



(12)发明专利申请

(10)申请公布号 CN 108874692 A

(43)申请公布日 2018. 11. 23

(21)申请号 201810355605.0

G06F 9/38(2006.01)

(22)申请日 2018.04.19

(30)优先权数据

62/505,318 2017.05.12 US

62/535,460 2017.07.21 US

15/690,931 2017.08.30 US

(71)申请人 三星电子株式会社

地址 韩国京畿道水原市

(72)发明人 爱德华·A·布瑞克鲍姆

阿伦·瑞德哈克施恩

(74)专利代理机构 北京铭硕知识产权代理有限公司

11286

代理人 曾世骁 韩明星

(51)Int. Cl.

G06F 12/0862(2016.01)

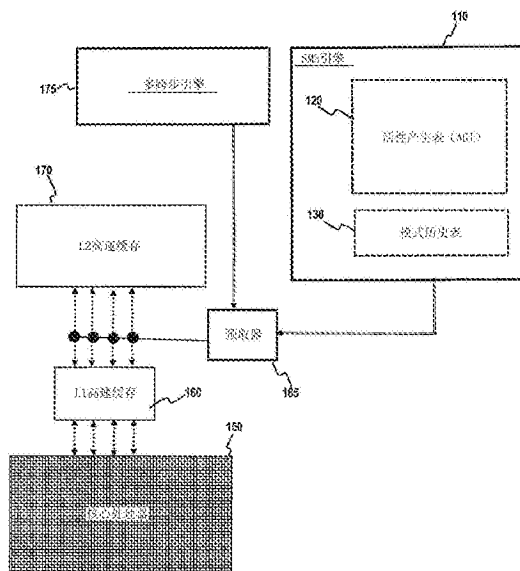
权利要求书3页 说明书15页 附图8页

(54)发明名称

空间存储器流传输置信机制

(57)摘要

一种空间存储器流传输置信机制。描述了用于空间存储器流传输(SMS)预取引擎的设备及其制造方法、系统和方法。一方面,SMS预取引擎包括:模式历史表(PHT),具有每个条目具有偏移列表字段和每偏移置信字段的表,其中,偏移列表字段包括针对从区域内的基础偏移值开始的偏移值的子字段,每偏移置信字段包括针对与每个偏移值相应的每偏移置信级别的子字段。当PHT条目被激活时,通过将当前存取与激活后的PHT条目的偏移列表字段中存储的偏移值进行匹配来更新与PHT条目的偏移列表字段中的每个偏移值相应的每偏移置信值。可至少按照每偏移置信级别向SMS预取引擎提供持续学习。



1. 一种空间存储器流传输SMS预取引擎,包括:

模式历史表PHT,包括至少一个表,其中,所述至少一个表中的每个条目具有偏移列表字段和每偏移置信字段,其中,偏移列表字段包括针对从区域内的基础偏移值开始的偏移值的子字段,每偏移置信字段包括针对与偏移列表字段中的偏移值相应的每偏移置信级别的子字段,

其中,当PHT条目被激活时,通过将当前存取与激活后的PHT条目中存储的偏移值进行匹配来更新与激活后的PHT条目中的偏移值相应的每偏移置信值,

其中,至少按照每偏移置信级别向SMS预取引擎提供持续学习。

2. 如权利要求1所述的SMS预取引擎,其中,当与每偏移置信级别相应的偏移值与相应的存取匹配时,通过提高每偏移置信级别来更新每偏移置信级别,

其中,当与每偏移置信级别相应的偏移值不与两个触发器程序计数器PC之间的任何相应存取匹配时,通过降低每偏移置信级别来更新每偏移置信级别,其中,所述两个触发器程序计数器PC具有开始和结束指令的队列中的空间模式的相同值。

3. 如权利要求1所述的SMS预取引擎,其中,每偏移置信级别被用于从PHT条目的偏移列表字段消除一个或更多个偏移值。

4. 如权利要求3所述的SMS预取引擎,其中,一个或更多个预设阈值被应用于每偏移置信级别以确定是否从PHT条目的偏移列表字段消除所述一个或更多个偏移值。

5. 如权利要求1所述的SMS预取引擎,其中,当与偏移列表字段内的多个偏移值相应的存储器位置在单个高速缓存行内时,SMS预取引擎仅对所述单个高速缓存行进行一次预取。

6. 如权利要求1所述的SMS预取引擎,其中,每个PHT条目还包括总体置信字段,其中,总体置信字段的值代表通过偏移列表字段中存储的偏移值所做出的全部空间模式中的置信级别,

其中,至少按照总体置信字段向SMS预取引擎提供持续学习。

7. 如权利要求6所述的SMS预取引擎,其中,PHT条目中的偏移列表字段中的偏移值的每偏移置信级别的每次提高包括:对该PHT条目的总体置信字段中的总体置信级别的提高的确认,

其中,PHT条目中的偏移列表字段中的偏移值的每偏移置信级别的每次降低导致该PHT条目的总体置信字段中的总体置信级别的降低。

8. 如权利要求1所述的SMS预取引擎,其中,每个PHT条目还包括:包括所述区域内的基础偏移值的基础偏移字段,

其中,基础偏移值提供字节粒度。

9. 一种用于空间存储器流传输SMS预取引擎的方法,包括:

维护模式历史表PHT,其中,模式历史表PHT包括至少一个表,其中,所述至少一个表中的每个条目具有偏移列表字段和每偏移置信字段,其中,偏移列表字段包括针对从区域内的基础偏移值开始的偏移值的子字段,每偏移置信字段包括针对与偏移值相应的每偏移置信级别的子字段;

当PHT条目被激活时,通过将当前存取与激活后的PHT条目中存储的偏移值进行匹配来更新与激活后的PHT条目中的偏移值相应的每偏移置信值,

其中,至少按照每偏移置信级别向SMS预取引擎提供持续学习。

10. 如权利要求9所述的方法,其中,更新每偏移置信级别的步骤包括:

当与每偏移置信级别相应的偏移值与相应存取匹配时,提高每偏移置信级别,

当与每偏移置信级别相应的偏移值不与两个触发器程序计数器PC之间的任何相应存取匹配时,降低每偏移置信级别,其中,所述两个触发器程序计数器PC具有开始和结束指令的队列中的空间模式的相同值。

11. 如权利要求9所述的方法,还包括:根据每偏移置信级别从PHT条目的偏移列表字段消除一个或更多个偏移值。

12. 如权利要求11所述的方法,其中,从PHT条目的偏移列表字段消除一个或更多个偏移值的步骤包括:

将一个或更多个预设阈值应用于每偏移置信级别以确定是否从PHT条目的偏移列表字段消除所述一个或更多个偏移值。

13. 如权利要求9所述的方法,其中,当与偏移列表字段内的多个偏移值相应的存储器位置在单个高速缓存行内时,SMS预取引擎仅对所述单个高速缓存行进行一次预取。

14. 如权利要求9所述的方法,其中,每个PHT条目还包括总体置信字段,其中,总体置信字段的值代表通过偏移列表字段中存储的偏移值所做出的全部空间模式中的置信级别,

其中,至少按照总体置信字段向SMS预取引擎提供持续学习。

15. 如权利要求14所述的方法,其中,PHT条目中的偏移列表字段中的偏移值的每偏移置信级别的每次提高包括:对该PHT条目的总体置信字段中的总体置信级别的提高的确认,

其中,PHT条目中的偏移列表字段中的偏移值的每偏移置信级别的每次降低导致该PHT条目的总体置信字段中的总体置信级别的降低。

16. 如权利要求9所述的方法,其中,每个PHT条目还包括:包括所述区域内的基础偏移值的基础偏移字段,

其中,基础偏移值提供字节粒度。

17. 一种设备,包括:

一个或更多个非暂时性计算机可读介质;

至少一个处理器,包括空间存储器流传输SMS预取引擎的至少一部分,其中,当所述至少一个处理器执行所述一个或更多个非暂时性计算机可读介质上存储的指令时,SMS预取引擎的所述至少一部分执行以下步骤:

维护模式历史表PHT,其中,PHT包括至少一个表,其中,所述至少一个表中的每个条目具有偏移列表字段和每偏移置信字段,其中,偏移列表字段包括针对从区域内的基础偏移值开始的偏移值的子字段,每偏移置信字段包括针对与偏移值相应的每偏移置信级别的子字段;

当PHT条目被激活时,通过将当前存取与激活后的PHT条目中存储的偏移值进行匹配来更新与激活后的PHT条目中的偏移值相应的每偏移置信值。

18. 一种方法,包括:

制造芯片组,其中,芯片组包括:

至少一个处理器,其中,当所述至少一个处理器执行一个或更多个非暂时性计算机可读介质上存储的指令时执行以下步骤:

维护用于空间存储器流传输SMS预取引擎的模式历史表PHT,其中,PHT包括至少一个

表,其中,所述至少一个表中的每个条目具有偏移列表字段和每偏移置信字段,其中,偏移列表字段包括针对从区域内的基础偏移值开始的偏移值的子字段,每偏移置信字段包括针对与偏移值相应的每偏移置信级别的子字段;

当PHT条目被激活时,通过将当前存取与激活后的PHT条目中存储的偏移值进行匹配来更新与激活后的PHT条目中的偏移值相应的每偏移置信值;

所述一个或更多个非暂时性计算机可读介质,存储所述指令。

19. 一种测试设备的方法,包括:

检测所述设备是否具有至少一个处理器,其中,所述至少一个处理器在执行一个或更多个非暂时性计算机可读介质上存储的指令时执行以下步骤:

维护用于空间存储器流传输SMS预取引擎的模式历史表PHT,其中,PHT包括至少一个表,其中,所述至少一个表中的每个条目具有偏移列表字段和每偏移置信字段,其中,偏移列表字段包括针对从区域内的基础偏移值开始的偏移值的子字段,每偏移置信字段包括针对与每个偏移值相应的每偏移置信级别的子字段;

当PHT条目被激活时,通过将当前存取与激活后的PHT条目中存储的偏移值进行匹配来更新与激活后的PHT条目中的每个偏移值相应的每偏移置信值;

测试所述设备是否具有存储所述指令的所述一个或更多个非暂时性计算机可读介质。

空间存储器流传输置信机制

[0001] 本申请要求于2017年5月12日提交的序列号为62/505,318的美国临时专利申请,于2017年7月21日提交的序列号为62/535,460的美国临时专利申请的优先权,以及于2017年8月30日提交的序列号为15/690,931的美国专利申请的优先权,所述三项申请的全部内容通过引用合并于此。

技术领域

[0002] 本公开总体上涉及处理器存储器管理,更具体地,涉及空间存储器流传输置信机制。

背景技术

[0003] 计算机处理器通过在实际需要之前将指令或数据从它们原来的较慢的存储器(即,具有较慢的读/写时间)中的寄存器取出至较快的本地存储器(即,具有较快的读/写时间并通常位于指令/数据管道附近)来执行高速缓存预取从而提高执行效率。大多数现代计算机处理器具有一个或多个快速的本地高速缓存存储器,其中,预取的数据和/或指令直到被需要前都保存在所述一个或多个快速的本地高速缓存存储器中。

[0004] 然而,预取是通过猜测来进行的。更严格地说,预取基于例如过去的性能、概率模型、和/或算法通过处理引擎利用当前对数据/指令的一系列存储器需求来预测处理引擎接下来将需要什么数据/指令。因此,不准确的预取是有问题的,因为必须将错误的数据从本地更快的存储器移除,并且必须存取正确的数据并将其移至本地更快的存储器中。不准确的预取会不必要地增加功耗,造成系统拥塞(至少是由错误数据与正确数据的额外移动/交换造成的),并且明显地污染和破坏高速缓存的正常功能。

[0005] 存在不同的预取方法,通常以它们预取数据/指令的模式(诸如顺序预取和跨步预取)来区分。尽管有些过于简化,但顺序预取可被认为是对连续连续存储器块进行预取,而跨步预取可被认为是向前跳过(或“跨步”s数量的块)以预取存储器块。

[0006] 还有一种涉及跨步的更专业的方案,叫做空间存储器流传输。见,例如Somogyi等人的《Spatial Memory Streaming》,发表于第33届Int'l Symposium on Computer Architecture (ISCA 2006),页码252-263(在下文中,“Somogyi 2006”);和Somogyi等人的《Spatial Memory Streaming》,发表于Journal of Instruction-Level Parallelism 13 (2011),页码1-26(在下文中,“Somogyi 2011”),所述两者通过引用它们的全部合并于此。在空间存储器流传输(SMS)中,检测并利用代码和存取模式之间的强相关性,以预测具有类似的相对间隔(“空间相对性”)的组中的存储器存取模式。

[0007] 然而,SMS方案存在各种缺点。SMS不能处理针对高速缓存之间的行边界的模式的移位对齐。此外,通常用于空间模式的空间位向量迫使每个存取的粒度更大,并且不能跟踪时间顺序。SMS还缺乏鲁棒性置信机制,并且不是动态适应的,即,SMS不能适应程序阶段改变(诸如当动态分支行为改变偏移模式时)。这些缺点导致了覆盖率和准确性的降低,以及时效性的丧失,从而降低了性能并增加了功耗。

发明内容

[0008] 因此,做出本公开以至少解决这里描述的问题和/或缺点,并至少提供下面描述的优点。

[0009] 根据本公开的一方面,提供了一种空间存储器流传输 (SMS) 预取引擎,包括:模式历史表 (PHT),包括至少一个表,其中,所述至少一个表中的每个条目具有偏移列表字段和每偏移置信字段,其中,偏移列表字段包括针对从区域内的基础偏移值开始的偏移值的子字段,每偏移置信字段包括针对与每个偏移值相应的每偏移置信级别的子字段,其中,当 PHT 条目被激活时,通过将当前存取与激活后的 PHT 条目中存储的偏移值进行匹配来更新与激活后的 PHT 条目中的每个偏移值相应的每偏移置信值,其中,至少按照每偏移置信级别向 SMS 预取引擎提供持续学习。

[0010] 根据本公开的一方面,提供了一种用于空间存储器流传输 (SMS) 预取引擎的方法,包括:维护模式历史表 (PHT),其中,PHT 包括至少一个表,其中,所述至少一个表中的每个条目具有偏移列表字段和每偏移置信字段,其中,偏移列表字段包括针对从区域内的基础偏移值开始的偏移值的子字段,每偏移置信字段包括针对与每个偏移值相应的每偏移置信级别的子字段;当 PHT 条目被激活时,通过将当前存取与激活后的 PHT 条目中存储的偏移值进行匹配来更新与激活后的 PHT 条目中的每个偏移值相应的每偏移置信值,其中,至少按照每偏移置信级别向 SMS 预取引擎提供持续学习。

[0011] 根据本公开的一方面,提供了一种设备,包括:一个或多个非暂时性计算机可读介质;以及至少一个处理器,其中,当所述至少一个处理器执行一个或多个非暂时性计算机可读介质上存储的指令时作为与空间存储器流传输 (SMS) 预取引擎的至少一部分来执行以下步骤:维护模式历史表 (PHT),其中,PHT 包括至少一个表,其中,所述至少一个表中的每个条目具有偏移列表字段和每偏移置信字段,其中,偏移列表字段包括针对从区域内的基础偏移值开始的偏移值的子字段,每偏移置信字段包括针对与每个偏移值相应的每偏移置信级别的子字段;当 PHT 条目被激活时,通过将当前存取与激活后的 PHT 条目中存储的偏移值进行匹配来更新与激活后的 PHT 条目中的每个偏移值相应的每偏移置信值。可至少按照每偏移置信级别向 SMS 预取引擎提供持续学习。

[0012] 根据本公开的一方面,提供了一种制造芯片组的方法,其中,所述至少一个芯片组包括:至少一个处理器,当所述至少一个处理器执行一个或多个非暂时性计算机可读介质上存储的指令时执行以下步骤:维护用于空间存储器流传输 (SMS) 预取引擎的模式历史表 (PHT),其中,PHT 包括至少一个表,其中,所述至少一个表中的每个条目具有偏移列表字段和每偏移置信字段,其中,偏移列表字段包括针对从区域内的基础偏移值开始的偏移值的子字段,每偏移置信字段包括针对与每个偏移值相应的每偏移置信级别的子字段;当 PHT 条目被激活时,通过将当前存取与激活后的 PHT 条目中存储的偏移值进行匹配来更新与激活后的 PHT 条目中的每个偏移值相应的每偏移置信值。可至少按照每偏移置信级别向 SMS 预取引擎提供持续学习。

[0013] 根据本公开的一方面,提供了一种测试设备的方法,包括:测试所述设备是否具有至少一个处理器,其中,所述至少一个处理器在执行一个或多个非暂时性计算机可读介质上存储的指令时执行以下步骤:维护用于空间存储器流传输 (SMS) 预取引擎的模式历史

表 (PHT), 其中, PHT 包括至少一个表, 其中, 所述至少一个表中的每个条目具有偏移列表字段和每偏移置信字段, 其中, 偏移列表字段包括针对从区域内的基础偏移值开始的偏移值的子字段, 每偏移置信字段包括针对与每个偏移值相应的每偏移置信级别的子字段; 当 PHT 条目被激活时, 通过将当前存取与激活后的 PHT 条目中存储的偏移值进行匹配来更新与激活后的 PHT 条目中的每个偏移值相应的每偏移置信值; 测试所述设备是否具有存储所述指令的所述一个或更多个非暂时性计算机可读介质。可至少按照每偏移置信级别向 SMS 预取引擎提供持续学习。

附图说明

[0014] 从结合附图的以下详细描述, 本公开的特定实施例的上述和其他方面、特征和优点将更加清楚, 其中:

[0015] 图1示出根据一个实施例的本系统的相关组件的示例性框图;

[0016] 图2是根据本公开的实施例的L2高速缓存170内的区域的示例性视图;

[0017] 图3是根据本公开的实施例的模式历史表 (PHT) 130 中的条目 300 的示例性视图;

[0018] 图4示出根据本公开的实施例的 SMS 引擎的相关组件的示例性框图, 其中, 框图示出了信息在其中的移动;

[0019] 图5是根据本公开的实施例的用于训练/学习以及存储发现的空间模式的示例性方法的修改的流程图;

[0020] 图6A至图6C提供根据本公开的实施例的基于过滤表状态和训练表状态的 SMS 训练结果的部分流程图格式的示例;

[0021] 图7示出根据本公开的实施例的 PHT 条目是如何从一系列存取或高速缓存行加载指令被创建或在重新排序队列 (ROQ) 中进行存取的示例;

[0022] 图8示出根据本公开的实施例的当在 ROQ 中识别出存储的 PHT 条目的程序计数器 (PC) 触发器时, PHT 条目是如何被用于从由它的偏移列表字段中存储的空间模式所指示的存储器位置预取数据/指令的示例;

[0023] 图9示出根据本公开的实施例的本设备的示例性框图;

[0024] 图10示出根据本公开的实施例的用于制造和测试本设备的示例性流程图。

具体实施方式

[0025] 在下文中, 参照附图详细地描述本公开的实施例。应注意的是, 尽管相同的元件被示出在不同的附图中, 但是相同的参考标号指示相同的元件。在下面的描述中, 提供诸如详细的构造和组件的详细细节仅为了助于全面地理解本公开的实施例。因此, 本领域的技术人员应该清楚的是, 在不脱离本公开的范围的情况下可对这里描述的实施例做出各种改变和修改。此外, 为了清楚和简洁, 省略公知的功能和构造的描述。下面描述的术语是考虑本公开中的功能而定义的术语, 并可根用户、用户的意图或习惯而有所不同。因此, 术语的定义应基于贯穿说明书的内容来确定。

[0026] 本公开可具有参照附图在下文详细描述实施例中的各种修改和各种实施例。然而, 应理解的是本公开不不限于所述实施例, 而是包括本公开的范围内的所有修改、等同物和替换物。

[0027] 尽管包括诸如第一和第二的序数词的术语可被用于描述各种元件,但是结构元件不受这些术语的限制。这些术语仅被用于将一个元件与另一个元件进行区分。例如,在不脱离本公开的范围的情况下,第一结构元件可被称为第二结构元件。类似地,第二结构元件也可被称为第一结构元件。如在这里使用的,术语“和/或”包括一个或更多个相关项的任意组合和所有组合。

[0028] 这里的术语仅用于描述本公开的各种实施例,而不意图限制本公开。除非上下文另外清楚地指示,否则单数形式意图包括复数形式。在本公开中,应该理解的是术语“包括”或“具有”指示特征、数字、步骤、操作、结构元件、部件或它们的组合的存在,并且不排除一个或更多个其他特征、数字、步骤、操作、结构元件、部件或它们的组合的存在或添加的可能性。

[0029] 除非另有定义,否则这里使用的所有术语具有与本公开所属领域的技术人员所理解的含义相同的含义。除非在本公开中清楚地定义,否则诸如在常用词典中定义的术语将被解释为具有与相关领域中的上下文含义相同的含义,并且将不被解释为具有理想化或过度正式的含义。

[0030] 各种实施例可包括一个或更多个元件。元件可包括被安排执行特定操作的任何结构。尽管通过示例的方式使用特定布置中的有限数量的元件来描述实施例,但是实施例可根据针对给定实施方式的需求而在供替换的布置中包括更多或更少元件。值得注意的是,任何对“一个实施例”或“实施例”的引用都意味着与该实施例相关描述的特定特征、结构或特性被包括在至少一个实施例中。在本说明书中的各种位置出现的短语“一个实施例”(或“实施例”)不必指示相同的实施例。

[0031] 本公开的实施例提供了用于更新和控制预取引擎的系统、方法和设备。根据一个实施例,本公开提供了适应于改变的程序行为的持续学习。

[0032] 根据一个实施例,本公开提供了一种捕捉时间顺序以及空间顺序的偏移列表,使得时效性得到提高。根据一个实施例,本公开使用偏移的列表,而不是在区域中的每个高速缓存行存储一个位。在另一实施例中,偏移被存储为空间位向量。

[0033] 根据一个实施例,本公开提供了字节粒度,其中,所述字节粒度使得可将对前一行的存取或对后一高速缓存行的存取(高速缓存加载指令)进行区分。根据本公开的实施例可将起始偏移存储器位置存储到字节。在一个实施例中,存取所需的尺寸也存储在附加字段中。

[0034] 根据各种实施例,本公开提供了每偏移置信,其中,每偏移置信使得单独的偏移针对精确度而被评估。因此,根据本公开的系统适应于例如由于分支而导致的改变的程序阶段行为。根据本公开的一个实施例的本系统可使用存取队列来确认训练。根据本公开的一个实施例的本系统可使用置信级别,例如,衡量出色的预取的数量。根据一个实施例,本系统提供了一种使得能够从训练最好的条目发出最多的预取的确认方案。

[0035] 根据各种实施例,本公开还提供了:通过在触发器的第二次出现具有的的提升以及使用触发器至触发器标签来进行更智能的训练;将触发器列入黑名单(可以是行内或行外);触发器至触发器跨步检测;以及使用触发器的散列来对表条目编索引。

[0036] 图1示出根据一个实施例的本系统的相关组件的示例性框图。一个或更多个核心处理器150执行涉及数据的指令,其中,数据和指令两者都存储在存储器中。为此,核心处理

器150在进行处理的同时针对数据和指令对L1高速缓存160进行存取。L1高速缓存160转而对L2高速缓存170进行存取。

[0037] 在对L2高速缓存170内存储的数据/指令进行存取方面,L2高速缓存170被分解为在下面被更详细地描述的空间区域。预取器165在多跨步引擎175和SMS引擎110的控制下执行从L2高速缓存170的空间区域的取出。如上所述,预取器165基于对核心处理器150将很快需要数据的预测来对L2高速缓存170中的存储器地址进行存取并将该数据加载至L1高速缓存160。对于关于这样的组件的操作和交互的更多细节,见Radhakrishan等人的序列号为9,665,491的标题为Adaptive Mechanism to Tune the Degree of Pre-Fetches[sic] Streams的美国专利以及Sundaram等人的公开号为2017/0116128的标题为Address Re-Ordering Mechanism for Efficient Pre-Fetch Training in an Out-Of-Order Processor的待审美国专利,所述两项与本申请都由同一受让人拥有,并通过引用合并于此。

[0038] 多跨步引擎175在它的预测的模式中使用多跨步,并且超出了本公开的范围。对于关于多跨步引擎175的操作的更多细节,见Radhakrishan等人的序列号为9,569,361标题为Prefetch Chaining的美国专利以及Radhakrishan等人的公开号为2016/0054997的标题为Computing System with Stride Prefetch Mechanism and Method of Operation Thereof的待审美国专利,所述两项与本申请都由同一受让人拥有,并通过引用合并于此。

[0039] SMS引擎110包括活性产生表(AGT)120和模式历史表(PHT)130。如在下文进一步讨论的,本质上,AGT 120记录和发现空间模式,而PHT 130存储来自AGT 120的包括其触发器(即,开始空间模式的开始程序计数器(PC)值)的空间模式。当识别出触发器时,根据存储的由触发器所指示的空间模式所指示的,PHT 130基于数据将被核心处理器150需要的可能性,对来自L2高速缓存170中的存储器地址的该数据进行存取,并将其加载至L1高速缓存160。

[0040] 根据实施例,AGT 120本身可以是单个表或多个表。如参照图4在下文所详细讨论的,常见的结构包含至少两个表(过滤表和训练表),这些表被用作一系列指令加载中的学习/发现空间模式的处理的一部分。

[0041] 因此,本公开可被认为是针对两个不同的处理:由AGT 120执行的训练/学习以及由PHT 130执行的实际操作。在解释训练/学习处理之前,下面提供了对SMS引擎110中的空间模式的存储的简要总结。

[0042] 图2是根据本公开的实施例的L2高速缓存170内的区域的示例性视图,图3是PHT 130中的条目300的示例性视图。

[0043] 如以上所讨论的,一旦AGT 120识别了空间模式,则包括针对该空间模式的触发器的空间模式被存储在PHT 130中。在图3中,PHT条目300被分解为多个字段。触发器以程序计数器(PC)的值(针对特定空间模式的触发器)的形式被存储为PC触发器字段310,区域字段320是以PC触发器字段310中的PC触发器开始的空间模式的区域的位置。基础偏移字段330指示与在PC触发器字段310中标识的PC触发器相关联的起始存储器位置(基础偏移),其中,起始存储器位置是从由区域字段320标识的区域的区域边界测量的。

[0044] 偏移列表字段340存储与在基础偏移字段330中标识的基础偏移相关的一系列偏移值(或“偏移”),其中,所述一系列偏移值构成触发器后面的空间模式。如本领域的普通技

术人员所理解的,例如,这样的偏移可被存储为单个空间向量(也被称为空间位向量)或被按顺序存储为在与PC触发器字段310中的值相关联的空间模式的区域字段320中标识的区域内的一系列位置。在这个实施例中,与如图2的区域200中所示的一系列偏移值的示例相同,偏移被存储为从基础偏移开始的一系列偏移。在将偏移存储为空间位向量的实施例中,空间模式被保留为N次迭代。在每次迭代中,使用(上一次迭代的)旧的模式对(当前迭代的)新的模式执行逐位AND函数(保守的)或OR函数(积极的)。

[0045] 返回图3,每偏移置信字段350存储与偏移列表字段340中存储的偏移中的每一个偏移相关联的置信级别,而总体置信字段360存储一系列偏移的整体(即,与PC触发器字段310中的值相关联的空间模式)的置信级别。置信级别是动态的,并且随着SMS引擎110从加载指令的正在进行的流学习而改变。

[0046] 图4示出根据一个实施例的SMS引擎的相关组件的示例性框图,其中,框图示出了信息在其中的移动。如以上所讨论的,SMS引擎400被划分为活性产生表(AGT)410和模式历史表(PHT)420。尽管AGT 410在概念上和逻辑上能够作为单个表(并以这种方式构建),但是在这个实施例中,AGT 410被实现为两个表:过滤表413和训练(或页)表415。为了便于解释,尽管(当然)在这个实施例中只有一个重新排序队列(ROQ),但是ROQ被示出在SMS引擎400的两侧上,其中,ROQ接受当前可用的各种高速缓存加载指令,并根据需要按顺序将它们放入正被执行的程序中。

[0047] 处理从ROQ中的高速缓存加载指令行开始,其中,该高速缓存加载指令行可能(或可能不)是对新的空间模式的缺失/触发,对新的空间模式的缺失/触发被暂时存储/分配在过滤表413中。在这个实施例中,条目包括与PHT条目300中的前三个字段相同的字段(PC触发器、区域(即,区域的地址)和基础偏移(即,从区域边界开始)),如下所示:

[0048] 过滤表条目字段

[0049]

PC触发器	区域	基础偏移

[0050] 如果下一加载指令在同一区域内(这意味着它可通过当前区域内的微小的偏移而被取出,而非取出另一区域),则相应的条目在训练表415中被做出。在这个实施例中,训练表条目包括与PHT条目300的前四个字段(即,PC触发器、区域、基础偏移和偏移列表字段)相同的字段。如下所示,偏移列表字段存储从基础偏移开始的一系列的8个不同偏移。

[0051] 训练表条目字段

[0052]

PC 触发器	区域	基础偏移	偏移列表								

[0053] 一旦来自过滤表413的PC触发器条目被分配至训练表条目,如果高速缓存加载指令仍然在同一个区域内,则训练表条目在偏移列表字段中存储附加的偏移。

[0054] 回收(eviction)(当PC触发器再次出现在ROQ中时发生)或失效(当正在使用的区域比训练表中的条目更多时发生)可停止空间模式,这导致回收的/失效的训练表条目在PHT 420中产生新的PHT条目。

[0055] 在一个实施例中,通过利用触发器至触发器标签(trigger-to-trigger stamp)来提供更智能的训练而非等待回收和失效,其中,在所述更智能的训练中,当PC触发器再次出现在ROQ中时发生从训练表到PHT的提升。在这个实现方案中,在第一个缺失处通过使用PC触发器、区域地址和基础偏移将条目分配到过滤表413中。在同一区域中的第二个缺失(具有不同的PC)处,将过滤表条目复制到训练表415中,并且从基础偏移开始的偏移被计算并还被存储在新的训练表条目的偏移列表字段中。在下次出现与PC触发器相同的针对PC的值时,将训练表条目提升至模式历史表420中的条目,并且使用新的地址作为基础区域和偏移来开始预取。在提升前,任何在相同区域中进行的附加存取都使其偏移被计算并被存储在训练表条目中。

[0056] 学习/训练过程大致如上所述,现在将在下文简单地描述SMS引擎400的活性操作。

[0057] 如图4中还示出的,SMS引擎400可从ROQ中的当前高速缓存加载指令识别PHT 420条目中存储的空间模式的PC触发器,并基于PHT 420条目中存储的空间模式来提供先前识别和存储的与PHT 420中的PC触发器相应的空间模式以用于预取数据。

[0058] 图5是根据本公开的实施例的用于训练/学习以及存储发现的空间模式的示例性方法的修改的流程图。

[0059] 图5从重新排序队列(ROQ) 505开始,其中,重新排序队列将当前可用的各种高速缓存加载指令按在程序中实际执行的顺序进行排序。在本实施例中,基于效率的考虑,大致并行地执行510、520和530。在510,本系统确定高速缓存加载指令中所指示的存储器存取是否已经在L1高速缓存器中,在这种情况下(即,存储器存取已经在L1高速缓存器中),所述方法针对ROQ中的下一高速缓存加载指令在515返回。如果所需的在由高速缓存加载指令所指示的存储器地址处存取的数据在L1高速缓存中不可用,则本系统在550执行如下文更全面地描述的SMS训练。在520,本系统确定高速缓存加载指令是否是用于多跨步引擎的。如果高速缓存加载指令是用于多跨步引擎的,则高速缓存加载执行被转发至多跨步引擎。如果高速缓存加载指令不是用于多跨步引擎的,则本系统550执行SMS训练。

[0060] 在530,本系统确定包含所需的在由高速缓存加载指令所指示的存储器地址处的数据的区域是否在活性模式区域内。如果包含所需的在由高速缓存加载指令所指示的存储器地址处的数据的区域落入活性模式区域内,则在535,本系统确定在包含所需的在由高速缓存加载指令所指示的存储器地址处存取的数据的区域中的位置是否通过当前PHT条目中的任何偏移被标识。如果包含所需的在由高速缓存加载指令所指示的存储器地址处存取的数据的区域中的位置未通过当前PHT条目中的任何偏移被标识,则在540,向训练表添加针对所需的数据的位置的偏移或执行另一种形式的再训练。如果包含所需的在由高速缓存加载指令所指示的存储器地址处存取的数据的区域中的位置通过当前PHT条目中的(下一个)偏移被标识,则在545,针对该偏移的每偏移置信级别提高。如下文进一步详细讨论的,这一提高也被视为对总体置信级别的“确认”。

[0061] 下面的表1示出在本公开的一个实施例中,SMS训练的逻辑输入和输出形式的结果。更具体地,基于过滤表的状态和训练表的状态,或者更确切地说,基于过滤表或训练表的PC和区域字段是否与当前高速缓存行加载指令中的相同的值匹配,表1示出了结果。表1仅意图帮助解释/描述,并且不与硬件中的实际表(诸如查找表(LUT))相应,而是示出基于输入(当前高速缓存行加载指令)以及训练表状态和过滤表状态来产生这样的结果的硬件

组件的系统的输出。

[0062] 表1:SMS训练的结果

[0063]

过滤表 PC匹配	过滤表 区域匹 配	训练表 PC匹配	训练表区 域匹配	结果
0	0	0	0	分配过滤表条目
0	0	0	1	向训练表条目添加偏移
0	0	1	0	将训练表条目提升至PHT条目（开始在新区域中进行预取）
0	0	1	1	将训练表条目提升至PHT条目（从新的偏移开始，开始在旧区域中进行预取）
0	1	0	0	将过滤表条目提升至训练表条目（添加偏移，并从过滤表解除分配）
0	1	0	1	（不可能）
0	1	1	0	将过滤表条目提升至训练表条目（添加偏移，并从过滤表解除分配） 将训练表条目提升至PHT条目
0	1	1	1	（不可能）
1	0	0	0	使用新区域和偏移重置过滤表条目
1	0	0	1	使用新区域和偏移重置过滤表条目 向训练表条目添加偏移
1	0	1	0	（不可能）
1	0	1	1	（不可能）
1	1	0	0	使用新的偏移重置过滤表条目
1	1	0	1	（不可能）
1	1	1	0	（不可能）
1	1	1	1	（不可能）

[0064] 在表1中标记为“（不可能）”的条目中，在代码/寄存器传输语言（RTL）中存在防止所述结果的发生的判断提示（assert）。条目n1h1（例如，0101和0111）被列为“不可能”，这是

因为该区域对于过滤表或训练表中的条目是独一无二的,并且一旦过滤表条目被提升至训练表条目后,任何后来的在相同区域中的PC存取仅仅更新训练表。条目1n1n(例如,1010和1111)被列为“不可能”,这是因为一旦给定的PC处于训练表中,第二次出现造成条目向PHT的提升,因此PC触发器的后来的出现不能被分配给过滤表,而是被用来重新训练PHT表条目。

[0065] 图6A至图6C提供根据本公开的实施例的基于过滤表状态和训练表状态的SMS训练结果的部分流程图格式的示例。与表1类似,图6A至图6C中的流程图决策菱形是用于解释/描述的目的,并且不与在软件或硬件中实现的决策点相应,而是示出根据本公开的基于输入(当前高速缓存行加载指令)以及训练表状态和过滤表状态的硬件组件的系统的输出。

[0066] 参照图6A,在610,本系统确定当前的ROQ高速缓存加载指令或存取的PC是否与训练表条目的PC匹配。如果当前的ROQ高速缓存加载指令或存取的PC与训练表条目的PC匹配(在610,是=1),则在615,训练表条目被提升至PHT条目。如果当前的ROQ高速缓存加载指令或存取的PC与训练表条目的PC不匹配(在610,否=0),则在620,本系统确定当前的ROQ高速缓存加载指令的PC是否与过滤表条目的PC匹配。如果当前的ROQ高速缓存加载指令的PC与过滤表条目的PC匹配(在620,是=1),则在625,过滤表条目被重置为新的区域。如果当前的ROQ高速缓存加载指令的PC与过滤表条目的PC不匹配(在620,否=0),则在627,不采取行动。

[0067] 参照图6B,在630,本系统确定当前的ROQ高速缓存加载指令的区域是否与过滤表条目的区域匹配。如果当前的ROQ高速缓存加载指令的区域与过滤表条目的区域匹配(在630,是=1)(这意味着下一存储器存取处于相同的区域内并且可由偏移来定义),则在635,将过滤表条目提升至训练表条目。如果当前的ROQ高速缓存加载指令的区域与过滤表条目的区域不匹配(在630,否=0),则在637,不采取行动。这些将PC以及过滤表的区域值与当前的ROQ高速缓存加载指令进行比较/匹配的操作(图6A中的620和图6B中的630)通常同时(并行地)操作。

[0068] 参照图6C,在640,本系统确定当前的ROQ高速缓存加载指令的区域是否与训练表条目的区域匹配。如果当前的ROQ高速缓存加载指令的区域与训练表条目的区域匹配(在640,是=1),则在645,该区域内的新存储器位置的偏移被添加到训练表条目。在640,当存在训练表区域匹配时,所有的训练表条目被更新。训练表条目根据它们的PC触发器是独一无二的,并且对于一个区域可存在多个PC触发器。为解决这个问题,针对传入区域,所有的训练表条目都被更新。如果当前的ROQ高速缓存加载指令的区域与训练表条目的区域不匹配(在640,否=0),则在647,不采取行动。

[0069] 根据一个实施例,特定的PC触发器被列入黑名单,以避免污染训练结构,从而即使在嘈杂的条件下也能进行良好的训练。黑名单上列出的PC触发器不能用于预取,从而防止无用的训练和发出不好的预取。“行内”黑名单使用PHT条目本身的存储的置信级别来防止其自身的使用,而“行外”黑名单使用单独的表来保存PHT容量。

[0070] 此外,触发器至触发器跨步检测使预取流能够在需要前到达。在触发器至触发器跨步检测中,在触发器PC第二次出现时,跨步在触发器PC的第一次出现与第二次出现之间被计算并被存储在模式历史表420中。随着触发器PC的每次新的出现,从之前的触发器PC开始的新的跨步被计算。这些跨步用作标准跨步检测算法的输入,其中,标准跨步检测算法的

输出能够被用于驱动更远区域中的空间相关预取。

[0071] 根据一个实施例,训练表和/或PHT条目的PC触发器被散列并被用作表条目(而非PC触发器自身)的索引值,从而在保持用于训练和激活的存取模式的分离的同时减少存储。完整的PC具有多个位-在64位架构中至少有40-并且,部分PC能够以保存大部分信息(并且至少是使得散列的版本彼此能够区分的足够信息)但使用更少的方式被算术地组合(“散列”),而非存储完整的40位。在这些实施例中,更小的“散列”值被存储在表的“PC触发器”字段而非PC触发器的实际的40个位中。此外,用于存储PC值的任何字段都能够由其散列的值替换,而非存储其完整的值。

[0072] 图7示出PHT条目是如何从一系列高速缓存行加载指令被创建或在ROQ中进行存取的示例。图7示出过滤表、训练表和SMS的PHT,以及来自ROQ的一系列高速缓存加载命令。图7中的训练表条目仅示出从基础偏移开始的前三个偏移(O_1 、 O_2 、 O_3),省略号指示能够存在训练表或PHT条目中存储的任意数量的偏移,尽管在这个实施例中,在训练表条目和PHT条目二者中仅存储了8个偏移。类似地,图7中的PHT条目具有所述前三个偏移(O_1 、 O_2 、 O_3)和省略号,并且,在这些字段下面,存在作为针对每个偏移的相应置信值的对应字段 Cf_1 、 Cf_2 、 Cf_3 、...。这种配置仅仅是为了便于描述,因此每个偏移出现在它的每偏移(per-offset)置信级别附近。针对整个空间模式的总体置信级别 Cf_{all} 是最后的字段。

[0073] 在图7中,具有 $PC=x$ 的第一命令在于将数据/命令加载到存储器位置103C(在十六进制计数法中,其中的每个数位可以是0-F,即,0-15)。在这个实施例中,假设每个区域是 $4kB=4096$ 字节,并且因为十进制计数法中的4096等于十六进制计数法中的1000,因此存储器位置的第一个数位标识其区域-在这种情况下,是区域1000。在710,存在一个缺失,因此条目在过滤表中被创建,其中,条目具有(可能的)PC触发器、区域(1000)和基础偏移(3C)。

[0074] 下一加载指令是针对与前一加载指令在相同的区域内的存储器位置1044。因此,由于所述两个加载指令形成了具有区域的空间模式,因此在720,在训练表中做出新的条目。训练表中的条目具有PC触发器(x)、区域(1000)、偏移(3C)和偏移723,其中,当与基础偏移组合时,偏移723获取被加载的当前存储器位置的值(十进制计数法中的+8以从十六进制计数法中的103C获得1044)。

[0075] 下一加载指令是在存储器位置1000,它仍位于新的训练表条目的区域内。因此,新的偏移725(十进制计数法中的-60)被添加到训练表条目,从而增加了区域内的空间模式的长度。

[0076] 第四个且时最后一个加载指令是在存储器位置5C0A,它位于远离新的训练表条目的区域之外。因此,在715,最老的过滤表条目被新的过滤表条目所替代,其中,新的条目具有新的空间模式的(可能的)PC触发器、起始区域(5000)和基础偏移(C0A)。

[0077] 在730,在某个时间点,训练表被解除分配并变成新的PHT条目。尽管在图7中715和730是同时出现的,但是训练表条目向PHT条目的提升是完全独立于过滤动作的。因为PHT条目是新的,因此每偏移置信级别字段中或总体置信级别字段中没有值存在。将参照图8在下文描述这样的值是如何被创建和改变的。

[0078] 图8示出当在ROQ中识别出存储的PHT条目的PC触发器时,PHT条目是如何被用于从由它的偏移列表字段中存储的空间模式所指示的存储器位置预取数据/指令的示例。此外,根据本公开的实施例,使用PHT条目中的每偏移置信字段和总体置信字段来跟踪单独的偏

移和总体空间模式二者的精确度。

[0079] 图8仅示出根据本公开的实施例的SMS的PHT和来自ROQ的一系列高速缓存加载命令,并且另一“预取/加载”列被添加以示出如何实现加载指令。由于偏移是按照十进制计数法的,因此预取/加载列的部分示出来自存储器位置的十进制的值的加减。

[0080] 在图8中,第一加载命令具有 $PC=x$,其中,第一加载命令将PHT条目与 x 的PC触发器匹配,这在810激活用于预取的PHT条目。至少因为开始位置是任意的,并且实际空间模式是由从起始点开始的相对偏移所定义的,因此在820,第一加载命令的区域和基础偏移值替换之前的值。尽管在820示出这个替换与第二加载命令相邻,但是在本实施例中,该替换可与PHT条目被激活同时发生,即,与第一加载命令大致同步。

[0081] 根据PHT条目,下一个加载是在从第一加载指令开始的+8偏移处。然而,从L2高速缓存下载的高速缓存行(或块)是从加载指令中的存储器位置开始的32字节、64字节或128字节长度。因此,在+8偏移处的存储器位置已由之前的加载指令下载了。在根据本公开的实施例中,使用偏移的尺寸以防止不必要的负担,因此不存在从L2高速缓存至L1高速缓存的额外下载,这是因为在+8偏移处的存储器位置已从前一加载指令进入L1高速缓存。

[0082] 然而,本公开的实施例的另一方面是对粒度的每偏移置信级别和总体置信级别二者方面对空间模式的精确度进行追踪。由于第二加载指令与预测出的PHT条目中的+8偏移(即,30F8)处的存储器位置相匹配,因此在830,加1(+1)被添加至与第一偏移 O_1 相应的每偏移置信字段 Cf_1 中。类似地,因为 O_2 的-60偏移也与下一加载指令(30B4)相匹配,因此在840,加1(+1)被添加至与第二偏移 O_2 相应的每偏移置信字段 Cf_2 中。

[0083] 尽管上文讨论了一系列事件,但是PHT条目的功能是预取,并且由于在激活的PHT条目中偏移值是立即可用的,因此从L2高速缓存下载30B4高速缓存行/块是与在810激活PHT条目大致同时做出的。只有每偏移置信级别的分配需要等待ROQ中的实际加载指令,以便对下载/预取的事物是否是实际需要的事物进行评估。

[0084] 如最初在图7中所记录的,PHT条目中存储的空间模式以 $O_2=-60$ 结束。然而,在这一系列指令中,用于存储器位置(3100)的下一加载指令处于PHT条目的区域(3000)内。正因为如此,在850,下一存储器位置的偏移(+16)作为 O_3 被添加至PHT条目。因此,在本公开的实施例中,在核心处理器的操作期间,PHT条目被持续细化和评估。因为 O_3 刚被添加至PHT条目,因此它还不具有相应的置信级别 CF_3 。根据本公开的其他实施例,新添加的偏移的置信级别首先可被分配+1的值或已表现出对于这一系列的指令是良好的起始置信级别的另一个值。

[0085] 在本公开的其他实施例中,特定偏移的随时间的每偏移置信级别可被用于消除不具有匹配实际加载指令的足够合理的概率的预取。在下面的PHT条目的示例中,6个偏移的空间模式具有比其他偏移的置信级别(O_2 、 O_3 和 O_5)更高的置信级别(O_1 和 O_6)的一些偏移。在本公开的一个实施例中,当PHT条目被激活时,置信级别为0的任何偏移(诸如 O_3)不会与其余偏移一起被预取。由错误的预取导致的资源浪费比不使用可能好也可能不好的预取的风险(即,不使用由于其置信级别太低而被忽略的正确的预取的风险)更重要。

[0086] 如另一示例,在本公开的另一实施例中,设置一阈值,在该阈值之下的任何偏移值将不被预取。使用下面的示例,如果建立阈值 >1 ,则由偏移 O_2 、 O_3 和 O_5 所指示的存储器位置将不被预取。以这种方式,本公开的实施例能够消除由于整体空间模式中的弱成分所导致

的错误的预取,同时保留已显示出随时间而精确的预取。

[0087] PHT条目的示例

PC触发器	区域	基础偏移	O_1 Cf ₁	O_2 Cf ₂	O_3 Cf ₃	O_4 Cf ₄	O_5 Cf ₅	O_6 Cf ₆	O_7 Cf ₇	O_8 Cf ₈	CF _{a11}
x	2000	C3	+15	-32	+2	+12	-60	+5	-	-	-
			+3	+1	0	+2	+1	+4	-	-	

[0089] PHT条目中存储的全部空间模式的总体置信级别(CF_{a11})可被计算,并可被用于以许多不同更多方式来提高预取精确度。例如,总体置信级别CF_{a11}可被计算为饱和计数器,其中,饱和计数器在确认时值增加,并且当触发器PC激活不具有相应确认的PHT条目时(在最坏的情况下)值减少。如上面所讨论的,如在图5的535所示的,确认可以是当存储的PHT条目的空间模式中的偏移被示出为正确并且它的每偏移置信级别增加+1。

[0090] 总体置信级别CF_{a11}可被用于例如通过确定在看到第一次确认前应该发出多少次预取和/或针对每个确认发出多少次预取(这可能是非线性的,例如,第一次确认触发再两次预取,而第二次确认触发3次预取等)来提高预取精确度。

[0091] 如上所详细描述,特别地,本公开还提供了以下内容:适应于改变程序行为的持续学习;捕捉时间顺序的偏移列表;能够将对前一行或后续行的存取进行区分的字节粒度;使得单独的偏移能够针对精确度随着时间被追踪的每偏移置信;以及使得从训练最好的条目发出最多的预取的确认方案。

[0092] 特别地,本公开还提供了以下内容:通过在PC触发器的第二次出现并使用触发器至触发器标签来进行更智能的训练;将PHT中的PC触发器列入黑名单(可以是行内或行外);触发器至触发器跨步检测;以及使用PC触发器的散列来对PHT条目编索引。

[0093] 图9示出根据一个实施例的本设备的示例性框图。设备900包括至少一个处理器910以及一个或多个非暂时性计算机可读介质920。当所述至少一个处理器910执行所述一个或多个非暂时性计算机可读介质920上存储的指令时,所述至少一个处理器910执行以下步骤:维护用于空间存储器流传输(SMS)预取引擎的模式历史表(PHT),其中,PHT包括至少一个表,所述至少一个表中的每个条目具有偏移列表字段和每偏移置信字段,其中,偏移列表字段包括针对从区域内的基础偏移值开始的偏移值的子字段,每偏移置信字段包括针对与每个偏移值相应的每偏移置信级别的子字段;当PHT条目被激活时,通过将当前存取与激活后的PHT条目中存储的偏移值进行匹配来更新与激活后的PHT条目中的每个偏移值相应的每偏移置信值。此外,所述一个或多个非暂时性计算机可读介质920存储所述至少一个处理器910用于执行那些步骤的指令。可至少按照每偏移置信级别向SMS引擎提供持续学习。

[0094] 在另一实施例中,当所述至少一个处理器910执行所述一个或多个非暂时性计算机可读介质920上存储的指令时,所述至少一个处理器910执行以下步骤:维护用于空间存储器流传输(SMS)引擎的过滤表,其中,过滤表中的每个条目包括程序计数器(PC)触发器、区域地址和基础偏移;维护用于SMS引擎的训练表,其中,训练表中的每个条目包括PC触发器、区域地址、基础偏移和针对与区域地址在相同区域的存储器位置的从基础偏移开始的偏移的列表;当SMS引擎的重新排序队列(ROQ)中的当前加载指令/存取导致缺失时,将新

的条目分配到过滤表中,其中,所述新的条目的PC触发器是所述当前加载指令/存取的PC,并且所述新的条目的区域地址和基础偏移包括由所述当前加载指令/存取所指示的存储器位置;当由ROQ中的下一加载指令/存取所指示的存储器位置在同一区域内时,通过分配新的训练表条目将新的过滤表条目提升至训练表中的条目,其中,所述新的训练表条目包括过滤表条目的PC触发器、过滤表条目的区域地址、过滤表条目的基础地址以及偏移的列表中的第一偏移,其中,所述第一偏移是从基础偏移开始测量的所述区域内的存储器位置的距离;以及,每当由ROQ中的下一加载指令/存取所指示的存储器位置在同一区域内时,向训练表条目中的偏移的列表添加另一偏移,其中,所述另一偏移是从基础偏移开始测量的所述区域内的存储器位置的偏移距离。此外,所述一个或更多个非暂时性计算机可读介质920存储所述至少一个处理器910用于执行那些步骤的指令。

[0095] 图10示出根据一个实施例的用于制造和测试包括空间存储器流传输(SMS)预取引擎的本设备的示例性流程图。

[0096] 在1050,包括至少一个处理器以及一个或更多个非暂时性计算机可读介质的设备(在这个示例中,芯片组)被制造。当执行所述一个或更多个非暂时性计算机可读介质上存储的指令时,所述至少一个处理器执行以下步骤:维护用于空间存储器流传输(SMS)预取引擎的模式历史表(PHT),其中,PHT包括至少一个表,所述至少一个表中的每个条目具有偏移列表字段和每偏移置信字段,其中,偏移列表字段包括针对从区域内的基础偏移值开始的偏移值的子字段,每偏移置信字段包括针对与每个偏移值相应的每偏移置信级别的子字段;当PHT条目被激活时,通过将当前存取与激活后的PHT条目中存储的偏移值进行匹配来更新与激活后的PHT条目中的每个偏移值相应的每偏移置信值。所述一个或更多个非暂时性计算机可读介质存储所述至少一个处理器用于执行那些步骤的指令。可至少按照每偏移置信级别向SMS引擎提供持续学习。

[0097] 在1060,设备(在这个示例中,芯片组)被测试。测试1060包括:测试所述设备是否具有至少一个处理器,其中,当所述至少一个处理器执行一个或更多个非暂时性计算机可读介质上存储的指令时,所述至少一个处理器执行以下步骤:维护用于空间存储器流传输(SMS)预取引擎的模式历史表(PHT),其中,PHT包括至少一个表,所述至少一个表中的每个条目具有偏移列表字段和每偏移置信字段,其中,偏移列表字段包括针对从区域内的基础偏移值开始的偏移值的子字段,每偏移置信字段包括针对与每个偏移值相应的每偏移置信级别的子字段;当PHT条目被激活时,通过将当前存取与激活后的PHT条目中存储的偏移值进行匹配来更新与激活后的PHT条目中的每个偏移值相应的每偏移置信值;以及测试所述设备是否具有存储所述至少一个处理器用于执行上述步骤的指令的一个或更多个非暂时性计算机可读介质。可至少按照每偏移置信级别向SMS引擎提供持续学习。

[0098] 在另一实施例中,包括至少一个处理器以及一个或更多个非暂时性计算机可读介质的芯片组被制造。当所述至少一个处理器执行所述一个或更多个非暂时性计算机可读介质上存储的指令时,所述至少一个处理器执行以下步骤:维护用于空间存储器流传输(SMS)引擎的过滤表,其中,过滤表中的每个条目包括程序计数器(PC)触发器、区域地址和基础偏移;维护用于SMS引擎的训练表,其中,训练表中的每个条目包括PC触发器、区域地址、基础偏移和针对与区域地址在相同区域的存储器位置的从基础偏移开始的偏移的列表;当SMS引擎的重新排序队列(ROQ)中的当前加载指令/存取导致缺失时,将新的条目分配到过滤表

中,其中,所述新的条目的PC触发器是所述当前加载指令/访问的PC,并且所述新的条目的区域地址和基础偏移包括由所述当前加载指令/存取所指示的存储器位置;当由ROQ中的下一加载指令/存取所指示的存储器位置在同一区域内时,通过分配新的训练表条目将新的过滤表条目提升至训练表中的条目,其中,所述新的训练表条目包括过滤表条目的PC触发器、过滤表条目的区域地址、过滤表条目的基础地址以及偏移的列表中的第一偏移,其中,所述第一偏移是从基础偏移开始测量的区域内的存储器位置的距离;以及,每当由ROQ中的下一加载指令/存取所指示的存储器位置在同一区域内时,向训练表条目中的偏移的列表添加另一偏移,其中,所述另一偏移是从基础偏移开始测量的区域内的存储器位置的偏移距离。此外,所述一个或更多个非暂时性计算机可读介质存储所述至少一个存储器用于执行上述步骤的指令。

[0099] 如本领域的普通技术人员所理解的,上面描述的关于本公开的实施例的步骤和/或操作可以根据特定的实施例和/或实现方案以不同的顺序或并行地发生,或者同时用于不同的时期等。不同的实施例可以以不同的顺序或通过不同的方法或手段来执行动作。如本领域的普通技术人员所理解的,一些附图是执行的动作的简化表示,它们的描述在此简化地概述,但是现实世界的实现方案会更加复杂,需要更多个阶段和/或组件,并还会根据特定实现方案的需求而改变。作为简化表示,这些附图未示出与可能被本领域的普通技术人员所公知和理解以及可能与本说明书不相关和/或对本说明书没有帮助的步骤相同的其他所需步骤。

[0100] 类似地,如本领域的普通技术人员所理解的,一些附图是仅示出相关组件的简化框图,这些组件中的一些组件仅代表在该领域中所公知功能和/或操作,而不代表实际的硬件。在这种情况下,可以以各种方式和/或方式的组合(诸如至少部分地以固件和/或硬件)来实现或提供一些或全部这些组件/模块,包括但不限于一个或更多个专用集成电路(“ASIC”)、标准集成电路、执行适当指令的控制器(包括微控制器和/或嵌入式控制器)、现场可编程门阵列(“FPGA”)、复杂可编程逻辑装置(“CPLD”)等。一些或全部系统组件和/或数据结构也可被存储为非暂时性计算机可读介质(例如,硬盘、存储器、计算机网络或蜂窝无线网络或其他数据传输介质、或通过适当的驱动器或经由连接将被读取的便携式媒体物(诸如DVD或闪存装置))上的内容(例如,可执行的或其他机器可读的软件指令或结构数据),以启用或配置计算机可读介质和/或一个或更多个相关的计算系统或装置来执行或使用或提供用于执行描述的技术中的至少一些技术的内容。

[0101] 无论是被单独布置还是多处理地布置,一个或更多个处理器、简单的微控制器、控制器等都可被用于执行非暂时性计算机可读介质上存储的指令的序列以实现本公开的实施例。在一些实施例中,硬接线电路可用来替换软件指令或与软件指令结合使用。因此,本公开的实施例不限于硬件电路、固件和/或软件的任何特定组合。

[0102] 这里使用的术语“计算机可读介质”指存储可被提供给处理器以用于执行的指令的任何介质。这样的介质可具有多种形式,包括但不限于非易失性的和易失性的介质。非暂时性计算机可读介质的常见形式包括:例如,软盘、软磁盘、硬盘、磁带或任何其他磁性介质、CD-ROM、任何其他光学介质、打孔卡、纸带、具有孔的模式的其他物理介质、RAM、PROM和EPROM、FLASH-EPROM、任何其他存储器芯片或盒(cartridge)、或存储由可由处理器执行的指令的任何其他介质。

[0103] 本公开的一些实施例可至少部分地在便携式装置上实现。这里使用的“便携式装置”和/或“移动装置”指具有接收无线信号的能力的任何便携式或可移动的电子装置,包括但不限于多媒体播放器、通信装置、计算装置、导航装置等。因此,移动装置包括(但不限于)用户设备(UE)、膝上型计算机、平板计算机、便携式数字助理(PDA)、MP3播放器、手持式PC、即时消息装置(IMD)、蜂窝电话、全球导航卫星系统(GNSS)接收器、手表、或可被一个人穿戴和/或携带的任何这样的装置。

[0104] 如本领域的普通技术人员所理解的,根据本公开,本公开的各种实施例可实现在集成电路(IC)(也被称为微芯片、硅芯片、计算机芯片或只是“芯片”)中。这样的IC可以是例如宽带和/或基带调制解调器芯片。

[0105] 本公开不限于这里讨论的实施例,并且如本领域的普通技术人员所理解的,具体实现方式的细节将取决于预期的用途、环境等而变化,因此本公开意图应用于其范围内的任何实现方案。

[0106] 虽然已描述了一些实施例,但是将理解的是,在不脱离本公开的范围的情况下可做出各种修改。因此,本领域的普通技术人员将清楚的是,本公开不限于这里描述的任何实施例,而是具有由权利要求和它们的等同物所限定的范围。

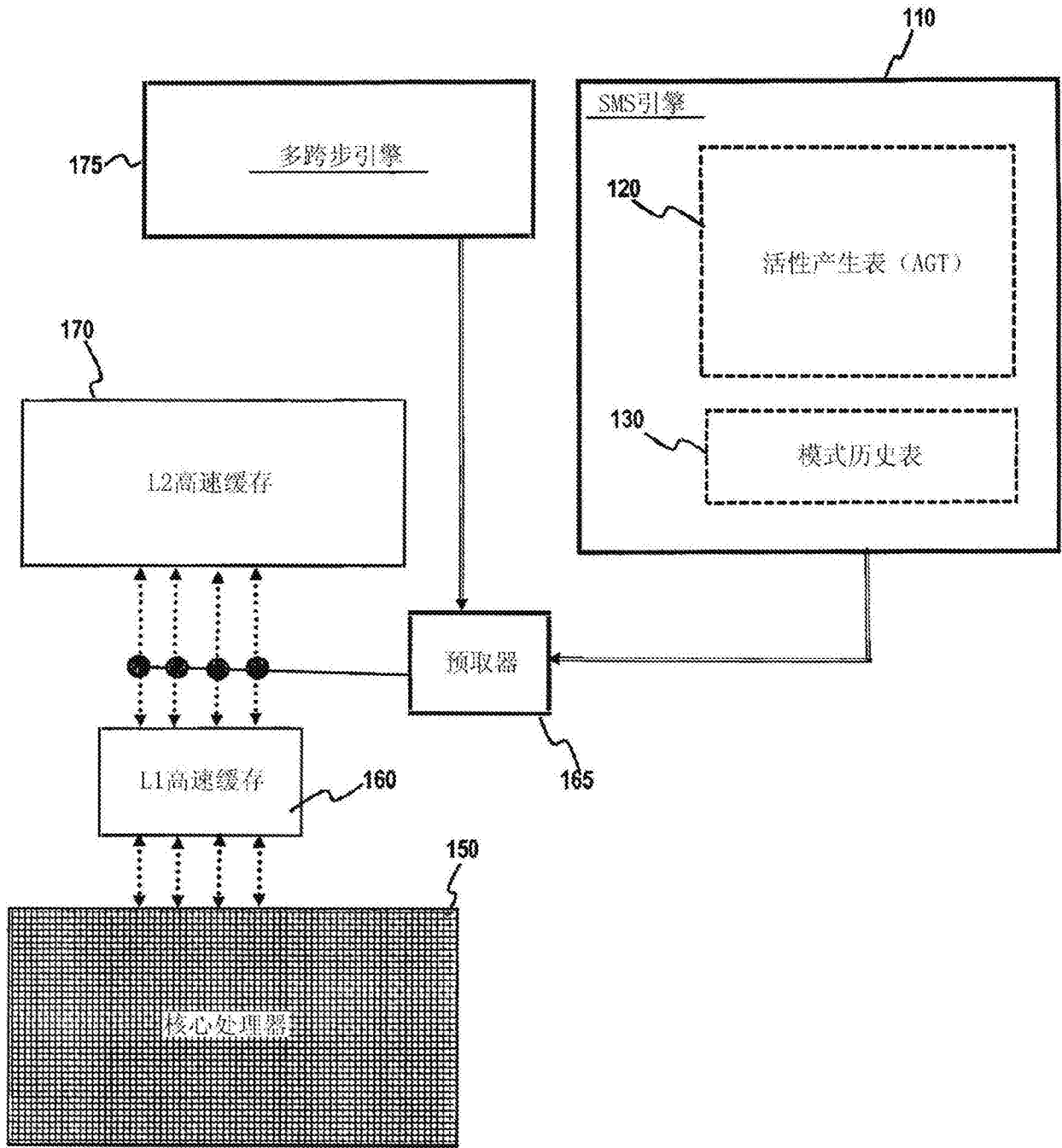


图1

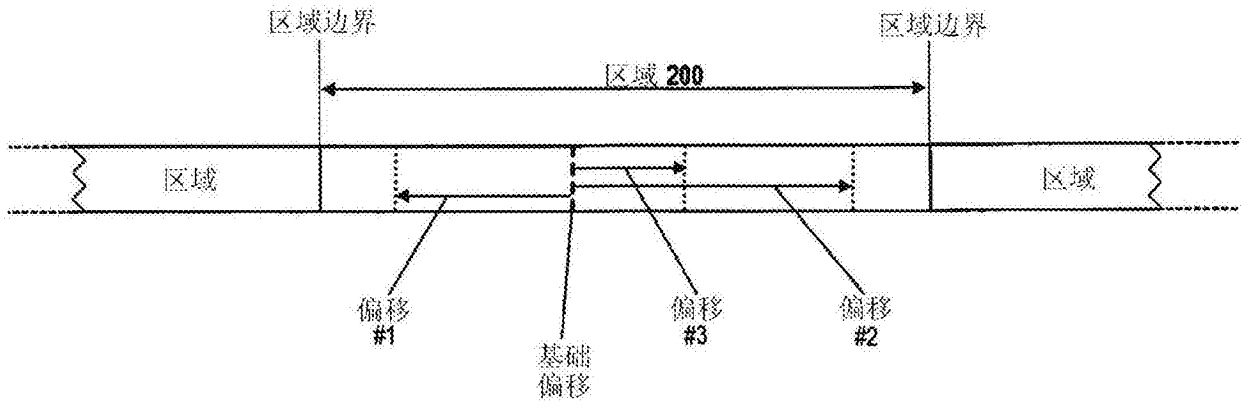


图2

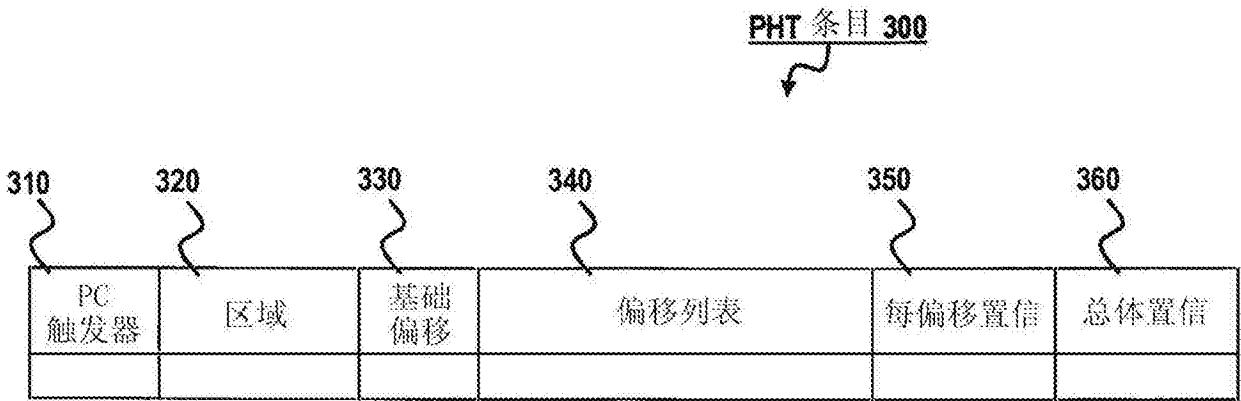


图3

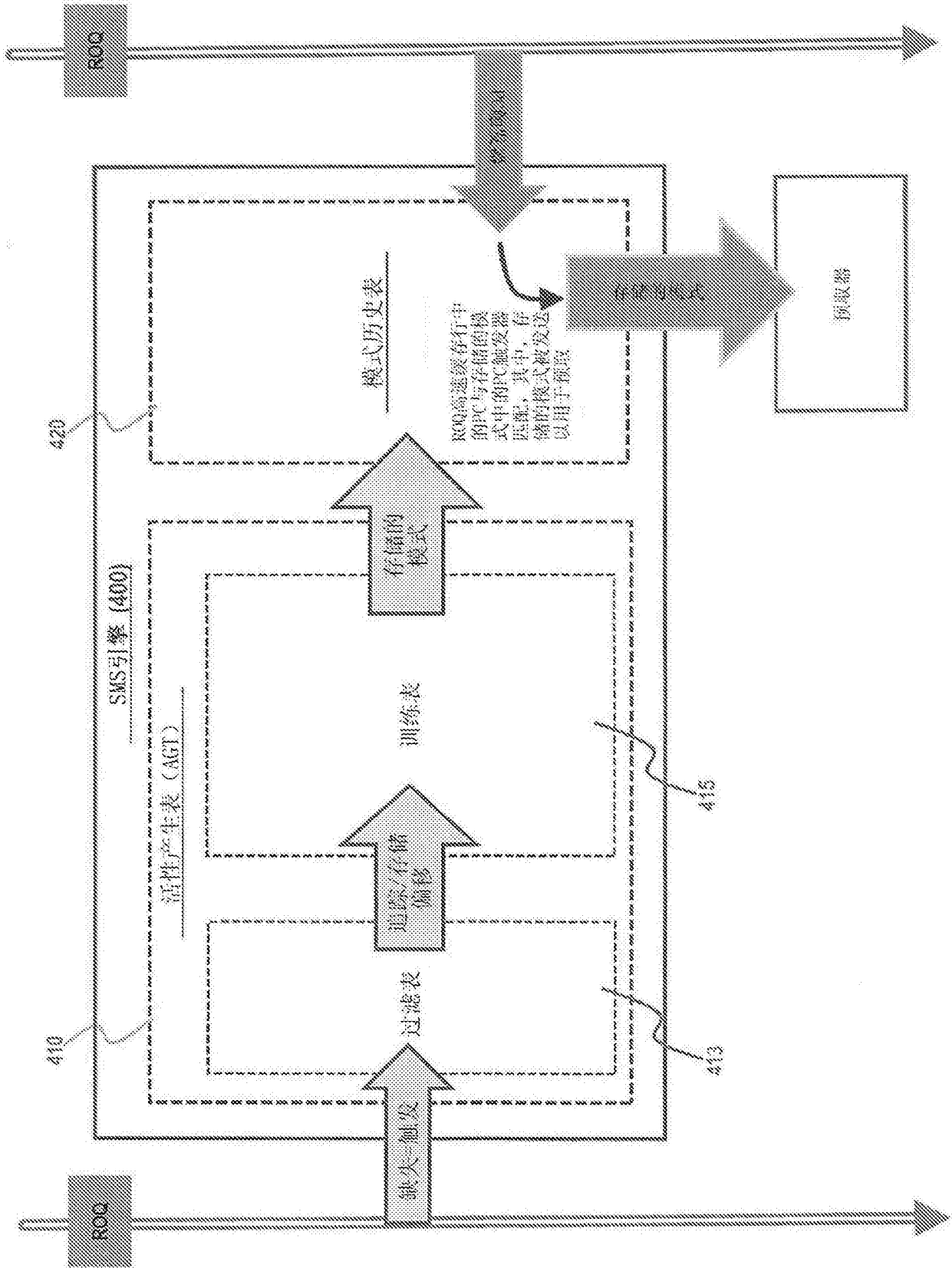


图4

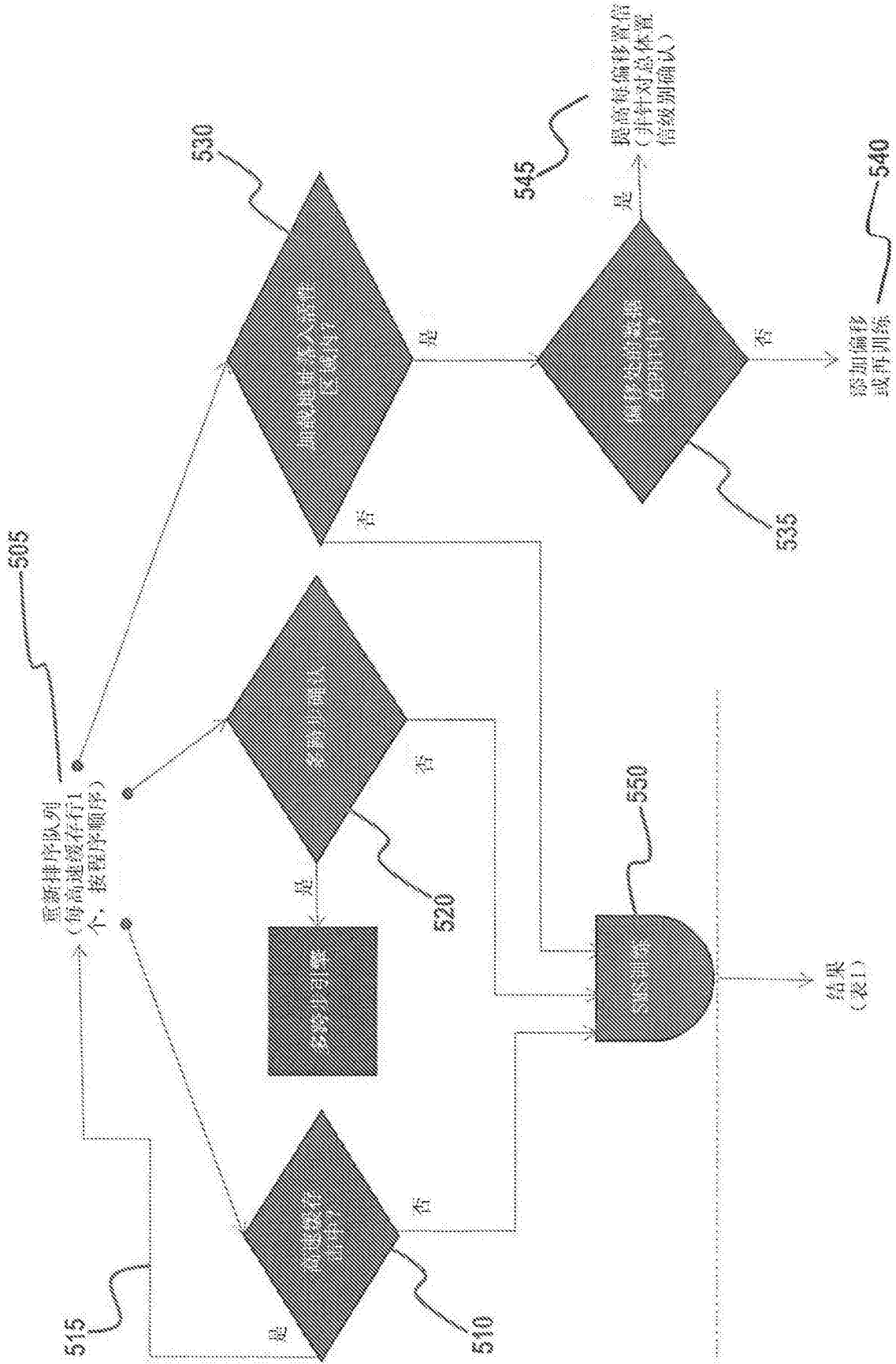


图5

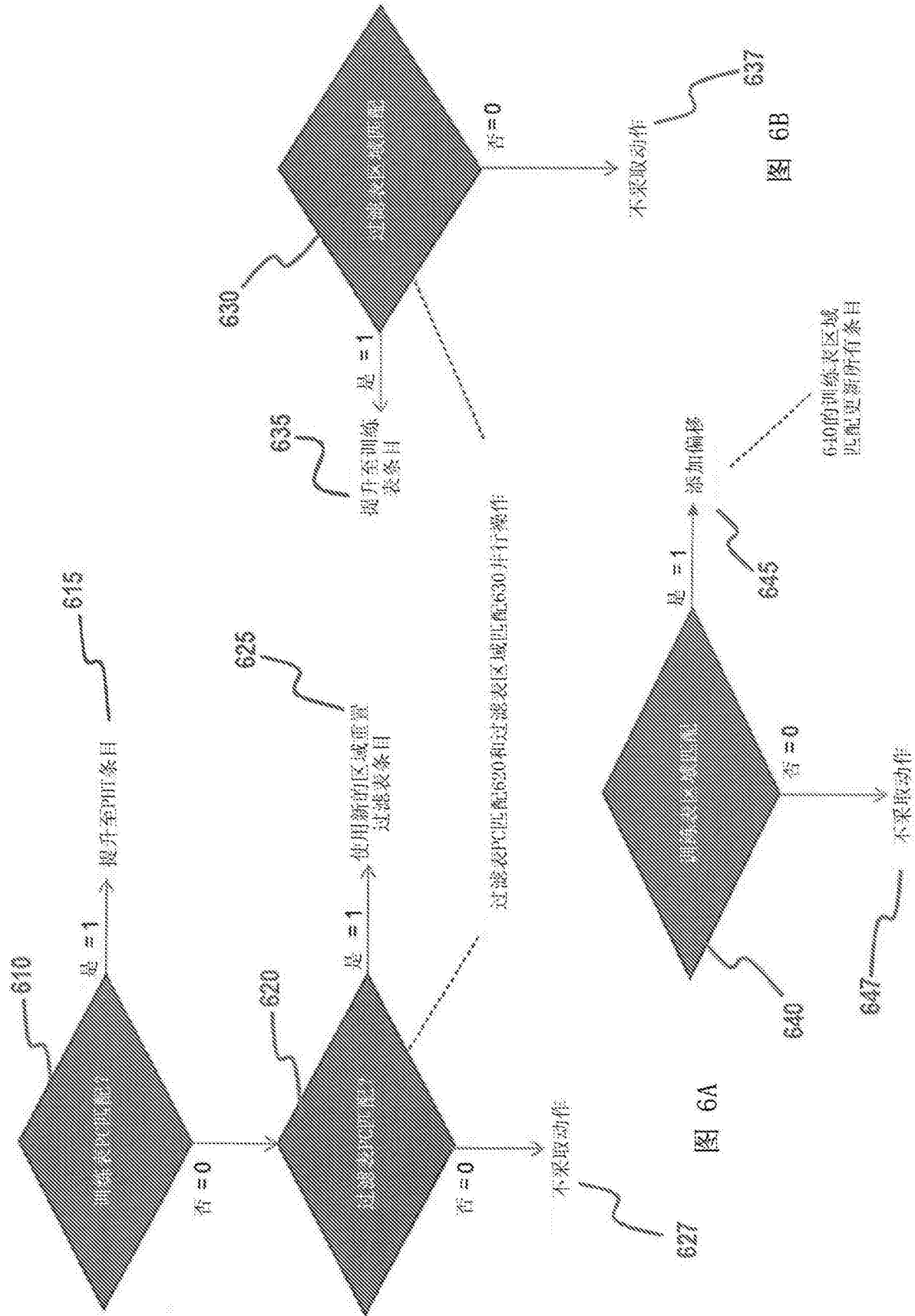


图 6C

图 6A

图 6B

学习/创建PHI条目

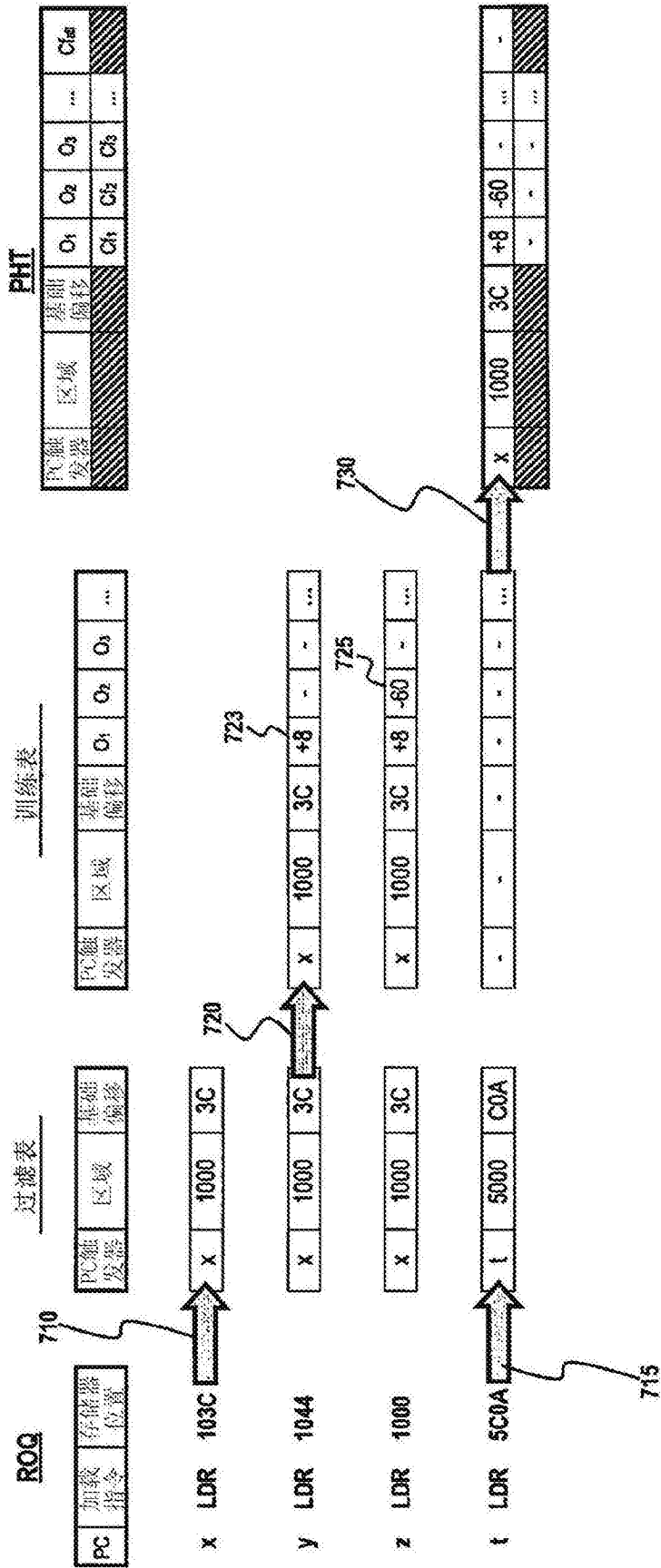


图7

触发所存储的PHT条目的空间模式

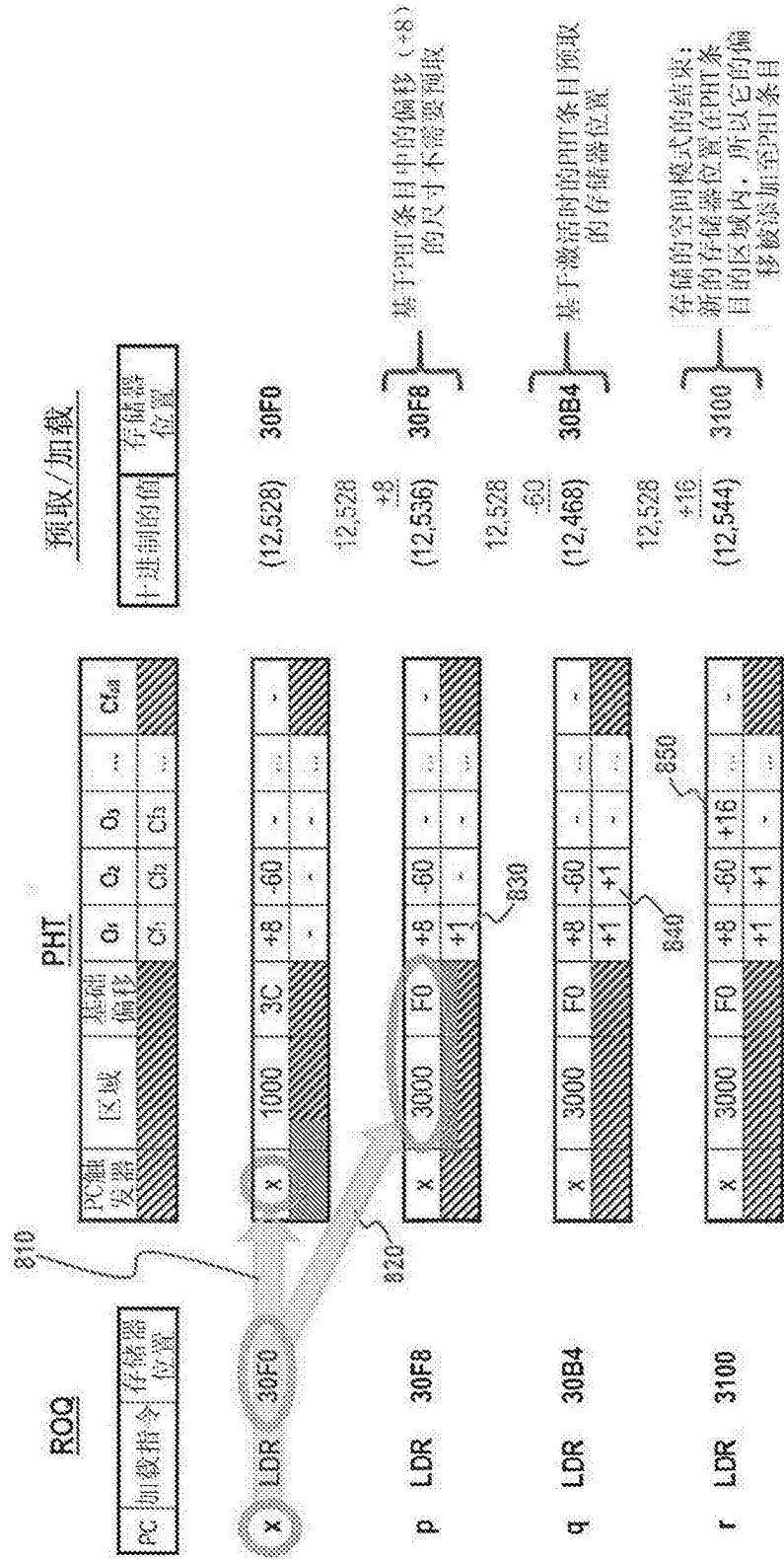


图8

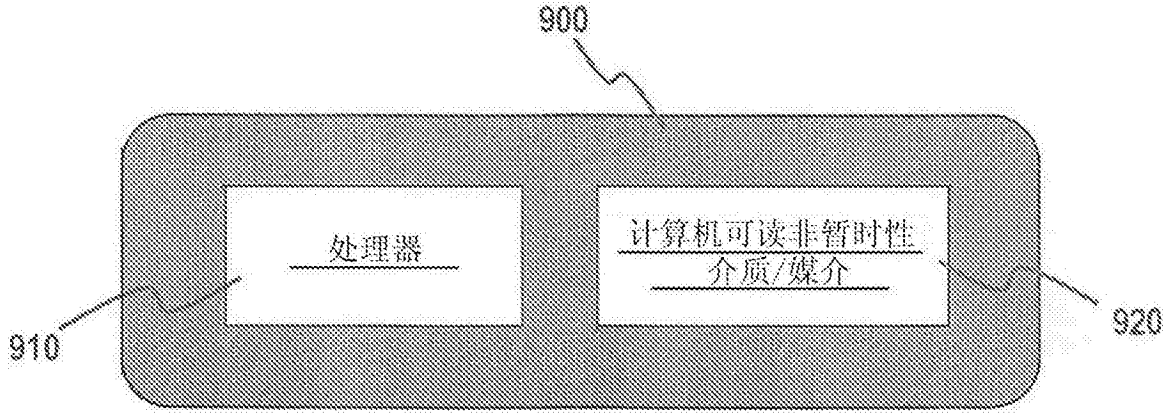


图9

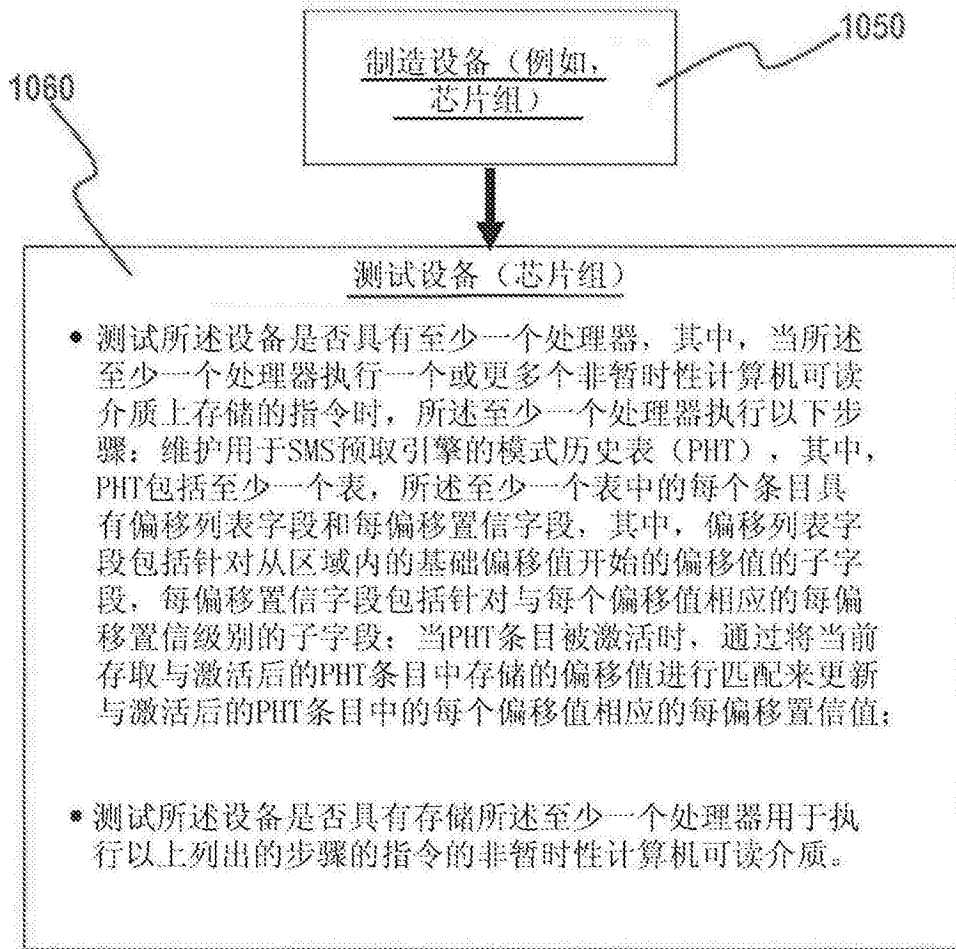


图10