



(12) 发明专利

(10) 授权公告号 CN 112543916 B

(45) 授权公告日 2022. 12. 06

(21) 申请号 201980052180.9
 (22) 申请日 2019.07.03
 (65) 同一申请的已公布的文献号
 申请公布号 CN 112543916 A
 (43) 申请公布日 2021.03.23
 (30) 优先权数据
 16/030,031 2018.07.09 US
 (85) PCT国际申请进入国家阶段日
 2021.02.05
 (86) PCT国际申请的申请数据
 PCT/US2019/040497 2019.07.03
 (87) PCT国际申请的公布数据
 W02020/014066 EN 2020.01.16

(73) 专利权人 超威半导体公司
 地址 美国加利福尼亚州
 (72) 发明人 托马斯·克卢库尔
 安东尼·贾维斯
 (74) 专利代理机构 上海胜康律师事务所 31263
 专利代理师 樊英如 邱晓敏
 (51) Int.Cl.
 G06F 9/38 (2006.01)
 (56) 对比文件
 CN 106681695 A, 2017.05.17
 审查员 刘璜

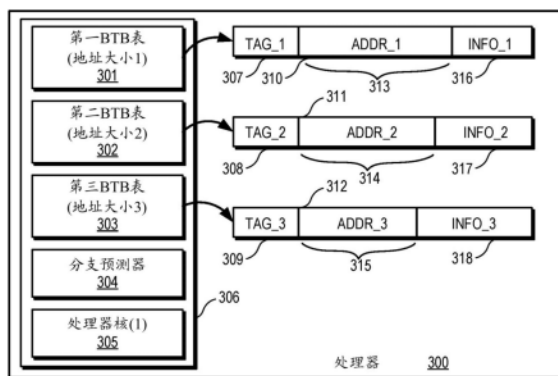
权利要求书1页 说明书17页 附图8页

(54) 发明名称

多表分支目标缓冲器

(57) 摘要

一种处理器[120],包括用于分支预测的两个或更多个分支目标缓冲器(BTB)表[301,302],每个BTB表存储不同目标大小或宽度的条目或存储不同分支类型的条目。每个BTB条目包括至少标签和目标地址。对于仅需要几个目标地址位的某些分支类型,相应BTB表较窄,从而允许按分支指令类型将所述处理器中的更多BTB条目分离到相应BTB表中。增加数量的可用BTB条目存储在处理器中相同或更少的空间中,从而提高指令处理速度。可定义不存储任何目标地址并依靠解码单元来提供所述目标地址的BTB表。



1. 一种处理器,其包括:

指令执行流水线;

第一分支目标缓冲器 (BTB) 表,所述第一分支目标缓冲器表具有第一数量的第一分支目标缓冲器表条目,每个第一分支目标缓冲器表条目包括第一标签和第一目标地址,每个第一目标地址具有第一宽度;

第二分支目标缓冲器表,所述第二分支目标缓冲器表具有第二数量的第二分支目标缓冲器表条目,每个第二分支目标缓冲器表条目包括第二标签和第二目标地址,每个第二目标地址具有不同于所述第一宽度的第二宽度;

分支预测器,所述分支预测器被配置为为给定预测地址提供预测目标地址;并且

其中所述处理器被配置为基于分支指令的分支特性将包括所述第一目标地址的分支描述符存储在所述第一分支目标缓冲器表的第一分支目标缓冲器表条目或所述第二分支目标缓冲器表的第二分支目标缓冲器表条目中。

2. 如权利要求1所述的处理器,其中所述分支特性是分支目标地址大小。

3. 如权利要求1所述的处理器,其中所述分支特性是分支类型。

4. 如权利要求3所述的处理器,其中所述第一分支目标缓冲器表条目被配置为存储用于跳转分支指令、调用分支指令、返回分支指令和条件分支指令中的至少一者的分支描述符。

5. 如权利要求1所述的处理器,其中所述分支特性是存储在所述第一分支目标缓冲器表条目或所述第二分支目标缓冲器表条目中的位的数量。

6. 如权利要求1所述的处理器,其中所述分支特性是所述第一分支目标缓冲器表条目或所述第二分支目标缓冲器表条目的线程标识符。

7. 如权利要求1所述的处理器,其还包括:

第一存储器级别高速缓存和第二存储器级别高速缓存,

其中所述第一分支目标缓冲器表和所述第二分支目标缓冲器表被包括在所述处理器的同一存储器级别高速缓存中。

8. 如权利要求1所述的处理器,其中:

所述第一分支目标缓冲器表的所述第一分支目标缓冲器表条目是N路关联的以用于预测查找;并且

所述第二分支目标缓冲器表的所述第二分支目标缓冲器表条目是M路关联的以用于预测查找,其中M和N大于或等于一并且M不同于N。

9. 如权利要求1所述的处理器,其中所述第一分支目标缓冲器表条目的所述第一数量不同于所述第二分支目标缓冲器表条目的所述第二数量。

10. 如权利要求1所述的处理器,其中:

所述第一分支目标缓冲器表条目中的每一个或所述第二分支目标缓冲器表条目中的每一个包括最近使用状态位,所述最近使用状态位在组内将相应第一分支目标缓冲器表条目或相应第二分支目标缓冲器表条目标记为最近最多使用 (MRU) 状态和最近最少使用 (LRU) 状态中的一者;并且

所述处理器被配置为基于所述最近使用状态位的状态根据更换策略逐出所述组内的第一分支目标缓冲器表条目或第二分支目标缓冲器表条目。

多表分支目标缓冲器

背景技术

[0001] 处理器中的指令流水线通过在多个流水线级处理指令来提高指令执行吞吐量,在多个流水线级中可并行执行指令流的不同指令。此类流水线通常包括单独单元,所述单独单元用于对指令进行提取、解码、映射和执行然后将结果写入另一单元(诸如寄存器)。流水线的指令提取单元将指令流提供给处理器流水线的下一级。指令提取单元通常使用指令高速缓存,以便保持连续地给流水线的其余部分供应指令。

[0002] 如果处理器一直等到分支在流水线中的执行级中得到解析才在指令提取级中提取下一条指令,则指令流中的分支指令可能会导致流水线停顿。分支预测器可尝试预测是否将采取条件分支。在一些实现方式中,在通过解码和执行分支指令本身来计算分支指令之前,分支预测器使用分支目标预测来预测所采取条件或无条件分支的目标。分支目标可基于与所计算地址的偏移量或通过寄存器的间接引用。

[0003] 常规上,分支目标缓冲器(BTB)是处理器中的存储包括所预测分支目标的分支信息的单个小存储器高速缓存。预测涉及将指令地址与已存储在BTB中的先前执行的指令地址进行比较。预测通常节省处理时间,因为成功的预测允许处理器跳过用于获取目标地址的步骤的执行。处理器通过在BTB中查找下一执行步骤的地址来节省时间。因此,BTB生成目标地址命中的频率直接影响处理器可执行指令的速度。通常,执行速度与BTB可存储的条目数量直接相关。

附图说明

[0004] 通过参看附图,可更好地理解本公开,并且本公开的众多特征和优点对本领域技术人员来说是显而易见的。在不同附图中使用相同的附图标记来指示类似或相同项。

[0005] 图1是根据一些实施方案的指令流水线体系结构的框图。

[0006] 图2是根据一些实施方案的处理系统的框图。

[0007] 图3是根据一些实施方案的具有基于目标大小的多个分支目标缓冲器(BTB)表的处理器的框图。

[0008] 图4是根据一些实施方案的具有基于分支类型的多个BTB表的处理器的框图。

[0009] 图5是根据一些实施方案的具有处于不同存储器级别中的多个BTB表的处理器的框图。

[0010] 图6是根据一些实施方案的BTB表的BTB条目的框图。

[0011] 图7是根据一些实施方案的BTB表的BTB条目的框图。

[0012] 图8是根据一些实施方案的BTB表的BTB条目的框图。

[0013] 图9是示出根据一些实施方案的用于按目标地址大小将分支目标地址存储到多个BTB表中的一个中的方法的流程图。

[0014] 图10是示出根据一些实施方案的用于按分支类型将分支目标地址存储到多个BTB表中的一个中的方法的流程图。

具体实施方式

[0015] 在处理器中使用分支目标缓冲器 (BTB) 或 BTB 表来存储分支信息显著提高处理器速度。常规上, 增加 BTB 中条目数量的唯一方式是增加缓冲器的大小。然而, 处理器内的存储元件数量存在限制, 包括对 BTB 的容量的限制, 因为增加一些组件 (诸如 BTB) 的本地存储会以处理器中其他组件的速度和物理空间为代价。此外, 在设计处理器的体系结构时, 需要权衡取舍并考虑各因素。例如, BTB 的一些改变实际上在将处理器投入操作时导致处理器更慢。在其他情况下, 更大的 BTB 导致处理器操作期间总体功耗更高, 这是不期望的。

[0016] 常规 BTB 的典型大小是 1024 个条目, 其中每个条目具有例如 20 个位用于目标地址。然而, 并非每个 BTB 条目都需要使用相同数量的位来在其中保存地址。例如, 对于某些地址, 分支预测器可能仅需要 20 个可用位中的 5 个至 10 个位, 但对于其他地址则可能需要全部 20 个位, 因为目标地址的最高有效位的一部分可在分支信息中编码为与预测地址相比不变。为了增加 BTB 的容量和实用性, 并且使 BTB 在处理器内的占用面积保持大致相同, 在至少一个实施方案中, 将 BTB 分成两个或更多个缓冲器或表。根据某些实施方案, 第一缓冲器被设定大小并指定用于短存储器地址, 诸如具有 6 个、8 个或其他数量的位的那些地址。第二缓冲器被设定大小并指定用于较长存储器地址, 诸如具有 14 个、16 个、18 个或 20 个位的那些地址。一些实现方式利用第三缓冲器, 所述第三缓冲器被设定大小并指定用于多种存储器地址长度, 诸如具有 6 个、8 个、10 个、12 个、14 个、16 个、18 个或 20 个位的那些地址。每个 BTB 条目还包括标签字段, 并且在一些实施方案中, 还包括放置到 BTB 条目中一个或多个单独字段中的其他信息。

[0017] 根据其他实施方案, 处理器设计者和体系结构师基于一种或多种指令分支类型来创建并指定多个 BTB 表, 并因此针对分支目标的相应类型创建 BTB 表。每个 BTB 条目具有与发现由 BTB 条目跟踪的分支的成本相关的值。无条件分支可在处理器流水线中由处理器解码器在早期发现, 而条件分支可能直到执行时间才被发现。处理器的一些实施方案在一个周期中递送两个或更多个分支预测, 并因此递送两个分支目标地址。这多个分支存储在单个 BTB 条目中。为了考虑 BTB 条目的各种值, 第一 BTB 例如被创建并指定用于无条件分支, 并且第二 BTB 例如被创建并指定用于单个条件分支, 并且第三 BTB 例如被创建用于记录多个分支的条目。第一 BTB 表、第二 BTB 表和第三 BTB 表中的每一者可具有不同目标地址大小并因此具有不同总体 BTB 表宽度, 从而提供与其中单个 BTB 具有特定固定宽度和特定固定长度 (条目数量) 的先前设计相比减少 BTB 表在处理器中的总体占用面积的机制。将 BTB 条目划分或放置到两个或更多个 BTB 表中的这种益处允许处理器设计者相对于彼此或相对于单个常规 BTB 的常规长度来增大或减小 BTB 表中的一个或多个的长度。对于三 BTB 系统, 第一 BTB 表、第二 BTB 表和第三 BTB 表可由每个 BTB 条目可指向的单独目标表进行补充, 以便为无法适应相应第一 BTB、第二 BTB 和第三 BTB 中的条目的大目标地址提供多个另外的地址位。在操作中, 并且根据一些实现方式, 针对每次 BTB 查找跨所有 BTB 缓冲器在处理器中执行 BTB 查找, 从而提供与单个常规 BTB 缓冲器类似的性能。

[0018] 在某些实现方式中, BTB 表包括在单个存储器类型内, 或者, 然而在其他实现方式中, BTB 表跨相对于处理器核的多个存储器、多个存储器类型或多级存储器层次结构的多个存储器级别分布。为了描述的简单起见, 在本文中参考各自对应于相对于处理器核或相对于处理器的存储器级别或布局的 1 级 BTB、2 级 BTB 等。每个 BTB 级别可包括单个 BTB 表或多个

BTB表。在其他实现方式中,可针对单个处理器核在单个存储器高速缓存或存储器级别中构建一个或多个BTB表。在另一些其他实现方式中,BTB表跨处理器核或处理器操作共享。在特定实施方案中,多个BTB级别中的一个由多个处理器核共享,其中处理器核中的每一个包括其自己的一个或多个BTB表。可首先使用与特定核相关联的那些BTB表,并且可视需要将BTB条目逐出到共享BTB表中。处理器具有跨处理器核共享的一个或多个BTB表的益处可被搜索速度、总体处理器速度和总体处理器操作的降低抵消。因此,可向多核处理器的每个单独处理器核提供多个BTB表以获得快速实现方式。

[0019] 在操作中,首先将较低值BTB条目(诸如最近最少使用(LRU)的那些条目)从第一BTB或第一BTB级别逐出到另一BTB或BTB级别(诸如位于同一或另一存储器级别中的BTB)。从第二BTB到第三BTB的逐出也基于LRU进行。各种类型的关联性可用于BTB表。举例来说,第一BTB表是四路组关联的。另一BTB表采取八路组关联处理器高速缓存的形式,其中每路存储512个、1024个、2048个条目或每路存储某一其他数量的条目。通常,第一BTB表可以是N路组关联的,并且第二BTB表可以是M路组关联的,其中N和M是彼此相同或不同的整数。在其他实施方案中,BTB表中的一个或多个是完全关联的。关联性水平根据特定处理器的期望功耗水平、处理器在其中运行的系统以及特定处理器的处理负荷的预期类型和量来选择。

[0020] 在一些实现方式中,BTB表用作牺牲者缓冲器,所述牺牲者缓冲器具有有限数量的条目,这些条目来自较低级BTB表的最近逐出的条件分支的条目填充。为了使这种牺牲者缓冲器针对给定区域的条目数量最大化,分支预测器不在BTB表中记录任何目标地址。当分支预测器在牺牲者缓冲器中找到分支并预测分支被采取(如由条件分支预测器所确定)时,分支预测器将关于分支位置的信息发送到解码单元,所述解码单元一旦完成解码操作,就触发到可用目标地址的重定向,并且确认所预测位置确实是条件分支。

[0021] 图1是根据一些实施方案的实施无序指令执行的处理器120的指令流水线体系结构100的框图。仅示出处理器120的少数元件。指令高速缓存101由指令提取单元103访问。数据高速缓存102由加载/存储单元110访问。指令高速缓存101的指令对包括来自数据高速缓存102的数据的数据进行操作。指令提取单元103包括分支预测器104,其生成分支目标地址,所述分支目标地址被存储或提供到一个或多个分支目标缓冲器(BTB)表(通常也称为BT缓冲器和BTB)105。根据一些实施方案,分支目标地址是相对于程序计数器113的。虽然BTB表105在图1中被示出为在分支预测器104内部,但BTB表105在处理器120中可靠近或不靠近分支预测器104或指令提取单元103的某些元件定位。解码器106将包括来自指令高速缓存101的那些指令的指令转换成处理器控制信号。

[0022] 重排序缓冲器107将指令按其原始提取顺序存储在由处理器的其他组件(诸如保留站109)访问的寄存器108中。保留站109对诸如寄存器108的寄存器进行重命名,并且促进动态指令调度。一旦已提取或计算数据,保留站109就允许处理器提取并重新使用数据,而不是等待将数据存储在寄存器中并重新读取。当由分支预测器104存储在BTB表105中的预测地址不正确或者在指令流中遇到不可恢复异常时,清除重排序缓冲器107中的所有指令,并且将保留站109重新初始化。重排序缓冲器107提供用于分支目标地址错误预测的回滚控制的机制。保留站109将信息提供给加载/存储单元110和一个或多个功能单元111,诸如算术逻辑单元(ALU)、浮点单元(FPU)和整数单元(IU)。指令提取单元103、解码器106、重排序缓冲器107、保留站109、加载/存储单元110和相关寄存器一起是指令执行流水线的的一个实

施方案。

[0023] 图2是根据一些实施方案的处理系统200的框图。处理系统200是图1的指令流水线体系结构100和BTB表105的一部分的实例。处理系统200包括分支预测器201,其具有分支检测器202、条件分支预测器203、返回地址预测器204和分支预测训练器205。处理系统200包括其他未示出元件,诸如如本领域技术人员所理解的算术单元、调度器、表巡行器等。

[0024] 对于每个当前地址,分支预测器201提供预测窗口,所述预测窗口包括要从当前地址开始提取的字节块的结束地址、所预测块中的分支的类型以及下一预测的起始地址。为了确定哪些分支是预测窗口的一部分,分支检测器202跨所有路在所有BTB表中查找,诸如跨表示为BTB表1 206至BTB表N 207的多个BTB表中的一个或多个。给定BTB条目上的标签匹配指示条目中记录的一个或多个分支存在于预测窗口中。每个BTB条目包括分支的位置和类型。当针对错误预测的分支发生重定向时,由分支预测训练器205填充BTB。

[0025] 图3是根据一些实施方案的具有多个BTB表的处理器的框图。处理器300是包括另外的结构和特征的图1的处理器120和BTB表105的具体实例。处理器300包括多个处理器核中位于第一指定区域306中的第一处理器核305。第一处理器核305与BTB表互操作,所述BTB表包括由分支预测器304使用的第一BTB表301、第二BTB表302和第三BTB表303。根据一些实施方案,每个过程周期,分支预测器304都根据一个或多个分支指令提供一个或两个预测。

[0026] 每个BTB表301、302、303分别包括用于标识条目所属于的预测地址的标签字段307、308、309;用于存储一个或多个分支的目标的目标地址字段310、311、312;以及用于存储分支在预测窗口中的位置和分支类型的信息字段316、317、318。分支训练器(诸如图2的分支预测训练器205)填充相应BTB表,并且针对正在训练的每个BTB条目确定是将BTB条目记录到第一BTB表301、第二BTB表302还是第三BTB表303中。

[0027] BTB表301-303中的每一个具有不同大小或宽度313、314、315的目标地址字段310、311、312(除非另外指明,否则“大小”和“宽度”在本文中可互换地使用)。例如,第二BTB表302具有第二地址大小314的地址311。根据一些实施方案,第二地址大小314小于第一地址大小313。第二地址大小314大于第三BTB表303的目标地址字段312的第三地址大小315。在某些实施方案中,与例如第一BTB表301的第一组BTB条目中的BTB条目的数量相比,第三BTB表303的第三组BTB条目中存在更多数量的BTB条目。这是可能的,因为当保持其他字段309、318的大小不变时,减少目标地址字段312的位数或宽度315允许第三BTB表303的更大数量的条目。即,对于每个BTB表301、302、303的相同量的处理器占用面积,减小的宽度315允许第三BTB表303针对固定大小的指定区域306具有更大数量的条目。在其他实施方案中,相对于单个BTB表针对所有目标地址具有恒定宽度的目标地址字段的先前实施方案,更小的指定区域306是可能的。因此,在此实例中,即使第三BTB表303在处理器300和指定区域306中相比其他BTB表301、302可能占据更小的占用面积,但也认为相比第二BTB表302和第一BTB表301,第三BTB表303的大小(条目)更大。

[0028] 当需要额外的位来诸如在第三BTB表303中的条目的第三地址312中或在第二BTB表302中的条目的第二地址311中保存地址位时,在第一BTB表301、第二BTB表302和第三BTB表303之间共享额外地址表(图3中未示出)。当使用额外地址表时,代替存储在地址字段的位中的地址值,将多个地址位索引到额外地址表中的条目。即,将指向额外地址表的条目的指针存储在BTB条目的指定用于分支目标地址字段310、311、312中的对应字段的至少一部

分中。

[0029] 图4是根据一些实施方案的具有基于分支类型的BTB表的处理器400的框图。基于分支类型组织BTB表通过将表专用于不同值的BTB条目并适当调整表的大小以将更多空间专用于更高值的条目来提供改进的性能。处理器400是包括另外的结构和特征的图1的处理器120和BTB表105的具体实例。处理器400包括多个处理器核中位于第一指定区域406中的第一处理器核405。第一处理器核405与BTB表互操作,所述BTB表包括由分支预测器404使用的的第一BTB表401、第二BTB表402和第三BTB表403。根据一些实施方案,每个过程周期,分支预测器404都根据一个或多个分支指令提供一个或两个预测。

[0030] 每个BTB表401、402、403包括用于标识条目所属于的预测地址的标签字段407、408、409;用于存储一个或多个分支的目标的目标地址字段410、411、412;以及用于存储分支在预测窗口中的位置和分支类型的信息字段416、417、418。分支训练器(诸如图2的分支预测训练器205)填充相应BTB表,并且针对正在训练的每个BTB条目确定是将BTB条目记录到第一BTB表401、第二BTB表402还是第三BTB表403中。此外,BTB表401-403中的每一个具有相应分支类型的目标地址字段410、411、412。BTB表401-403中的每一个可具有基于对于存储在其中的一种或多种相应类型的分支地址来说典型的目标地址大小的不同大小或宽度413、414、415的目标地址字段410、411、412。BTB表401、402、403中的每一个被指定用于一种或多种分支类型。例如,第二BTB表402存储与第一BTB表401的条目不同的分支类型的地址411。在某些实施方案中,与例如第二BTB表402的一组BTB条目中的BTB条目的数量相比,第三BTB表403的第三组BTB条目中存在更多数量的BTB条目,因为就正确地对BTB条目进行预测时保存的周期数量而言,第三BTB表403中条目的类型更有益于处理器400的操作效率。相应BTB表401-403的大小(条目数量)的选择和BTB表401-403的构建是基于要存储在其中的分支地址类型以及它们相对于彼此的操作效率。在一个实施方案中,每个BTB条目包括至少处于相应字段中的标签和目标地址。每个条目还可包括其他信息,诸如线程ID和共享指示符或标志。

[0031] 图5是根据一些实施方案的具有处于不同存储器级别中的多个BTB表的处理器的框图。处理器500包括处理器核501,所述处理器核501包括处理分支指令的分支预测器502。分支预测器502包括各种结构并与它们互操作,所述结构包括1级BTB表503、一组2级BTB表504、505、506以及3级BTB表507。1级BTB表503位于第一级存储器508中。2级BTB表504、505、506位于第二级存储器509中。3级BTB表507位于第三级存储器510中。根据一些实施方案,当一组1级BTB表503已满或所有组BTB条目已满时,覆写BTB条目或将其从1级BTB表503逐出到多个或一组2级BTB表504-506中的一个中。2级BTB表504-506中的每一个是针对特定目标地址大小或目标地址大小范围或者针对特定分支指令类型创建的,如关于其他附图所描述的。

[0032] 当2级BTB表504、505、506中的一个已满时,处理器500覆写其中的BTB条目或将其中的BTB条目逐出到第三级存储器510中的3级BTB表507中。在一些实施方案中,覆写BTB条目(诸如最低价值条目)。在其他实施方案中,将BTB条目逐出到未示出的一个或多个BTB表中的一个。这样的其他BTB表可位于同一存储器或不同存储器(诸如另一级别的存储器508-510中的BTB表)中,或者位于处理器中的另一位置中,或者位于处理器核501之外的存储器中,或者位于处理器500本身中诸如位于通过总线或桥接器耦接到处理器500的存储器中。

例如,逐出是到2级存储器509中的溢出BTB表。根据一些实施方案,允许基于分支类型或分支目标地址大小逐出仅BTB条目的子集,诸如仅一个BTB表504、505、506中的那些BTB条目。

[0033] 根据说明性实施方案,2级BTB表504、505、506中的每一个位于2级存储器509中。在一些实施方案中,BTB表504、505、506中的每一个包括四路关联的1,024个条目。每个BTB条目在大小上为68个位(68b),并且可记录多达37个位的目标大小。大小大于37个位的目标可通过间接级别记录到单独表中。本文描述的大小说明可能的多种大小(诸如图3的BTB表301-303、图4的BTB表401-403以及图5的BTB表503-507)。在一些实施方案中,使用虚拟提取地址的散列来对BTB表进行索引和标记。

[0034] 根据一些实施方案,BTB表支持每个分支预测周期预测多达两个分支;在其他实施方案中,每个过程循环或分支预测周期评估仅一个分支。优选地,BTB表支持所有分支类型和目标大小。每个BTB条目(诸如图5的1级BTB表503中的那些条目)针对预测块存储多达两个目标及伴随分支信息,所述预测块从预测地址开始并且延伸到对齐的64B的末尾、或第一静态分支的末尾、或第一动态分支或预测窗口中的“非最后”分支对(NLIPW)的末尾(以先到者为准)。

[0035] 根据一些实施方案并且取决于处理器设计,BTB表中的每个条目包括某些字段。为了说明,图6是根据一些实施方案的BTB表的BTB条目600的框图。在图6中,以下是对BTB表的68位条目的BTB条目600内的相应位置位处的字段的描述,每个BTB条目600具有从零(0)到67索引的68个位。位的大小或数量仅作为实例,并且可使用本文所提供的指南来实现其他大小和位分布。AUXPREDEN 601是位置[67]处的一个位,并且指示辅助预测器的启用。SHRD 602是位置[66]处的一个位,其指示条目是否是在线程之间共享的或可共享的。TID 603是位置[65]处的一个位,并且指示线程标识符(ID)。TAG 604是位置[64:52]处的13个位,并且存储分支条目的标签。LE 605是位于位置[51:50]处的两个位,并且指示编码长度或“长度编码”。LE 605状态包括具有以下这些值的状态:1,1;1,0;和0,1。PAIR 606是位置[49]处的一个位,并且指示与溢出地址缓冲器的配对。BRTYPE 607是位于位置[48:44]内的5个位,并且指示分支类型编码,如下文更详细地描述。BR1END 608是位于位置[43:38]处的6个位,并且指示可以[5:0]的范围递增一的Branch1结束指针。

[0036] LE 605和PAIR 606的状态一起指示每个(第一级或第二级)BTB条目的位置[37:0]处的位的使用方式。USEOVERFLOW (USEOA) 609是位置[37]处的一个位,并且是指溢出地址缓冲器的使用。例如,位于位置[48:32]处的目标来自通过溢出地址标识符OAIDX 610(参见下文)指向的溢出地址缓冲器条目,并且在PAIR=0时使用。OAIDX 610是位于位置[36:32]处的5个位,并且是指溢出地址缓冲器的索引。OAIDX 610在PAIR=0且USEOA=1时使用。BROEND 611是位置[37:32]处的六个位的替代分配612,并且指示可以[5:0]的范围递增一的Branch0结束指针。

[0037] BR0TGT 613是索引(X+1)到31的第一范围615内的Branch0目标或第一分支目标地址,其中X是整数并且第一分支目标地址具有取决于LE 605的两个位的状态的可变长度。例如,对于LE=1,1,BR0TGT 613是位置[31:20]处的12个位;对于LE=1,0,BR0TGT 613是位置[31:16]处的16个位;并且对于LE=0,1,BR0TGT 613是[31:12]位置处的20个位。当LE=0,0时,BR0TGT 613无效或不使用。

[0038] BR1TGT 614是从位置索引[0]到位置索引[X]的第二范围616内的Branch1目标或

第二分支目标地址,其中X是整数。例如,当PAIR=0且USEOA=0时,BR1TGT 614是位置[36:0]处的37个位;当PAIR=0且USEOA=1时,BR1TGT 614是位置[31:0]处的32个位;当PAIR=1且LE=1,1时,BR1TGT 614是位置[19:0]处的20个位(BROTGT613为12个位,如上所述);当PAIR=1且LE=1,0时,BR1TGT 614是位置[15:0]处的16个位(BROTGT 613为16个位,如上所述);当PAIR=1且LE=0,1时,BR1TGT 614是位置[11:0]处的12个位(BROTGT 613为20个位,如上所述)。以此方式,根据一些实施方案,BTB表的每个1级、2级或3级的BTB条目能够存储两个分支目标地址BROTGT 613和BR1TGT 614。

[0039] 关于分支类型编码,根据一些实施方案,5位分支类型指示以下Branch1特性。对于五个分支类型位中的前两个位,每个分支指令的分支类型如下在位置[1:0]处指定:值为0,0的第一状态对应于无效(Invalid);值为0,1的第二状态对应于JMP(跳转类型);值为1,0的第三状态对应于CALL(调用类型);并且值为1,1的第四状态对应于RET(返回类型)。在5位分支类型的位置[2]处,位针对值0指定固定目标,并且针对值1指定可变目标。默认情况下,分支被安装为固定目标,并且仅间接分支会在以下情况下转换为可变目标:分支所达到的目标不同于最初安装到BTB条目中的目标。

[0040] 对于5位分支类型的最后两个位,根据以下方案在位置[4:3]处提供方向性状态。直接固定分支和间接固定分支通过查看位置[4:3]来辨别。值0,0对应于通过DE(解码)重定向找到的无条件直接(UD)。根据一些实施方案,UD按定义是静态的并且是固定的。需要针对UD进行区别,以便知晓分支属于2级BTB的哪个表。位置[4:3]处的值1,0对应于非UD、静态。默认情况下,分支就方向而言被安装为静态的,并且如果条件分支被重定向为未被采取,则仅将条件分支转换到动态方向。

[0041] 位置[4:3]处的值1,1对应于“非UD、动态、预测窗口中最后一个(LIPW)”的状态。LIPW是针对处理器标识为64B高速缓存行中的最后一个分支的任何分支设置的。这种状态有助于BTB读取逻辑在BTB条目中的任何分支都未被采取的情况下决定下一提取地址(FA)。如果LIPW已设置(并且假设未采取任何分支),则下一FA是顺序高速缓存行地址,否则下一FA是紧接在BTB条目中最后一个分支之后的指令地址。位置[4:3]处的值0,1的状态对应于“非UD、动态NLIPW”。

[0042] 由于分支目标配对仅在第一分支可被预测为未被采取时才具有值,因此Branch0只能是动态固定JMP类型的分支。在一个实施方案中,分支目标对具有以下特性。每对仅一个分支可具有可变目标。对不能包括Call/Call、Call/Ret、Ret/Call或Ret/Ret。对支持如前所述的多种目标大小组合。为了知晓配对是否可能以及使用哪种组合,单一类型区分短、中和长分支预测目标。所述区分还用于选择BTB表中的哪个BTB表来在将来自另一BTB表的分支目标地址牺牲或逐出到诸如溢出2级BTB表或3级BTB表时使用。

[0043] 以下描述提供根据说明性实施方案的关于第一BTB表(诸如图4的BTB表401、402、403和图5的BTB表503-507)的第一BTB分支目标地址位的进一步细节。根据一些实施方案,需要目标字段的32个位以便存储匹配目标地址的两个分支并且诸如基于分支类型覆盖两个分支的许多可能情况。所需的目标大小跨直接分支从1个位到49个位不等,而返回类型分支不需要任何目标位。对于可变目标分支,如本文进一步描述地在BTB中训练所遇到的第一目标,并且在间接目标预测器中训练其他目标。

[0044] 当条目存储两个分支目标地址时,目标字段在各自通常具有不同长度的目标之间

共享。由于从时序角度来看实现可变长度分支的成本太高,因此存储目标的更优化方法是针对两个分支使用三个目标长度变型,并且在相应BTB表(诸如图4的BTB表401、402、403和图5的504、505、506的BTB表)中使用分支类型(诸如跳转、调用和返回类型)进行编码。存在以下情况:分支目标太大,以至于其目标无法与第二分支的目标并排放置。在这种情况下,分支需要其自己的BTB条目来存储完整的目标地址或目标位。存在两种情况:(1) 每条目存储单个分支,如上所述,将另一分支类型字段和结束字段重新用作目标位。此机制支持37位目标地址;以及(2) 长度超过37位的目标,其中高位由0AIDX字段所指向的溢出地址条目提供。在命中第一BTB之后对需要溢出地址结构的目标地址的重定向引起额外的重定向惩罚。作为实例,对于BTB更换策略,根据一些实施方案,第一BTB表401或BTB表503使用轮询(round robin)方案作为更换策略的一部分。第一BTB表中的256组中的每一个包括两位轮询(RR)计数器。

[0045] 就流水线操作而言,诸如在图1的指令流水线体系结构100、图4的处理器400或图5的处理器500中,针对每个预测流读取BTB表,除非为了节能而抑制读取。根据一些实施方案,第一BTB读取在表示为BP0的第一步骤中启动。后续级被设计为BP1、BP2等,作为预测流水线中的级。BTB表从BP2重定向到BP0。

[0046] 根据处理器(诸如图4的处理器400和图5的处理器500)的一些实施方案,存在三个逻辑表,如体现在如上所述的BTB表401、402、403和表504、505、506中。BTB表中的每个表支持如图4中的分支类型或如图3中的目标大小的子集。体现在BTB表401-403、503-507中的此类多表BTB提供若干益处。例如,这种布置在处理器中是面积有效的。每个BTB表401-403、504-506仅具有用于其分支类型或目标大小子集的存储容量。在一些实施方案中,每个BTB表提供大量关联性,但在某些实施方案中,每个BTB表401-403、504-506可实现全LRU。使用高实际关联性实现起来成本高昂。多个BTB表的使用将较高值分支(例如,成对分支和动态分支)与其他分支分开。多个BTB表301-303、401-403、503-507支持无限的分支目标大小,但多个BTB表301-303、401-403中的仅一个BTB表需要支持全分支目标地址大小。多个BTB表的使用促进处理器平面图灵活性。例如,BTB表401-403中的仅两个需要支持动态分支。因此,这些BTB表可靠近条件预测器放置。在操作中并且根据处理器的某些实施方案,BTB表中的仅两个驱动大量目标地址位。

[0047] 以下提供关于说明性BTB组织和大小调整方案的进一步细节。第一BTB表(索引0)(诸如第一BTB表401)存储小条目。第二BTB表(索引1)(诸如第二BTB表402)存储通过DE重定向发现的分支(例如,具有低重定向惩罚的那些分支)。第三BTB表(索引2)(诸如第三BTB表403)存储所有其他类型。例如,分支类型分配如下。第一BTB表存储单个RET分支类型和单个静态固定条件JMP短分支类型的地址。第二BTB表存储单个静态固定无条件直接调用非“超长”分支类型和单个静态固定无条件直接跳转非超长分支类型。第三BTB表存储所有其他分支类型。除了相对于图3至图5所描述的那些之外,其他数量的BTB表以及其他类型和目标大小分配也是可能的。

[0048] 在具有总计5632个BTB条目的示例性实现方式中,BTB表的容量实例如下。第一BTB表具有512个组和4个路,总计2048个BTB条目。第一BTB条目每条目具有36个位,小计73728个位。第二BTB表具有256个组和4个路,总计1024个BTB条目。第二BTB条目每条目具有64个位,小计65536个位。第三BTB表具有512个组和5个路,总共2560个BTB条目。第三BTB条目每

个条目具有72个位,小计184320个位。总共,这组BTB表具有13个路、5632个条目,并且需要一个或多个指定区域(诸如图4的指定区域406)中的323584个位诸如元素。

[0049] 在此实例中,对于多表BTB表,在图4的一组多个BTB表401-403或图5的一组BTB表504-506之间共享溢出地址缓冲器。分支类型编码在第一BTB表401、504、第二BTB表402、505和第三BTB表403、506之间是不同的。然而,在根据{LE,PAIR,TYPE}定义的BTB分支类型与根据{TableNumber,LE,PAIR,TYPE}描述的BTB表分支类型之间存在直接映射。这些BTB描述(字段)在交换和BTB逐出期间使用。

[0050] 根据其他实施方案,对于第一BTB表(诸如图4的第一BTB表401)中的BTB条目的字段和位置位,这些字段和位置位包括:UNUSED,其具有位置[35:34]处的两个位;SHRD,其具有位置[33]处的一个位;TID,其具有位置[32]处的一个位;TAG,其具有位置[31:20]处的12个位;BREND,其具有位置[19:14]处的六个位;BRTYPE,其具有位置[13:12]处的两个位;BRTGT,其具有位置[11:0]处的十二个位。UNUSED字段指示在相应宏位置中是否存在未使用的位。SHRD字段指示BTB条目在线程之间是否是共享的或可共享的。TID字段是线程ID。BREND字段是可以[5:0]的范围递增一的分支结束指针。BRTYPE字段指示分支类型编码。根据一些实施方案,第一BTB表401包括若干类型的分支(包括RET(返回)类型分支和至少一种类型的JMP分支)的信息。根据一些实施方案,BRTYPE字段编码如下:状态0,X(其中X可以是0或1)对应于无效状态;值为1,0的状态对应于JUMP;以及值为1,1的状态对应于RET,其中位置[4:0]处的BRTGT在,从而记录返回类型的全分支类型。

[0051] 图7是根据一些实施方案的第二BTB表的BTB条目700的框图。以下是对第二BTB表(诸如图4的第二BTB表402或图5的第二BTB表505)中的BTB条目的字段和位置位的描述。图7的每个BTB条目700是64位宽,这短于图6的BTB条目600。在图7中,BTB条目700包括以下字段:SHRD 701,其具有位置[63]处的一个位;TID 702,其具有位置[62]处的一个位;TAG 703,其具有位置[61:49]处的十三个位;LE 704,其具有位置[48:47]处的两个位;BRTYPE 705,其具有[46:45]位置处的两个位;BREND 706,其具有位置[44:39]处的六个位;BRTGT 707,其具有[38:0]位置处的39个位。SHRD 701指示BTB条目是否是在线程之间共享的或可共享的。TID 702是线程ID。LE 704是指长度编码,并且使用与第一BTB表(诸如第一BTB表401)的LE 605相同的编码。BRTYPE 705指示分支类型编码。根据一些实施方案,第二分支类型BTB包括仅一种分支类型(诸如仅无条件直接分支)的信息。此字段用于编码JMP、CALL和无效之间的区别。根据一些实施方案,第二分支类型BTB的BRTYPE编码如下:值为0,0的状态对应于无效状态;值为0,1的状态对应于JMP(跳转)分支类型;状态1,0对应于CALL分支类型;并且值为1,1的状态对应于另一非法状态。第二BTB条目700的BREND 706是以[5:0]的范围递增一的分支结束指针。BRTGT 707是遵循与第一BTB条目600相同或类似格式的分支目标。需要在溢出地址缓冲器中可用的额外地址位的分支被指派给第三分支类型BTB,如在图8的BTB条目800中。因此,根据一些实施方案,在第二分支类型BTB中不包括1个位的USE0A字段,以适应对额外地址位的需要。如本文所提及,0A是指溢出地址或其中的特定字段、位或标志。

[0052] 图8是根据一些实施方案的第三BTB表的第三BTB条目800的框图。以下是对第三分支类型BTB条目800(诸如图4的第三BTB表403中的BTB条目或图5的第三BTB表506)中的条目的字段和位置位的描述。每个BTB条目800是从零(0)到71索引的72位宽,并且包括以下字

段:UNUSED 801,其具有位置[71]处的一个位;AUXPREDEN 802,其具有位置[70]处的一个位。SHRD 803,其具有位置[69]处的一个位;TID 804,其具有位置[68]处的一个位;TAG 805,其具有位置[67:56]处的十二个位;LE 806,其具有位置[55:54]处的两个位;BR1LBIAS 807,其具有位置[53:52]处的两个位;PAIR 808,其具有位置[51]处的一个位;BRTYPE 809,其具有位置[50:46]处的五个位;BR1END 810,其具有位置[45:40]处的六个位;以及BROLBIAS 811,其具有位置[39:38]处的两个位。字段群组{BREND 812,BRTGT 814}跨位置[37:0]处的相应可变长度范围813、815分布。例如,BREND 812从(Y+1)到37定位,并且BRTGT 814从0到(Y)定位,其中Y是整数。UNUSED 801指示在对应宏位置中是否存在未使用的位。AUXPREDEN 802指示辅助预测器(诸如环路退出预测器)的启用。SHRD 803指示BTB条目是否是在线程之间共享的或可共享的。TID 804是线程ID。LE 806是指长度编码,并且使用与第一BTB表(诸如第一BTB表401)的相同LE字段相同的编码,并且如上关于图6的LE 605所述。在图8中,BR1LBIAS 807指示Branch1是否存在局部偏置。如本文其他地方进一步描述地使用PAIR 808。根据一些实施方案,第三分支类型BTB条目800的BRTYPE 809如针对其他BTB表(诸如第一BTB条目600)的BRTYPE所述地进行编码。第三分支类型BTB条目800的BR1END 810是Branch1的分支结束指针,其以[5:0]的范围递增一。BROLBIAS 811指示Branch0是否存在局部偏置。BREND 812和BRTGT 814分别用于存储Branch0和Branch1的一个或两个分支目标连同来自第一BTB表(诸如第一BTB表401)的字段USEOA、OAIDX、BROEND、BR1TGT和BROTGT。参考第一、第二和第三分支类型BTB表401-403以及相应BTB条目600、700、800所描述的位数(宽度)和位置仅是示例性的。当在处理器或系统中实现多个BTB表时,所描述字段或者其他变量或字段的其他位数和位置也是可能的。

[0053] 根据说明性的多个BTB表更换策略,提供以下作为若干种可能的实施方案中的一个。在此实施方案中,第一级BTB表首先接受所有BTB条目,直到一组条目中的一个或多个已满为止,其中在此实例中的组是指一组组相关联条目群组。在另一实例中,第一级BTB表首先接受所有BTB条目,直到1级BTB表已满为止。然后,随着处理器或处理器核操作,BTB条目被逐出到基于目标地址大小或分支类型设计的多个2级BTB表(相应地诸如图3的BTB表301-303和图4的BTB表401-403)中的一个。例如,更换策略是针对每个2级BTB表(诸如图3的BTB表301-303和图4的BTB表401-403)的全LRU。分支类型BTB更换策略支持将条目标记为最近最少使用(LRU)或最近最多使用(MRU)。在将组的路清除之后将所述路标记为LRU,使得下次将新的BTB条目写入所述组时,空出的条目被更换,并且不覆写任何有效条目。在已安装路之后将路标记为MRU,使得路进入队列的后方进行更换。下表1详述以下情况:1级是指1级存储器,并且2级是指2级存储器,依此类推:

[0054] 表1

事件	新值	更新
2 级 BTB 命中(1 级 BTB 未命中)	MRU	如果更换为 1 级 BTB 牺牲者, 则更新命中条目
2 级 BTB 命中(1 级 BTB 未命中)	MRU	如果不同于 2 级 BTB 命中位置, 则更新 1 级 BTB 牺牲者的 2 级位置
[0055] 2 级 BTB 命中(1 级 BTB 未命中)	LRU	如果未更换为 1 级 BTB 牺牲者, 则更新命中条目(也使 2 级 BTB 中的命中条目无效)
1 级 BTB 和 2 级 BTB 命中	LRU	如果在 1 级 BTB 和 2 级 BTB 中命中, 则更新 2 级位置(也使 2 级 BTB 中的命中条目无效)
训练检查	MRU	为逐出的 1 级 BTB 条目更新 2 级 BTB

[0056] 以下描述提供根据一些实施方案的关于流水线操作的进一步细节。除非通过功率滤波器取消, 否则将为每个预测流读取分支类型BTB。读取在第一位置BP0中启动, 并且从BP4重定向到BPN1, 其中N是指后续周期。1级BTB未命中/2级BTB命中交换情况的流水线时序在下表2中示出。记录了来自1级BTB的牺牲者路和来自2级BTB的命中路。牺牲者路可具有与命中路不同的2级BTB表和存储库, 这需要特别考虑。例如, 2级牺牲者存储库基于LRU状态来选择并将其覆写。对1级BTB和2级BTB的写入在写入启用(WrEn)信号在BP4中断言的情况下发生, 其中实际写入位置为BP5。

[0057] 表2

BP 0	BP1	BP2	BP3	BP4	BP5
--	--	1 级 BTB 未命中	2 级 BTB 命中	1 级 BTB WrEn	Wr 到 1 级 BTB
--	--	1 级 BTB 牺牲者	--	2 级 BTB WrEn	Wr 到 2 级 BTB

[0059] 以下描述提供根据一些实施方案的关于处理器(诸如图4中的处理器400)中的2级BTB物理组织的进一步细节。第一2级BTB表包括具有4个路的512个组、每个条目36个位、每个组2个宏、72b的宏类型以及总计8个宏。第二2级BTB表包括具有4个路的256个组、每个条目64个位、每个组4个宏、64b的宏类型以及总计8个宏。第三2级BTB表包括具有5个路的512个组、每个条目72个位、每个组5个宏、72b的宏类型以及总计20个宏。

[0060] 索引的一个或多个高位用作读取使能, 使得对于第一2级BTB表每个路读取仅宏的一半, 并且对于第二和第三2级BTB表每个路读取一个宏。由于第一和第三2级BTB表具有的数量是1级BTB表(诸如图4的1级BTB表401)的两倍, 因此, 标签的低位用作这些BTB表的索引

最高有效位 (MSB), 使得BTB表仅需要存储12位标签。

[0061] 在用于相对较小客户端装置或移动电话的处理器实施方案中, 使用多个分支类型BTB表相对于性能而言具有较高功耗, 因此可执行另外的方法来减少功耗。例如, 静态方法包括在一些或全部时间内对2级BTB表进行功率门控并且在一些或全部时间内对2级BTB表进行时钟门控。在每个步骤中, 2级BTB使能功能可诸如通过BIOS或熔断器设置。用于减少功耗的动态方法包括通过识别何时需要特定2级BTB表的额外控制来自适应地对2级BTB表进行功率门控。另一动态方法包括调试由操作系统、固件等执行的应用程序, 使得应用程序根据应用程序对电力的优选使用或根据操作系统、固件或应用程序在其上活动的装置的功率设置来调试其对2级BTB表的使用。

[0062] 以下描述提供关于供多个BTB表 (诸如图3的第一BTB表301、第二BTB表302和第三BTB表303以及图4中的对应类似表) 使用的溢出地址缓冲器的进一步细节。为了在不将每个位存储在预测结构中的情况下支持多达49个位的目标大小的分支, 在处理器中提供溢出地址缓冲器。溢出地址缓冲器用于存储特定数量的位, 诸如例如分支目标地址的前17个位或17位组。根据一些实施方案, 由预测器结构 (诸如BTB表) 引用的前17个位是通过指针进入溢出地址缓冲器的, 从而节省BTB表中的存储空间。当需要存储大量目标地址位时, 使用溢出地址缓冲器。根据测试, 轨迹分析表明, 对于给定轨迹, 提取地址的前17个位仅采取指令组中各种分支的有限数量的值。

[0063] 根据一些实施方案, 溢出地址 (0A) 缓冲器包括32个条目, 每个条目在位置[48:32]处的BTB条目中的位置中包含虚拟地址的单个字段, 如本文所述。每个溢出地址缓冲器条目在线程之间共享。在预测时间处, 溢出地址缓冲器基于BTB提供的5位指针向BTB提供地址的前17个位。可能的是, 在预测时间处所使用的溢出地址缓冲器条目已被覆写, 因为指针记录在预测器中的一个中。在这种情况下, 预测是错误的并且被重定向, 这导致预测器被校正。不同预测器在预测管道的不同级读取溢出地址缓冲器, 从而需要来自溢出地址缓冲器的多次同时读取。

[0064] 溢出地址缓冲器在BTB的训练期间分配如下。训练首先确定分支目标是否与范围[48:32]的当前地址不同, 在这种情况下, 需要溢出地址缓冲器。如果需要溢出地址缓冲器, 则将当前溢出地址缓冲器条目与位置[48:32]处的目标地址进行比较, 以确定溢出地址缓冲器条目是否已包括位置[48:32]处的期望目标地址。在匹配时, 匹配的溢出地址缓冲器索引记录在预测器BTB表中。如果没有溢出地址缓冲器条目匹配位置[48:32]处的目标, 则使用全LRU更换策略在溢出地址缓冲器中分配新条目。根据一些实施方案, 训练写入是推测性的, 并且一些溢出地址缓冲器条目可被分配, 因为它们是在不良路径上被分配的, 所以它们从未被使用。

[0065] 以下描述提供关于BTB训练的进一步细节。当预测被重定向时 (诸如在预测流水线的解码和执行级处), 当在程序序列中再次遇到预测时, BTB表通过训练算法进行更新以提高预测地址处的预测准确性。当接收到重定向时, 使用来自重定向源的重定向类型、分支目标和EndAddr以及来自预测历史缓冲器的预测分支类型、分支目标、EndAddr和其他状态来确定需要哪种训练/更新动作。

[0066] 以下重定向情况需要训练BTB表或BTB表组。情况1: 所采取的预测分支不匹配任何分支, 对应于使在预测时所使用的BTB条目无效。情况2: 没有预测分支对应于训练在BTB中

新发现的分支。情况3:重定向分支EndAddr不匹配任何预测分支的EndAddr,并且对应于训练在BTB中新发现的分支。情况4:重定向分支EndAddr匹配预测分支的EndAddr但不匹配Type,并且对应于更新BTB中的分支类型。此第四种情况包括需要改变分支类型以将分支标记为动态或可变的情况。

[0067] BTB训练过程是通过训练流水线进行的。每个线程与一个错误预测寄存器相关联,所述错误预测寄存器捕获关于线程最后一次重定向的信息,并且指示是否请求BTB更新。训练管道基于请求在两个线程之间进行选取,并且在两个线程同时请求BTB训练时发生请求冲突的情况下通过轮询进行仲裁。一般来说,一个线程的训练可在另一线程已处于训练管道中时开始。然而,线程无法在另一线程当前正在针对同一BTB索引进行训练并且无法及时更新BTB表以使第二次训练在同一索引下可见的情况下进行选取。BTB训练是推测性地发生,并且在错误路径上发生的重定向可造成BTB更新。然而,如果错误预测寄存器被覆写,则取消针对线程正在进行的训练。

[0068] 训练管道以授予训练请求的周期BTN2开始,接着是周期BTN1、周期BT0、周期BT1、周期BT2、周期BT3、周期BT4,并且在写入BTB的周期BT5中完成。根据一些实施方案,仅第一BTB表(诸如图4的第一BTB表401)可用新的或修改的条目更新。训练管道启动训练检查读取(TCR)过程,所述过程将预测管道停顿一个周期。TCR过程的一个目的是在训练查找位置查明当前在1级BTB表或2级BTB表中的一个中是否存在条目。如果在2级BTB表中找到匹配,则TCR过程触发2级BTB与1级BTB之间的交换,此后重复训练流程。

[0069] 跟随在2级BTB中未命中的TCR过程,不同训练任务处理如下。仅当TCR命中时才执行无效。对于新的写入和更新,BTB更新训练将错误预测的分支与BTB条目命中合并,创建新的BTB条目,或者执行这两项动作。不同情形如下。当TCR进程在1级BTB中未命中时,诸如通过基于更换策略选择路并将所选择条目逐出到2级BTB(如果适用)来在查找地址处创建新的BTB条目。在一些实施方案中,当考虑未命中并执行如本文所述的后续步骤时,TCR过程检查三个级别的BTB表的全部三个级别的条目。

[0070] 当TCR流程命中1级BTB时(诸如在BTB表中命中),查找地址处存在现有BTB条目,这意味着此BTB条目中已存在一个或两个分支。如果训练任务是更新,则错误预测EndAddr匹配找到的分支中的一个。否则,需要引入新的分支。在BTB中存在需要写回的多达三个分支。这些分支基于EndAddr进行排序并且标记为Pos0、Pos1和Pos2。以下情况处理如下。情况1发生在仅Pos0有效时。然后在TCR查找地址处执行Pos0分支训练。情况2发生在仅Pos0和Pos1有效时。如果Pos0和Pos1分支可配对,则在TCR查找地址处执行配对训练。否则,仅Pos0在TCR查找地址处被训练为预测窗口中的“非最后”(NLIPW),并且当Pos1是错误预测的分支时,在Pos0EndAddr+1查找地址处启动Pos1的新训练流程。如果Pos1不是错误预测的分支,则放弃训练。情况3发生在Pos0、Pos1和Pos2都有效时。如果Pos0分支和Pos1分支可配对,则将TCR查找地址处的对训练为NLIPW,并且如果Pos2是错误预测的分支,则在Pos1EndAddr+1查找地址处启动Pos2的新训练流程。如果Pos2不是错误预测的分支,则将其放弃。否则,仅Pos0在TCR查找地址处训练为NLIPW。如果Pos0不是错误预测的分支,则在错误预测分支的Pos0EndAddr+1查找地址处启动新的训练流程。同样,Pos1、Pos2或Pos1和Pos2两者被放弃。

[0071] 上述训练过程尝试将所有分支保持在错误预测的分支之前,并且使用TCR流程安装错误预测的分支。每个重定向被限制为触发不超过两个训练流程,因此不能保证对预测

错误的分支进行训练。然而,标记条目NLIPW致使下一预测将预测窗口分成若干预测,使得新分支最终可得到训练。每个查找地址处的TCR流程触发1级/2级BTB交换,并且致使TCR流程重放。因此,对于给定重定向,可发生多达四个TCR流程。

[0072] 图9是根据一些实施方案的用于按目标地址大小将分支目标地址存储到多个BTB表中的一个中的方法900。方法900在图1所示的指令流水线体系结构100的一些实施方案、图2所示的处理系统200的一些实施方案以及图3的处理器300的一些实施方案中实现。在所示的实施方案中,处理器指令正在处理系统中的一个或多个处理器核(诸如图3所示的处理器300中的处理器核305)上执行或者被分配或调度以便在其上执行。如本文所论述,处理器指令包括分支指令。

[0073] 在框901处,BTB训练逻辑通过将目标地址与分支指令的预测地址进行比较来确定指令目标地址大小。基于目标地址大小,所述方法包括确定或选择要将预测目标地址存储在多个基于大小的BTB表中的哪一个中。例如,在框902处,处理器确定目标大小是否小于或等于第一大小“SIZE 1”。如果是,则在框903处,处理器将BTB条目存储在基于大小的BTB表(诸如图3的第一BTB表301)的基于第一大小的BTB条目中。如果不是,则在框904处,处理器确定目标地址大小是否小于或等于三尺寸BTB系统的第二大小“SIZE 2”。如果是,则在框905处,处理器将分支目标存储在基于第二大小的BTB表中。例如,目标地址存储在图3的第二BTB表302中。如果在框904处分支指令类型不小于或等于第二大小“SIZE 2”,则在框906处,处理器将分支目标存储在基于第三大小的BTB中。

[0074] 图10是根据一些实施方案的示出用于按分支类型将分支目标地址存储到多个BTB表中的一个中的方法的方法1000。方法1000在图1所示的指令流水线体系结构100的一些实施方案、图2所示的处理系统200、图4的处理器400以及图5的处理器500的一些实施方案中实现。在所示的实施方案中,处理器指令正在处理系统中的一个或多个处理器核(诸如图4所示的处理器400中的处理器核405)上执行或者被分配或调度以便在其上执行。如本文所论述,处理器指令包括分支指令。

[0075] 方法1000在三BTB表系统的示例性上下文中描述。在框1001处,处理器确定条件指令的BTB条目类型。基于分支指令类型,在框1002处,处理器确定分支指令类型对于BTB表系统来说是否是高值BTB条目。如果是,则在框1003处,处理器将BTB条目存储在第一BTB表(诸如图4的BTB表401)中。如果不是,则在框1004处,处理器确定分支指令类型是否是中值BTB条目。如果是,则在框1005处,处理器将BTB条目存储在第二BTB表(诸如图4的BTB表402)中。如果不是,则在框1006处,处理器将包括地址的BTB条目存储在第三BTB表(诸如图4的BTB表403)的条目中。

[0076] 关于高值、中值等,特定类型的条件分支指令可以是高值、中值或其他类型的BTB条目。例如,跳转类型可以是高值条目,并且返回类型指令可以是中值条目。根据一些实施方案,条目的值对应于相对于不保存条目或相对于可保存在BTB或BTB表中的其他可能条目节省多个处理器周期的条目。如果分支指令有价值,诸如在未来过程周期中 useful,则处理器将分支目标存储在BTB表中。例如,目标地址存储在图4的类型BTB表401-403中的一者中。如果分支指令类型不是有用的分支类型,则处理器不将目标地址存储在BTB表条目中。

[0077] 如本文所公开,在一些实施方案中,一种处理器包括:指令执行流水线;第一分支目标缓冲器(BTB)表,所述第一BTB表具有第一数量的第一BTB表条目,每个第一BTB表条目

包括第一标签和第一目标地址,每个第一目标地址具有第一宽度;第二BTB表,所述第二BTB表具有第二数量的第二BTB表条目,每个第二BTB表条目包括第二标签和第二目标地址,每个第二目标地址具有不同于第一宽度的第二宽度;分支预测器,所述分支预测器被配置为提供给定预测地址的预测目标地址;并且其中所述处理器被配置为基于分支指令的分支特性将包括所述第一目标地址的分支描述符存储在所述第一BTB表的第一BTB表条目或所述第二BTB表的第二BTB表条目中。在一个方面,所述分支特性是分支目标地址大小。在另一方面,所述分支特性是分支类型。在又一方面,所述第一BTB表条目被配置用于跳转分支指令、调用分支指令、返回分支指令和条件分支指令中的至少一者。

[0078] 在一个方面,所述分支特性是存储在所述BTB条目中的分支的数量。在另一方面,所述分支特性是所述BTB条目的线程标识符。在又一方面,所述处理器包括:第一存储器级别高速缓存和第二存储器级别高速缓存;并且所述第一BTB表和所述第二BTB表包括在所述处理器的同一存储器级别高速缓存中。在再一方面,所述第一BTB表的所述第一BTB表条目是N路关联的以用于预测查找;并且所述第二BTB表的所述第二BTB表项是M路关联的以用于预测查找,其中M和N大于或等于一并且M不同于N。

[0079] 在另一方面,所述第一BTB表条目的第一数量不同于所述第二BTB表条目的第二数量。在又一方面,所述BTB表条目中的每一个包括最近使用状态位,所述最近使用状态位在组内将所述相应BTB条目标记为组内最近最多使用(MRU)状态和最近最少使用(LRU)状态中的一者;并且所述处理器被配置为基于所述最近使用状态位的状态根据更换策略逐出组内的BTB条目。

[0080] 如本文所公开,在一些实施方案中,一种方法包括:基于由对分支指令的错误预测造成的重定向来确定分支目标缓冲器(BTB)条目的分支类型;以及基于所确定的BTB条目的分支类型,将所述BTB条目存储在以下中的一者中:处理器的第一分支目标缓冲器(BTB)表的第一条目,所述第一条目具有对应于BTB条目的第一分支类型的第一目标地址宽度;以及处理器的第二BTB表的第二条目,所述第二条目具有对应于不同于所述第一分支类型的第二分支类型的不同于所述第一目标地址宽度的第二目标地址宽度。在一方面,所述分支类型是分支目标地址大小。在另一方面,所述BTB条目的所述存储由处理器的预测训练器单元执行。

[0081] 在一个方面,所述方法包括:基于在组内将所述第一BTB表或所述第二BTB表的BTB条目标记为最近最多使用(MRU)状态和最近最少使用(LRU)状态中的一者的最近使用状态位来标识所述BTB条目;以及在将所述条目存储在所述第一BTB表或所述第二BTB表中之前,逐出所述第一BTB表或所述第二BTB表的所标识BTB条目。在另一方面:所述第一BTB表和所述第二BTB表包括在所述处理器的同一存储器级别高速缓存中。在又一方面,所述方法包括:在存储所述BTB条目之前,基于所述分支指令的指令标签搜索所述第一BTB表和所述第二BTB表中的至少一者;并且在存储所述BTB条目之前,标识出在所述第一BTB表和所述第二BTB表中的所述至少一者中未找到BTB条目。在再一方面,所述方法包括:当所述BTB条目的目标地址的大小超过相应的第一BTB表或所述第二BTB表的BTB条目的地址大小时,将所述BTB条目的所述目标地址的溢出位存储在溢出BTB表中。

[0082] 如本文所公开,在一些实施方案中,一种方法包括:通过在处理器的BTB的多个分支目标缓冲器(BTB)表中在当前预测地址处查找来预测当前预测块的结束地址和下一预测

块的起始地址,所述BTB包括第一BTB表和第二BTB表;以及基于所预测结束地址和所预测起始地址中的至少一者提供预测作为用于在所述处理器处执行的指令的一部分。在一个方面,所述第一BTB表的BTB条目具有第一宽度的目标地址字段,所述第一宽度小于所述第二BTB表的BTB条目的目标地址字段的第二宽度。在另一方面,所述当前预测地址是相对于程序计数器的。

[0083] 在一些实施方案中,上文所描述的设备和技术被实现在包括一个或多个集成电路(IC)装置(也称为集成电路封装或微芯片)的系统(诸如上文参考图1至图7所述的系统、处理器和BTB表)中。电子设计自动化(EDA)和计算机辅助设计(CAD)软件工具可用于设计和制作这些IC装置。这些设计工具通常被表示为一个或多个软件程序。所述一个或多个软件程序包括代码,所述代码可由计算机系统执行以操纵计算机系统来对表示一个或多个IC装置的电路的代码进行操作,以便执行用以设计或调适制造系统从而制造电路的过程的至少一部分。这个代码可包括指令、数据或指令和数据的组合。表示设计工具或制作工具的软件指令通常存储在可由计算系统访问的计算机可读存储介质中。同样,代表IC装置的设计或制作的一个或多个阶段的代码可以存储在相同的计算机可读存储介质或不同的计算机可读存储介质中并且从其进行访问。

[0084] 计算机可读存储介质可包括任何非暂时性存储介质或者非暂时性存储介质组合,其在使用期间可由计算机系统访问以向计算机系统提供指令和/或数据。此类存储介质可包括但不限于光学介质(例如,压缩光盘(CD)、数字通用光盘(DVD)、蓝光光盘)、磁性介质(例如,软盘、磁带或磁性硬盘驱动器)、易失性存储器(例如,随机存取存储器(RAM)或高速缓存)、非易失性存储器(例如,只读存储器(ROM)或快闪存储器)或基于微机电系统(MEMS)的存储介质。计算机可读存储介质可嵌入在计算系统中(例如,系统RAM或ROM)、固定地附接到计算系统(例如,磁性硬盘驱动器)、可拆卸地附接到计算系统(例如,光盘或基于通用串行总线(USB)的快闪存储器),或者经由有线或无线网络耦接到计算机系统(例如,网络可访问存储装置(NAS))。

[0085] 在一些实施方案中,上述技术的某些方面可由执行软件的处理系统的一个或多个处理器来实现。软件包括存储或以其他方式有形地体现在非暂时性计算机可读存储介质上的一个或多个可执行指令集。所述软件可包括当由一个或多个处理器执行时操纵一个或多个处理器以执行上述技术的一个或多个方面的指令和某些数据。非暂时性计算机可读存储介质可包括例如磁盘或光盘存储装置、固态存储装置诸如快闪存储器、缓存、随机存取存储器(RAM)或其他一个或多个非易失性存储器装置等。存储在非暂时性计算机可读存储介质上的可执行指令可呈源代码、汇编语言代码、目标代码或者由一个或多个处理器解译或以其他方式执行的其他指令格式。

[0086] 并不需要上文在一般描述中所描述的所有活动或元件,特定活动或装置的一部分可能是不需要的,并且可执行一个或多个其他活动,或者可包括除所描述的那些元件之外的元件。此外,列出活动的顺序不一定是执行活动的顺序。另外,已参考具体实施方案描述各概念。然而,本领域普通技术人员应了解,在不脱离如所附权利要求中所阐述的本公开范围的情况下,可作出各种修改和改变。因此,本说明书和附图将被视为说明性的而非限制性的,并且所有此类修改都意图被包括在本公开的范围之内。

[0087] 上文已参考具体实施方案描述益处、其他优点以及问题的解决方案。然而,所述益

处、优点、问题解决方案以及可使任何益处、优点或问题解决方案出现或变得更突出的任何特征都不应被解释为是任何或所有权利要求的关键、必需或必要特征。此外，上文所公开特定实施方案仅为说明性的，因为所公开主题可按受益于本文教义的本领域技术人员显而易见的不同但等效的方式来修改和实践。除了如所附权利要求中所描述的之外，并不意图限制本文所示出的构造或设计的细节。因此，上文所公开的特定实施方案可更改或修改，并且所有此类变化形式被认为在所公开主题的范围内。因此，本文所寻求的保护正如所附权利要求中所陈述。

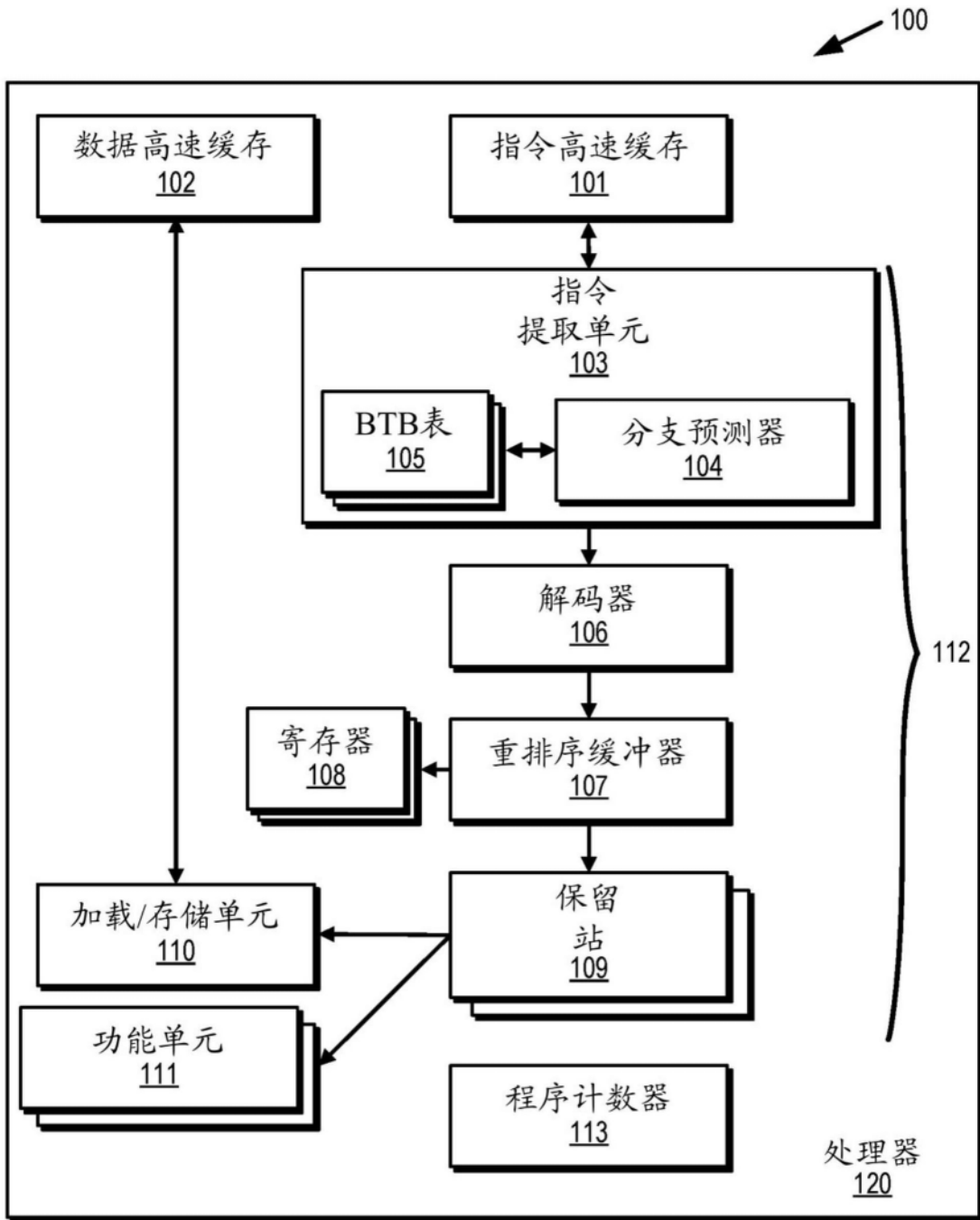


图1

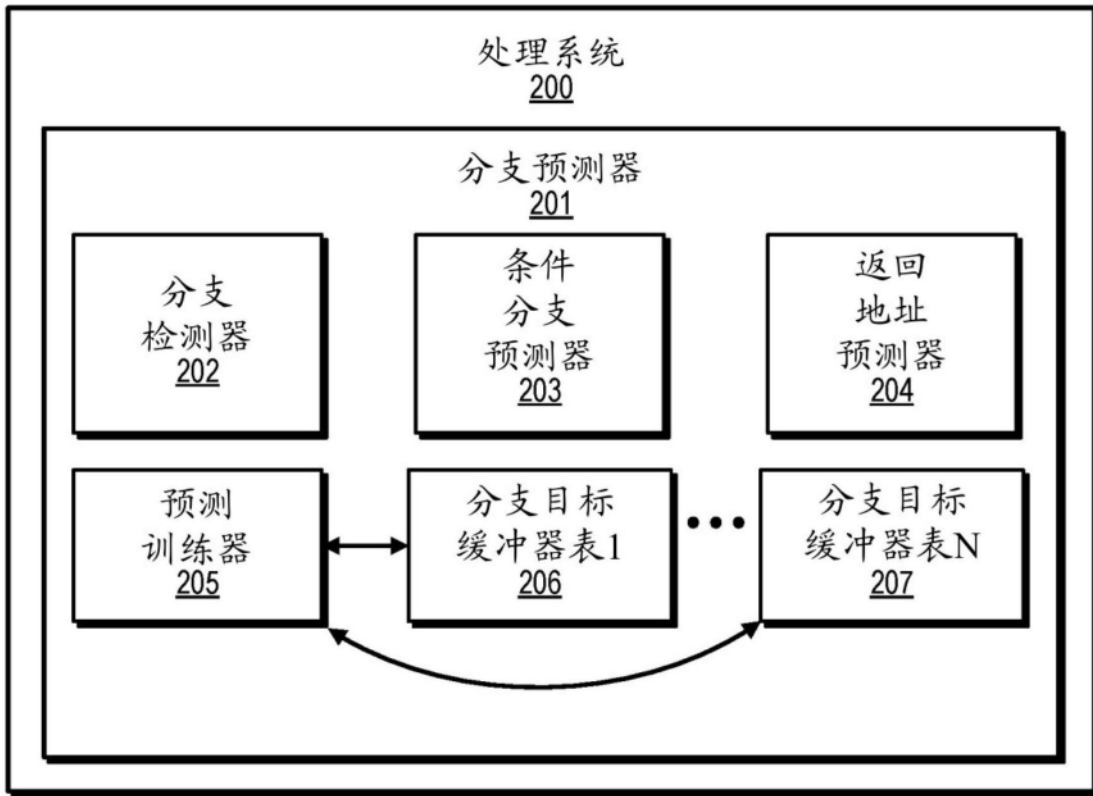


图2

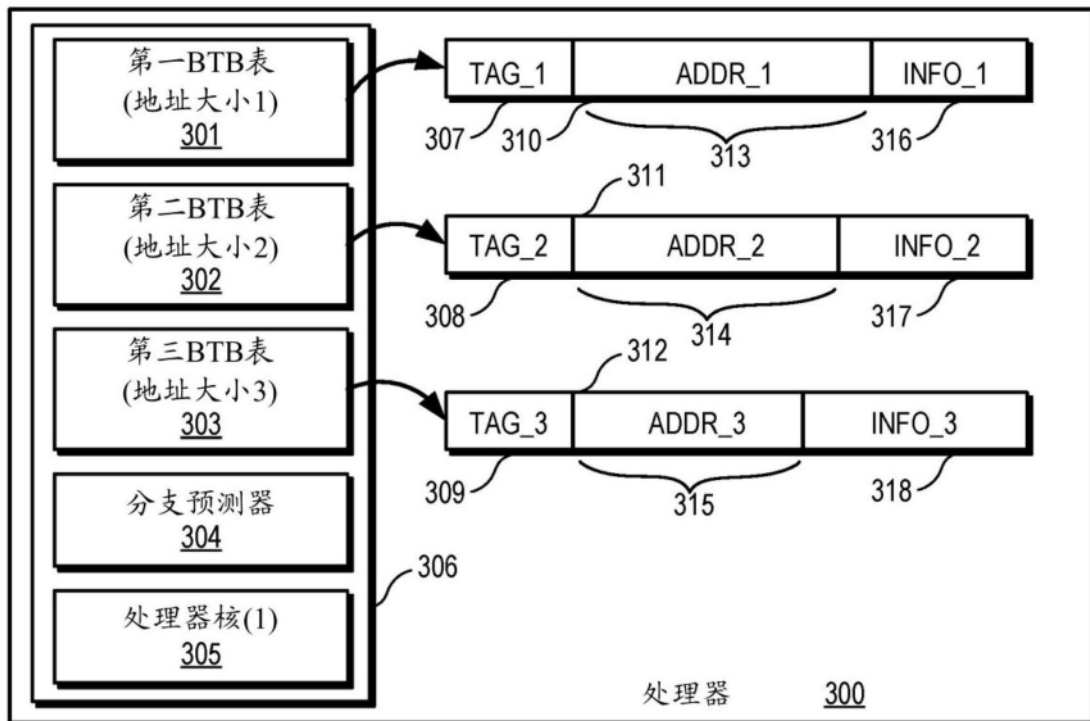


图3

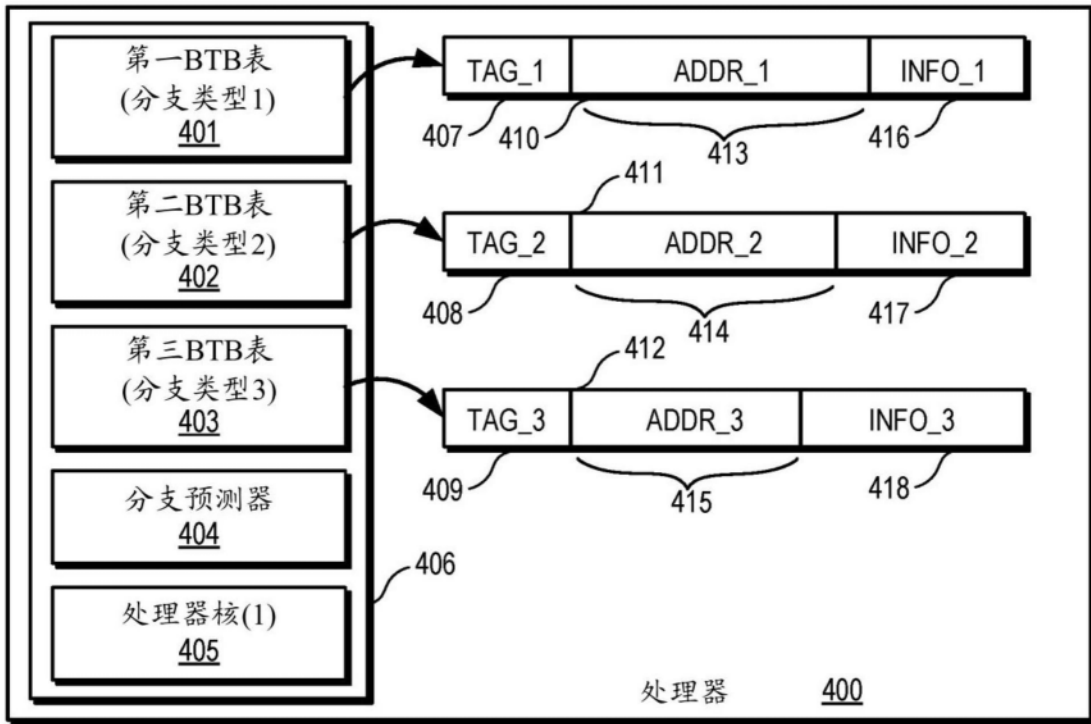


图4

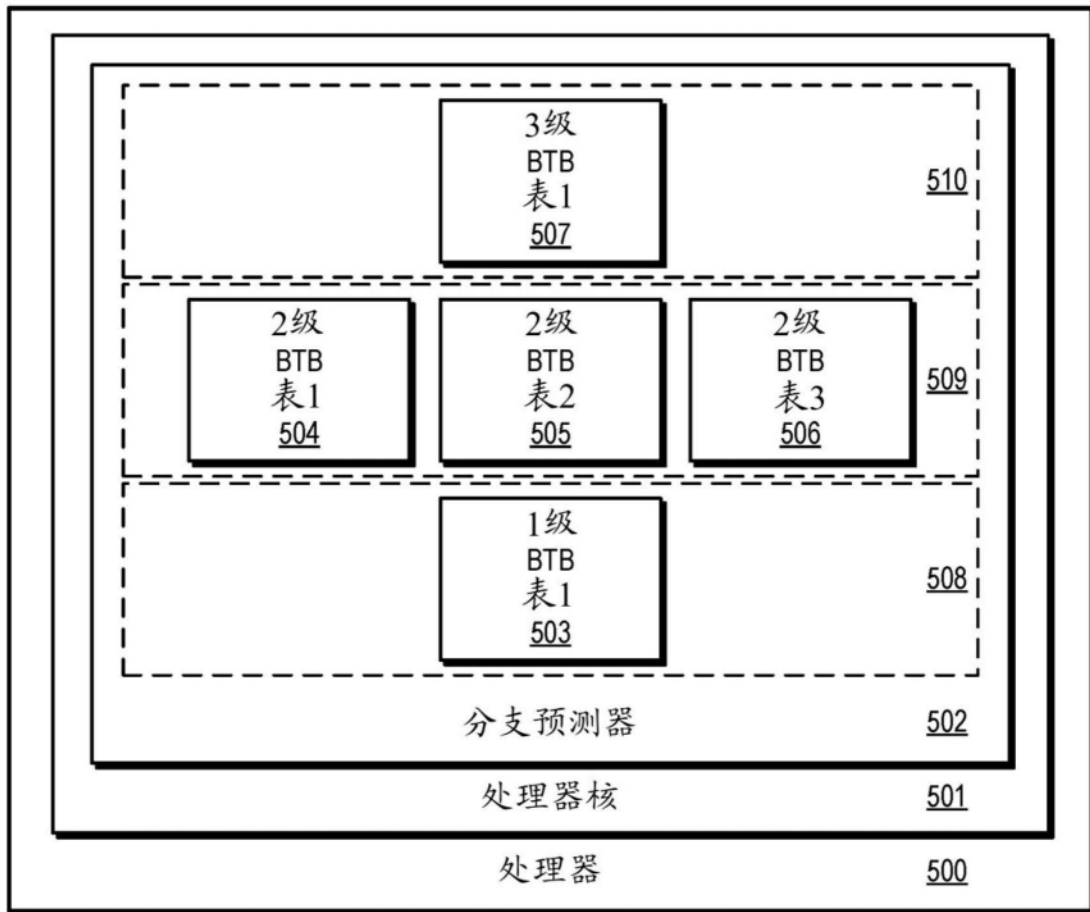


图5

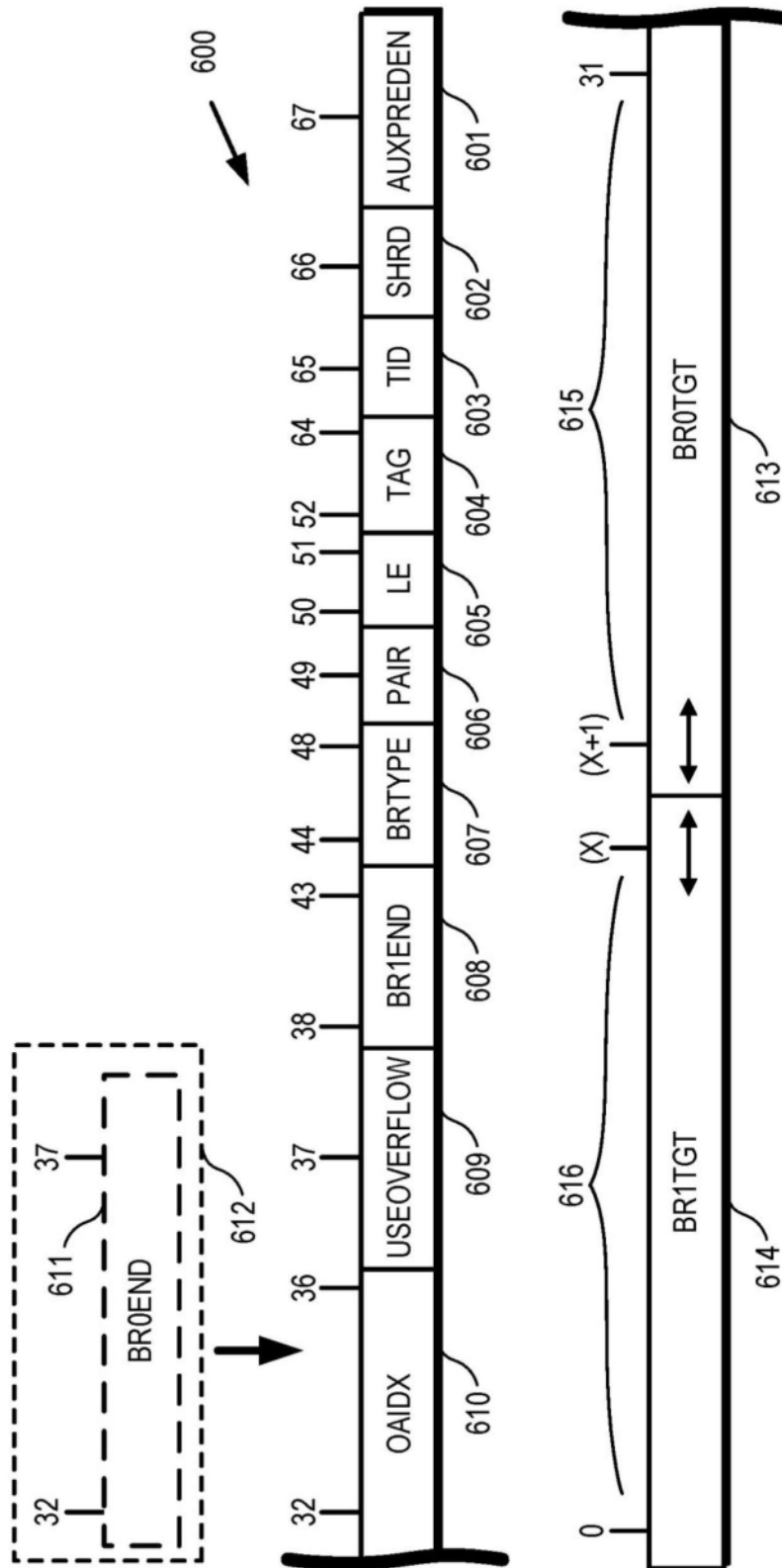


图6

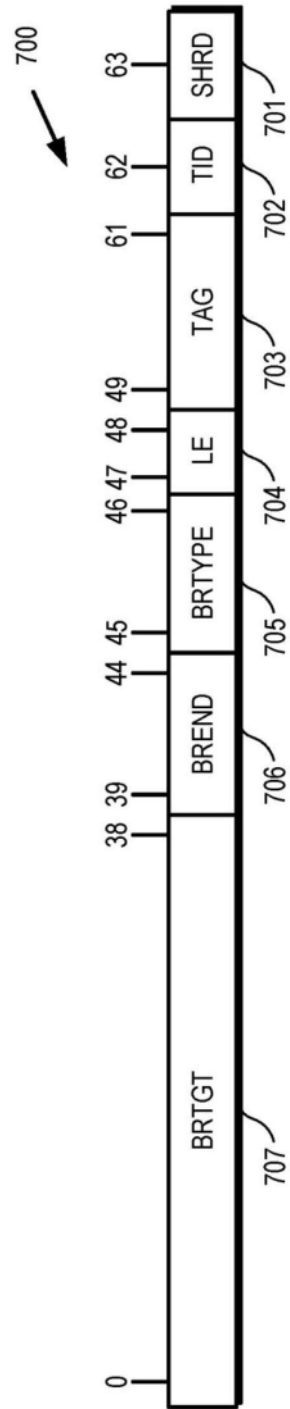


图7

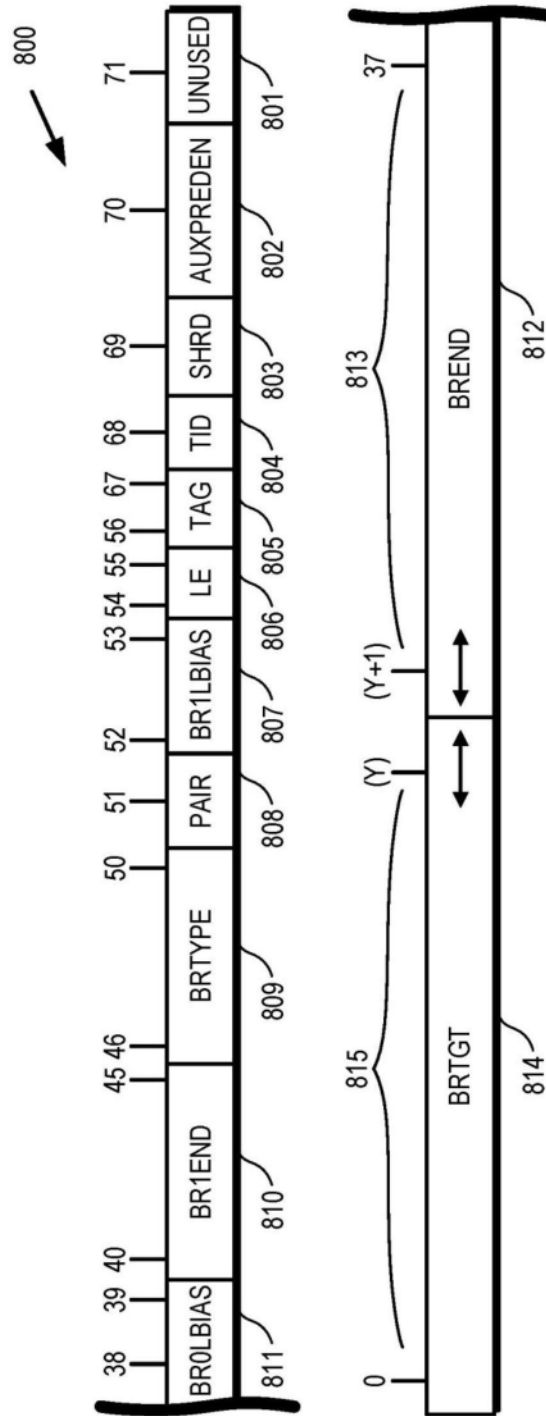


图8

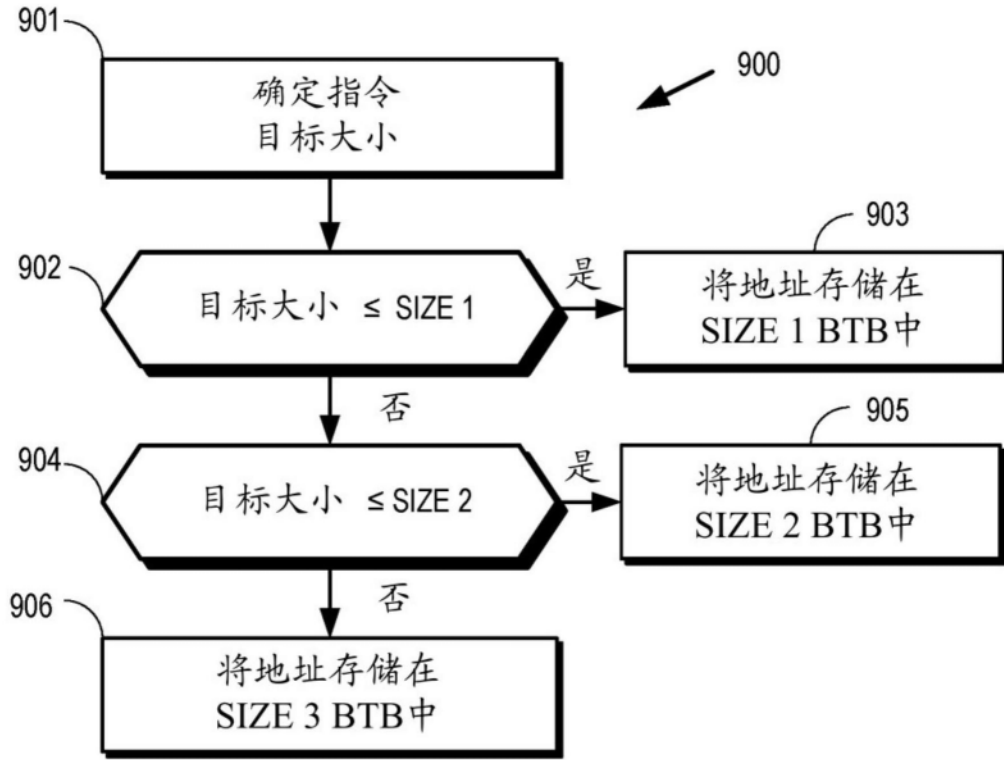


图9

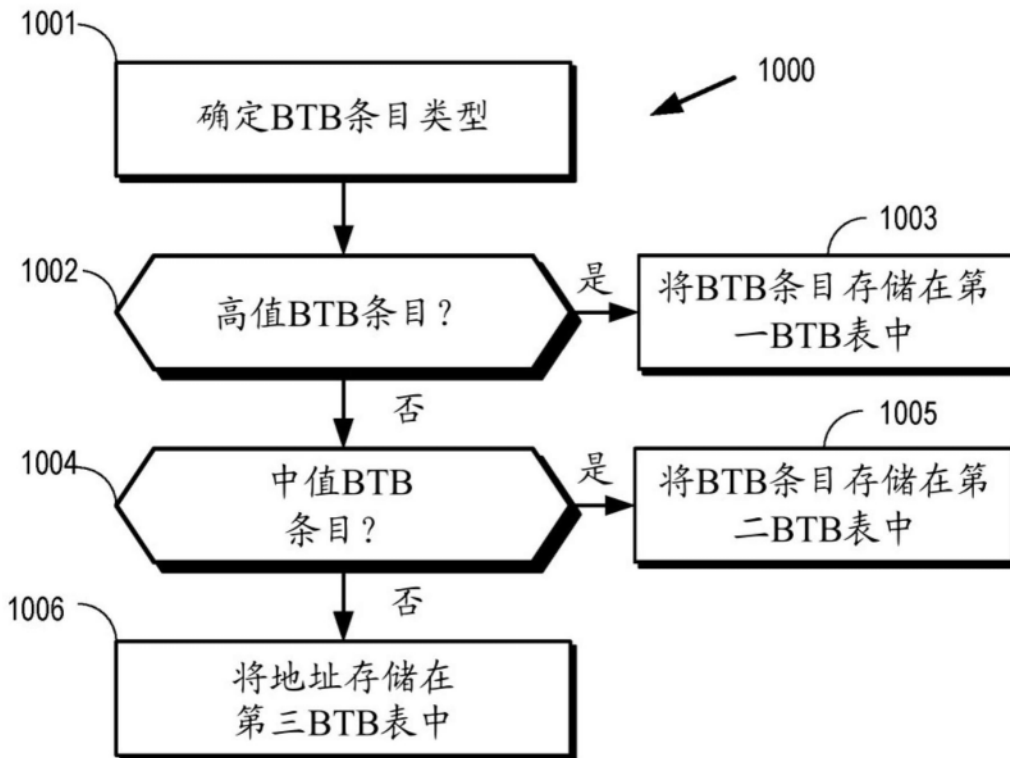


图10