(b)
150b
158
157
窗口标题显示部
159
菜单显示部
模式显示部
通常程序用调试窗口
160
保护程序用调试窗口
161

图 18

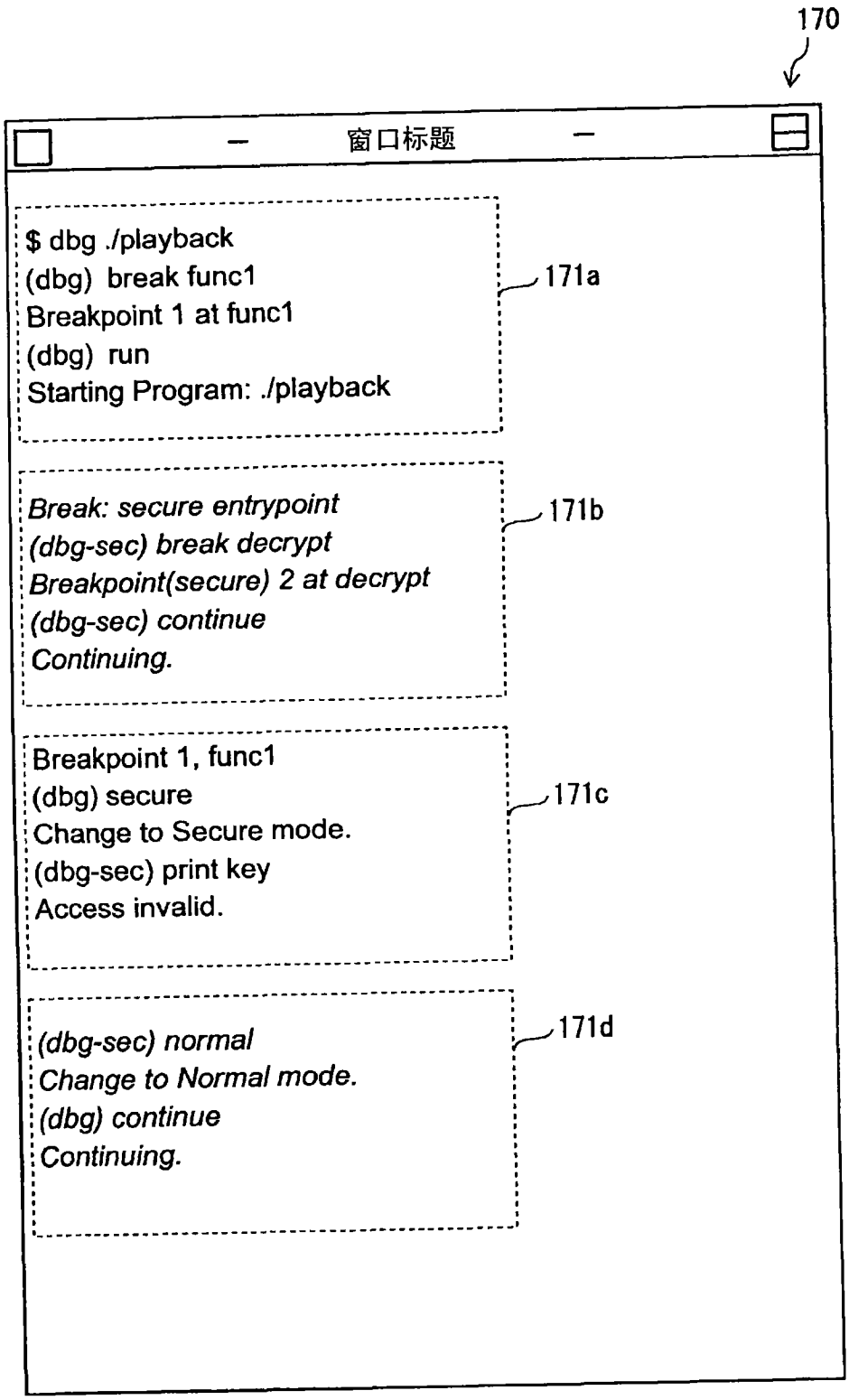

170
窗口标题
$ dbg ./playback
(dbg) break func1
Breakpoint 1 at func1
(dbg) run
Starting Program: ./playback
171a
Break: secure entrypoint
(dbg-sec) break decrypt
Breakpoint(secure) 2 at decrypt
(dbg-sec) continue
Continuing.
171b
Breakpoint 1, func1
(dbg) secure
Change to Secure mode.
(dbg-sec) print key
Access invalid.
171c
(dbg-sec) normal
Change to Normal mode.
(dbg) continue
Continuing.
171d

图 19