



(12) 发明专利

(10) 授权公告号 CN 109960607 B

(45) 授权公告日 2021.04.20

(21) 申请号 201711405263.0  
 (22) 申请日 2017.12.22  
 (65) 同一申请的已公布的文献号  
 申请公布号 CN 109960607 A  
 (43) 申请公布日 2019.07.02  
 (73) 专利权人 龙芯中科技术股份有限公司  
 地址 100095 北京市海淀区中关村环保科  
 技示范园龙芯产业园2号楼  
 (72) 发明人 李策 吴瑞阳  
 (74) 专利代理机构 北京同立钧成知识产权代理  
 有限公司 11205  
 代理人 杨泽 刘芳  
 (51) Int.Cl.  
 G06F 11/14 (2006.01)

(56) 对比文件  
 US 2010161951 A1,2010.06.24  
 US 5964868 A,1999.10.12  
 US 6253315 B1,2001.06.26  
 CN 1560734 A,2005.01.05  
 CN 101535951 A,2009.09.16  
 CN 104252586 A,2014.12.31  
 CN 104820580 A,2015.08.05  
 吴瑞阳 等.龙芯GS464E处理器核架构设计.  
 《中国科学:信息科学》.2015,第45卷(第4期),  
 480-500.  
 Hans Vandierendonck 等.Speculative  
 return address stack management  
 revisited.《ACM Transactions on  
 Architecture and Code Optimization  
 (TACO)》.2008,第5卷(第3期),1-15.

审查员 宋秀之

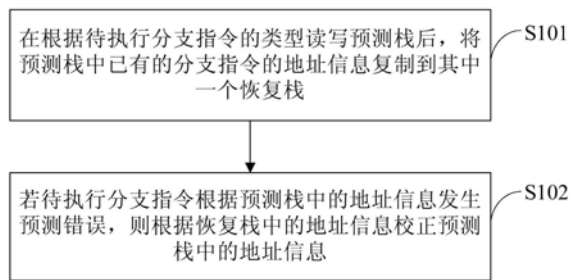
权利要求书2页 说明书10页 附图8页

(54) 发明名称

预测栈的错误恢复方法、装置及存储介质

(57) 摘要

本发明提供一种预测栈的错误恢复方法、装置及存储介质,该方法包括:在根据待执行分支指令的类型读写预测栈后,将预测栈中已有的分支指令的地址信息复制到其中一个恢复栈;若待执行分支指令根据预测栈中的地址信息发生预测错误,则根据恢复栈中的地址信息校正预测栈中的地址信息。本发明提供的预测栈的错误恢复方法、装置及存储介质,提高了返回地址预测栈的准确率。



1. 一种预测栈的错误恢复方法,其特征在于,配置有对预测栈中的地址信息进行备份的至少一个恢复栈,所述至少一个恢复栈中的每个恢复栈有相应的恢复栈编号;该方法包括:

在根据待执行分支指令的类型读写预测栈后,将所述预测栈中已有的分支指令的地址信息复制到其中一个恢复栈;

若所述待执行分支指令根据所述预测栈中的地址信息发生预测错误,则根据所述恢复栈中的地址信息校正所述预测栈中的地址信息;

获取预测错误信息;所述预测错误信息包括所述待执行分支指令对应的所述预测栈的栈顶指针和所述对应的恢复栈的恢复栈编号;

根据所述恢复栈编号对应的所述恢复栈中的地址信息校正所述栈顶指针对应的预测栈中的地址信息。

2. 根据权利要求1所述的方法,其特征在于,

所述在根据待执行分支指令的类型读写预测栈后,将所述预测栈中已有的分支指令的地址信息复制到所述恢复栈,包括:

在所述待执行分支指令的类型为函数调用类分支指令时,在所述预测栈的栈顶指针中写入所述待执行分支指令的地址信息;将所述预测栈中已有的分支指令的地址信息复制到对应的恢复栈,所述待执行分支指令携带所述预测栈的栈顶指针和所述对应的恢复栈的恢复栈编号;

在所述待执行分支指令的类型为函数返回类分支指令时,将所述预测栈的栈顶指针的地址信息确定为所述待执行分支指令的预测地址信息,并将所述栈顶指针中的地址信息从所述预测栈中读出;将所述预测栈中已有的分支指令的地址信息复制到对应的恢复栈,所述待执行分支指令携带所述预测栈的栈顶指针和所述对应的恢复栈的恢复栈编号;

在所述待执行分支指令的类型为其他类分支指令时,确定所述待执行分支指令的前一条分支指令对应的恢复栈;将所述前一条分支指令对应的恢复栈确定为所述待执行分支指令对应的恢复栈,所述待执行分支指令携带所述预测栈的栈顶指针和所述对应的恢复栈的恢复栈编号。

3. 根据权利要求1所述的方法,其特征在于,还包括:

在新的待执行分支指令读写预测栈时,根据预设规则将所述恢复栈中的地址信息清空,得到新的恢复栈;

将所述预测栈中已有的分支指令的地址信息复制到所述新的恢复栈。

4. 根据权利要求3所述的方法,其特征在于,所述在新的待执行分支指令读写预测栈时,根据预设规则将所述恢复栈中的地址信息清空,得到新的恢复栈,包括:

当待执行分支指令的数量大于预设阈值时,在新的待执行分支指令读写预测栈时,根据所述恢复栈的编号将所述恢复栈中的地址信息清空,得到所述新的恢复栈。

5. 一种预测栈的错误恢复装置,其特征在于,配置有对预测栈中的地址信息进行备份的至少一个恢复栈,所述至少一个恢复栈中的每个恢复栈有相应的恢复栈编号;该装置包括:

复制单元,用于在根据待执行分支指令的类型读写预测栈后,将所述预测栈中已有的分支指令的地址信息复制到其中一个恢复栈;

校正单元,用于若所述待执行分支指令根据所述预测栈中的地址信息发生预测错误,则根据所述恢复栈中的地址信息校正所述预测栈中的地址信息;

所述校正单元,具体用于获取预测错误信息;所述预测错误信息包括所述待执行分支指令对应的所述预测栈的栈顶指针和所述对应的恢复栈的恢复栈编号;并根据所述恢复栈编号对应的所述恢复栈中的地址信息校正所述栈顶指针对应的预测栈中的地址信息。

6. 根据权利要求5所述的装置,其特征在于,

所述复制单元,具体用于在所述待执行分支指令的类型为函数调用类分支指令时,在所述预测栈的栈顶指针中写入所述待执行分支指令的地址信息;将所述预测栈中已有的分支指令的地址信息复制到对应的恢复栈,所述待执行分支指令携带所述预测栈的栈顶指针和所述对应的恢复栈的恢复栈编号;

所述复制单元,具体用于在所述待执行分支指令的类型为函数返回类分支指令时,将所述预测栈的栈顶指针的地址信息确定为所述待执行分支指令的预测地址信息,并将所述栈顶指针中的地址信息从所述预测栈中读出;将所述预测栈中已有的分支指令的地址信息复制到对应的恢复栈,所述待执行分支指令携带所述预测栈的栈顶指针和所述对应的恢复栈的恢复栈编号;

所述复制单元,具体用于在所述待执行分支指令的类型为其他类分支指令时,确定所述待执行分支指令的前一条分支指令对应的恢复栈;将所述前一条分支指令对应的恢复栈确定为所述待执行分支指令对应的恢复栈,所述待执行分支指令携带所述预测栈的栈顶指针和所述对应的恢复栈的恢复栈编号。

7. 根据权利要求5所述的装置,其特征在于,还包括:

清空单元,用于在新的待执行分支指令读写预测栈时,根据预设规则将所述恢复栈中的地址信息清空,得到新的恢复栈;

所述复制单元,还用于将所述预测栈中已有的分支指令的地址信息复制到所述新的恢复栈。

8. 一种计算机可读存储介质,其特征在于,

计算机可读存储介质上存储有计算机程序,在所述计算机程序被处理器执行时,执行权利要求1~4任一项所述的预测栈的错误恢复方法。

## 预测栈的错误恢复方法、装置及存储介质

### 技术领域

[0001] 本发明涉及计算机技术领域,尤其涉及一种预测栈的错误恢复方法、装置及存储介质。

### 背景技术

[0002] 返回地址预测栈是处理器中分支预测器的一种,主要用于预测函数返回类分支指令,由于函数返回类分支指令与函数调用类分支指令一一对应,因此,当遇到函数调用类分支指令时,将其下一条指令的地址压入预测栈;当遇到函数返回类分支指令时,将该地址弹出栈用作预测地址,返回地址预测栈相对于条件分支预测器与间接跳转预测器,拥有很小的硬件开销,而且能够达到较高的预测准确率。

[0003] 但是,现代处理器中存在猜测执行,即执行的分支指令的跳转地址是不确定的,这样一旦猜测错误,返回地址预测栈就很有可能发生错误,导致后续的预测出现错误,降低了返回地址预测栈的预测准确率。为了提高返回地址预测栈的预测准确率,现有技术中通过维护一个恢复栈,将猜测执行过程中被弹出的栈内项放入恢复栈内,当猜测执行错误时,将恢复栈内的项全部复写到返回地址预测栈中。然而,由于存在乱序执行机制,分支错误返回时不定时顺序返回,由此造成的错误覆盖同样会污染栈内的相关项,从而导致返回地址预测栈的准确率不高。

### 发明内容

[0004] 本发明提供一种预测栈的错误恢复方法、装置及存储介质,以解决现有技术中返回地址预测栈准确率不高的问题。

[0005] 本发明实施例提供一种预测栈的错误恢复方法,配置有对预测栈中的地址信息进行备份的至少一个恢复栈;该方法包括:

[0006] 在根据待执行分支指令的类型读写预测栈后,将所述预测栈中已有的分支指令的地址信息复制到其中一个恢复栈;

[0007] 若所述待执行分支指令根据所述预测栈中的地址信息发生预测错误,则根据所述恢复栈中的地址信息校正所述预测栈中的地址信息。

[0008] 在本发明一实施例中,所述至少一个恢复栈中的每个恢复栈有相应的恢复栈编号;

[0009] 所述在根据待执行分支指令的类型读写预测栈后,将所述预测栈中已有的分支指令的地址信息复制到所述恢复栈,包括:

[0010] 在所述待执行分支指令的类型为函数调用类分支指令时,在所述预测栈的栈顶指针中写入所述待执行分支指令的地址信息;将所述预测栈中已有的分支指令的地址信息复制到对应的恢复栈,所述待执行分支指令携带所述预测栈的栈顶指针和所述对应的恢复栈的恢复栈编号;

[0011] 在所述待执行分支指令的类型为函数返回类分支指令时,将所述预测栈的栈顶指

针的地址信息确定为所述待执行分支指令的预测地址信息,并将所述栈顶指针中的地址信息从所述预测栈中读出;将所述预测栈中已有的分支指令的地址信息复制到对应的恢复栈,所述待执行分支指令携带所述预测栈的栈顶指针和所述对应的恢复栈的恢复栈编号;

[0012] 在所述待执行分支指令的类型为其他类分支指令时,确定所述待执行分支指令的前一条分支指令对应的恢复栈;将所述前一条分支指令对应的恢复栈确定为所述待执行分支指令对应的恢复栈,所述待执行分支指令携带所述预测栈的栈顶指针和所述对应的恢复栈的恢复栈编号。

[0013] 在本发明一实施例中,所述若所述待执行分支指令根据所述预测栈中的地址信息发生预测错误,则根据所述恢复栈中的地址信息校正所述预测栈中的地址信息,包括:

[0014] 获取预测错误信息;所述预测错误信息包括所述待执行分支指令对应的所述预测栈的栈顶指针和所述待对应的恢复栈的恢复栈编号;

[0015] 根据所述恢复栈编号对应的所述恢复栈中的地址信息校正所述栈顶指针对应的预测栈中的地址信息。

[0016] 在本发明一实施例中,该预测栈的错误恢复方法还包括:

[0017] 在新的待执行分支指令读写预测栈时,根据预设规则将所述恢复栈中的地址信息清空,得到新的恢复栈;

[0018] 将所述预测栈中已有的分支指令的地址信息复制到所述新的恢复栈。

[0019] 在本发明一实施例中,所述在新的待执行分支指令读写预测栈时,根据预设规则将所述恢复栈中的地址信息清空,得到新的恢复栈,包括:

[0020] 当待执行分支指令的数量大于预设阈值时,在新的待执行分支指令读写预测栈时,根据所述恢复栈的编号将所述恢复栈中的地址信息清空,得到所述新的恢复栈。

[0021] 本发明实施例还提供一种预测栈的错误恢复装置,配置有对预测栈中的地址信息进行备份的至少一个恢复栈;该装置包括:

[0022] 复制单元,用于在根据待执行分支指令的类型读写预测栈后,将所述预测栈中已有的分支指令的地址信息复制到其中一个恢复栈;

[0023] 校正单元,用于若所述待执行分支指令根据所述预测栈中的地址信息发生预测错误,则根据所述恢复栈中的地址信息校正所述预测栈中的地址信息。

[0024] 在本发明一实施例中,所述至少一个恢复栈中的每个恢复栈有相应的恢复栈编号;

[0025] 所述复制单元,具体用于在所述待执行分支指令的类型为函数调用类分支指令时,在所述预测栈的栈顶指针中写入所述待执行分支指令的地址信息;将所述预测栈中已有的分支指令的地址信息复制到对应的恢复栈,所述待执行分支指令携带所述预测栈的栈顶指针和所述对应的恢复栈的恢复栈编号;

[0026] 所述复制单元,具体用于在所述待执行分支指令的类型为函数返回类分支指令时,将所述预测栈的栈顶指针的地址信息确定为所述待执行分支指令的预测地址信息,并将所述栈顶指针中的地址信息从所述预测栈中读出;将所述预测栈中已有的分支指令的地址信息复制到对应的恢复栈,所述待执行分支指令携带所述预测栈的栈顶指针和所述对应的恢复栈的恢复栈编号;

[0027] 所述复制单元,具体用于在所述待执行分支指令的类型为其他类分支指令时,确

定所述待执行分支指令的前一条分支指令对应的恢复栈;将所述前一条分支指令对应的恢复栈确定为所述待执行分支指令对应的恢复栈,所述待执行分支指令携带所述预测栈的栈顶指针和所述对应的恢复栈的恢复栈编号。

[0028] 在本发明一实施例中,所述校正单元,具体用于获取预测错误信息;所述预测错误信息包括所述待执行分支指令对应的所述预测栈的栈顶指针和所述待对应的恢复栈的恢复栈编号;并根据所述恢复栈编号对应的所述恢复栈中的地址信息校正所述栈顶指针对应的预测栈中的地址信息。

[0029] 在本发明一实施例中,该预测栈的错误恢复装置还包括:

[0030] 清空单元,用于在新的待执行分支指令读写预测栈时,根据预设规则将所述恢复栈中的地址信息清空,得到新的恢复栈;

[0031] 所述复制单元,还用于将所述预测栈中已有的分支指令的地址信息复制到所述新的恢复栈。

[0032] 在本发明一实施例中,所述清空单元,具体用于当待执行分支指令的数量大于预设阈值时,在新的待执行分支指令读写预测栈时,根据所述恢复栈的编号将所述恢复栈中的地址信息清空,得到所述新的恢复栈。

[0033] 本发明实施例还提供一种计算机可读存储介质,计算机可读存储介质上存储有计算机程序,在所述计算机程序被处理器执行时,执行上述任一实施例所述的预测栈的错误恢复方法。

[0034] 本发明实施例提供的预测栈的错误恢复方法、装置及存储介质,配置有对预测栈中的地址信息进行备份的至少一个恢复栈;在根据待执行分支指令的类型读写预测栈后,将预测栈中已有的分支指令的地址信息复制到其中一个恢复栈;若待执行分支指令根据预测栈中的地址信息发生预测错误,则根据恢复栈中的地址信息校正预测栈中的地址信息。这样,通过为预测栈构建相应的恢复栈,使得在待执行分支指令在预测错误时,可以根据恢复栈编号,将恢复栈编号对应的恢复栈中的地址信息恢复至栈顶指针对应的预测栈中,从而提高了返回地址预测栈的准确率。

## 附图说明

[0035] 此处的附图被并入说明书中并构成本说明书的一部分,示出了符合本公开的实施例,并与说明书一起用于解释本公开的原理。

[0036] 图1为本发明实施例提供的一种预测栈的错误恢复方法的示意图;

[0037] 图2为本发明实施例提供的一种恢复栈的示意图;

[0038] 图3为本发明实施例提供的一种新的恢复栈的获取方式的示意图;

[0039] 图4为本发明实施例提供的一种A函数调用类分支指令读写预测栈的示意图;

[0040] 图5为本发明实施例提供的一种B函数调用类分支指令读写预测栈的示意图;

[0041] 图6为本发明实施例提供的一种C函数返回类分支指令读写预测栈的示意图;

[0042] 图7为本发明实施例提供的一种恢复预测栈的示意图;

[0043] 图8为本发明实施例提供的另一种恢复预测栈的示意图;

[0044] 图9为本发明实施例提供的一种E函数调用类分支指令读写预测栈的示意图;

[0045] 图10为本发明实施例提供的再一种恢复预测栈的示意图;

[0046] 图11为本发明实施例提供的一种预测栈的错误恢复装置的结构示意图；

[0047] 图12为本发明实施例提供的另一种预测栈的错误恢复装置的结构示意图；

[0048] 图13为本发明提供的一种电子设备的结构示意图。

[0049] 通过上述附图,已示出本公开明确的实施例,后文中将有更详细的描述。这些附图和文字描述并不是为了通过任何方式限制本公开构思的范围,而是通过参考特定实施例为本领域技术人员说明本公开的概念。

### 具体实施方式

[0050] 这里将详细地对示例性实施例进行说明,其示例表示在附图中。下面的描述涉及附图时,除非另有表示,不同附图中的相同数字表示相同或相似的要素。以下示例性实施例中所描述的实施方式并不代表与本公开相一致的所有实施方式。相反,它们仅是与如所附权利要求书中所详述的、本公开的一些方面相一致的装置和方法的例子。

[0051] 本发明的说明书和权利要求书及上述附图中的术语“第一”、“第二”、“第三”“第四”等(如果存在)是用于区别类似的对象,而不必用于描述特定的顺序或先后次序。应该理解这样使用的数据在适当情况下可以互换,以便这里描述的本发明的实施例例如能够以除了在这里图示或描述的那些以外的顺序实施。此外,术语“包括”和“具有”以及他们的任何变形,意图在于覆盖不排他的包含,例如,包含了一系列步骤或单元的过程、方法、系统、产品或设备不必限于清楚地列出的那些步骤或单元,而是可包括没有清楚地列出的或对于这些过程、方法、产品或设备固有的其它步骤或单元。

[0052] 为了提高返回地址预测栈的预测准确率,现有技术中通过维护一个恢复栈,将猜测执行过程中被弹出的栈内项放入恢复栈内,当猜测执行错误时,将恢复栈内的项全部复写到返回地址预测栈中。然而,由于存在乱序执行机制,分支错误返回时不定时顺序返回,由此造成的错误覆盖同样会污染栈内的相关项,从而导致返回地址预测栈的准确率不高。为了提高返回地址预测栈的预测准确率,本发明实施例提供了一种预测栈的错误恢复方法,配置有对预测栈中的地址信息进行备份的至少一个恢复栈;在根据待执行分支指令的类型读写预测栈后,将预测栈中已有的分支指令的地址信息复制到其中一个恢复栈;若待执行分支指令根据预测栈中的地址信息发生预测错误,则根据恢复栈中的地址信息校正预测栈中的地址信息。这样,通过为预测栈构建相应的恢复栈,使得在待执行分支指令在预测错误时,可以根据恢复栈编号,将恢复栈编号对应的恢复栈中的地址信息恢复至栈顶指针对应的预测栈中,从而提高了返回地址预测栈的准确率。

[0053] 下面以具体的实施例对本发明的技术方案以及本申请的技术方案如何解决上述技术问题进行详细说明。下面这几个具体的实施例可以相互结合,对于相同或相似的概念或过程在某些实施例中不再赘述。下面将结合附图,对本发明的实施例进行描述。

[0054] 图1为本发明实施例提供的一种预测栈的错误恢复方法的示意图,该预测栈的错误恢复方法可以由预测栈的错误恢复装置执行,该预测栈的错误恢复装置可以独立设置,也可以集成在处理器中。示例的,请参见图1所示,该预测栈的错误恢复方法可以包括:

[0055] S101、在根据待执行分支指令的类型读写预测栈后,将预测栈中已有的分支指令的地址信息复制到其中一个恢复栈。

[0056] 其中,待执行分支指令的类型可以根据待执行分支指令的操作码的前五位进行确

定,待执行分支指令的类型包括函数调用类分支指令、函数返回类分支指令和其他类分支指令。对于函数调用类分支指令,在其读写预测栈时,是执行入栈操作,对于函数返回类分支指令,在其读写预测栈时,执行出栈操作,对于其他类分支指令,在其读写预测栈时,既不执行入栈操作,也不执行出栈操作。

[0057] 在将预测栈中已有的分支指令的地址信息复制到其中一个恢复栈之前,可以预先配置对预测栈中的地址信息进行备份的至少一个恢复栈,且该恢复栈为空的恢复栈,通过预先配置至少一个恢复栈,使得在待执行分支指令的类型读写预测栈后,可以将预测栈中已有的分支指令的地址信息复制到预先配置的其中一个恢复栈中。示例的,在本发明实施例中,至少一个恢复栈的个数可以为八个,请参见图2所示,图2为本发明实施例提供的一种恢复栈的示意图,整体的恢复栈共32项,该恢复栈分成独立的八组,每组四项,恢复栈编号循环分配,设恢复栈编号为RAS\_REC,初始值为1,最大值为8。

[0058] 在预先配置好至少一个恢复栈之后,就可以执行S101在根据待执行分支指令的类型读写预测栈后,将预测栈中已有的分支指令的地址信息复制到恢复栈,具体可以包括以下三种可能的实现方式:

[0059] 在第一种可能的实现方式中:当待执行分支指令的类型为函数调用类分支指令时,在预测栈的栈顶指针中写入待执行分支指令的地址信息,则此时预测栈中的地址信息包括当前待执行分支指令的地址信息,将预测栈中已有的分支指令的地址信息复制到对应的恢复栈,待执行分支指令携带预测栈的栈顶指针和对应的恢复栈的恢复栈编号。

[0060] 在第二种可能的实现方中:当待执行分支指令的类型为函数返回类分支指令时,将预测栈的栈顶指针的地址信息确定为待执行分支指令的预测地址信息,并将栈顶指针中的地址信息从预测栈中读出,则此时预测栈中的地址信息不包括当前待执行分支指令的地址信息,将预测栈中已有的分支指令的地址信息复制到对应的恢复栈,待执行分支指令携带预测栈的栈顶指针和对应的恢复栈的恢复栈编号。

[0061] 在第三种可能的实现方中:在待执行分支指令的类型为其他类分支指令时,确定待执行分支指令的前一条分支指令对应的恢复栈;将前一条分支指令对应的恢复栈确定为待执行分支指令对应的恢复栈,待执行分支指令携带预测栈的栈顶指针和对应的恢复栈的恢复栈编号。

[0062] 需要说明的是,对于其他类分支指令,在该其他类分支指令读写预测栈时,该其他类分支指令既不执行地址信息写入操作,也不执行地址信息读出操作,因此,在确定该其他类分支指令对应的恢复栈时,需要先确定该其他类分支指令的前一条分支指令对应的恢复栈;将前一条分支指令对应的恢复栈确定为其他类分支指令对应的恢复栈,且其他类分支指令携带预测栈的栈顶指针和对应的恢复栈的恢复栈编号。

[0063] S102、若待执行分支指令根据预测栈中的地址信息发生预测错误,则根据恢复栈中的地址信息校正预测栈中的地址信息。

[0064] 通过上述S101,在得到三种不同类型分支指令对应的恢复栈之后,就可以执行S102,在待执行分支指令根据预测栈中的地址信息发生预测错误时,可以根据恢复栈中的地址信息校正预测栈中的地址信息,从而提高预测栈中地址信息的准确率。

[0065] 可选的,S102若待执行分支指令根据预测栈中的地址信息发生预测错误,则根据恢复栈中的地址信息校正预测栈中的地址信息,可以包括:



[0066] 获取预测错误信息;预测错误信息包括待执行分支指令对应的预测栈的栈顶指针和待对应的恢复栈的恢复栈编号;根据恢复栈编号对应的恢复栈中的地址信息校正栈顶指针对应的预测栈中的地址信息。

[0067] 若待执行分支指令执行一端时间后,获取到包含该待执行分支指令的预测错误信息,则可以根据预测错误信息中包括的预测栈的栈顶指针和待对应的恢复栈的恢复栈编号,将恢复栈编号对应的恢复栈中的地址信息复制到栈顶指针对应的预测栈中。由此可见,通过为预测栈构建相应的恢复栈,使得在待执行分支指令在预测错误时,可以根据恢复栈编号,将恢复栈编号对应的恢复栈中的地址信息恢复至栈顶指针对应的预测栈中,从而实现对预测栈中的地址信息的校正,进而提高了预测栈中地址信息的准确率。

[0068] 本发明实施例提供的预测栈的错误恢复方法,配置有对预测栈中的地址信息进行备份的至少一个恢复栈;在根据待执行分支指令的类型读写预测栈后,将预测栈中已有的分支指令的地址信息复制到其中一个恢复栈;若待执行分支指令根据预测栈中的地址信息发生预测错误,则根据恢复栈中的地址信息校正预测栈中的地址信息。这样,通过为预测栈构建相应的恢复栈,使得在待执行分支指令在预测错误时,可以根据恢复栈编号,将恢复栈编号对应的恢复栈中的地址信息恢复至栈顶指针对应的预测栈中,从而提高了返回地址预测栈的准确率。

[0069] 为了更好地复用预先配置的至少一个恢复栈,则在至少一个恢复栈中每一个恢复栈被占用之后,在新的待执行分支指令读写预测栈之前,可以先清空其中一个恢复栈中的地址信息,从而为新的待执行分支指令提供空的恢复栈,其具体实现方式可以参见图3所示,图3为本发明实施例提供的一种新的恢复栈的获取方式的示意图。

[0070] S301、在新的待执行分支指令读写预测栈时,根据预设规则将恢复栈中的地址信息清空,得到新的恢复栈。

[0071] 可选的,当待执行分支指令的数量大于预设阈值时,在新的待执行分支指令读写预测栈时,根据恢复栈的编号将恢复栈中的地址信息清空,得到新的恢复栈。

[0072] 需要说明的是,由于其他类分支指令不专门占用恢复栈,因此,此处的待执行分支指令的数量是指调用类分支指令和返回类分支指令的数量之和。预设阈值的值具体可以根据预先配置的至少一个恢复栈的个数确定。示例的,若预先配置八个恢复栈,则预设阈值的值为8,当然,本发明实施例只是以预先配置八个恢复栈,且预设阈值的值为8为例进行说明,具体可以根据实际需要进行设置。

[0073] S302、将预测栈中已有的分支指令的地址信息复制到新的恢复栈。

[0074] 示例的,在第九个待执行分支指令读写预测栈时,由于前八个待执行分支指令(为调用类分支指令或返回类分支指令,其他类分支指令不专门占用恢复栈)对应的预测栈已经相应占用了预先配置的八个恢复栈,则可以按顺序清空恢复栈编号为1的恢复栈中的地址信息,以释放恢复栈编号为1的恢复栈中的地址信息,从而得到新的恢复栈,再将第九个待执行分支指令对应的预测栈中的地址信息复制到该新的恢复栈中。在第十个待执行分支指令读写预测栈时,可以按顺序清空恢复栈编号为2的恢复栈中的地址信息,以释放恢复栈编号为2的恢复栈中的地址信息,从而得到新的恢复栈,再将第十个待执行分支指令对应的预测栈中的地址信息复制到该新的恢复栈中,以此类推。

[0075] 由此可见,通过上述S301和S302,当待执行分支指令的数量大于预先配置的恢复

栈的个数时,可以根据恢复栈编号将预先配置的恢复栈中的地址信息清空,以释放恢复栈中的地址信息,之后再该待执行分支指令对应的预测栈中的地址信息复制到新的恢复栈,以实现恢复栈的复用,从而提高了恢复栈的利用率。

[0076] 为了更清楚地说明本发明实施例所示的技术方案,假设有五条分支指令,分别为:{A函数调用类分支指令、B函数调用类分支指令、C函数返回类分支指令、D其他类分支指令、E函数调用类分支指令}。

[0077] 在A函数调用类分支指令读写预测栈时,预测栈栈顶指针RAS\_TOS的初始值为0,将A函数调用类分支指令对应的PC+8的地址写入到预测栈中,并将预测栈的栈顶指针RAS\_TOS加1,此时预测栈中包括的是A函数调用类分支指令的PC+8的地址信息,将预测栈中已有的A函数调用类分支指令对应的PC+8的地址复制到恢复栈编号RAS\_REC=1的恢复栈中,则恢复栈编号RAS\_REC=1的恢复栈也备份了A函数调用类分支指令的地址信息,则A函数调用类分支指令对应的恢复栈为恢复栈编号RAS\_REC=1的恢复栈。示例的,请参见图4所示,图4为本发明实施例提供的一种A函数调用类分支指令读写预测栈的示意图。

[0078] 在B函数调用类分支指令读写预测栈时,将B函数调用类分支指令对应的PC+8的地址写入到预测栈中,并将预测栈的栈顶指针RAS\_TOS加1,此时预测栈中包括的是A函数调用类分支指令的PC+8的地址信息和B函数调用类分支指令的PC+8的地址,将预测栈中已有的地址信息复制到恢复栈编号RAS\_REC=2的恢复栈中,则恢复栈编号RAS\_REC=2的恢复栈中也顺序备份了A函数调用类分支指令对应的PC+8的地址和B函数调用类分支指令对应的PC+8的地址,则B函数调用类分支指令对应的恢复栈为恢复栈编号RAS\_REC=2的恢复栈。示例的,请参见图5所示,图5为本发明实施例提供的一种B函数调用类分支指令读写预测栈的示意图。

[0079] 在C函数返回类分支指令读写预测栈时,将预测栈的栈顶指针RAS\_TOS=1对应的B函数调用类分支指令的PC+8的地址确定为C函数返回类分支指令的预测地址信息,对预测栈中的B函数调用类分支指令的PC+8的地址执行出栈操作,此时预测栈包括的是A函数调用类分支指令的PC+8的地址,将预测栈中已有的地址信息复制到恢复栈编号RAS\_REC=3的恢复栈中,则恢复栈编号RAS\_REC=3的恢复栈中也只备份了A函数调用类分支指令的PC+8的地址,则C函数返回类分支指令对应的恢复栈编号RAS\_REC=3的恢复栈。示例的,请参见图6所示,图6为本发明实施例提供的一种C函数返回类分支指令读写预测栈的示意图。

[0080] 一段时间之后,若获取到B函数调用类分支指令的预测错误信息,该B函数调用类分支指令的预测错误信息包括B函数调用类分支指令对应的预测栈的栈顶指针和B函数调用类分支指令对应的恢复栈编号RAS\_REC=2,则可以将恢复栈编号RAS\_REC=2的恢复栈中的地址信息复制到栈顶指针对应的预测栈中,从而完成对预测栈中的地址信息的校正。请参见图7所示,图7为本发明实施例提供的一种恢复预测栈的示意图,这样,在B函数调用类分支指令读写预测栈时,若B函数调用类分支指令预测错误,则根据B函数调用类分支指令对应的恢复栈编号RAS\_REC=2,将恢复栈编号RAS\_REC=2的恢复栈中的地址信息复制到栈顶指针对应的预测栈中。这样,通过为预测栈构建相应的恢复栈,使得在待执行分支指令预测错误时,可以根据恢复栈编号,将恢复栈编号对应的恢复栈中的地址信息恢复至栈顶指针对应的预测栈中,从而提高了返回地址预测栈的准确率。

[0081] 示例的,若继续执行D其他类分支指令,由于D指令为其他类分支指令,在执行D其

他类分支指令时，D其他类分支指令对应的PC+8的地址不写入预测栈，也不在恢复栈中备份，因此，D其他类分支指令携带的还是前一条C函数返回类分支指令对应的预测栈的栈顶指针和C函数返回类分支指令对应的恢复栈编号RAS\_REC=3。

[0082] 一段时间之后，若获取到C函数返回类分支指令的预测错误信息，该C函数返回类分支指令的预测错误信息包括C函数返回类分支指令对应的预测栈的栈顶指针和C函数返回类分支指令对应的恢复栈编号RAS\_REC=3，则将恢复栈编号RAS\_REC=3的恢复栈中的地址信息复制到栈顶指针对应的预测栈中，从而完成对预测栈中的地址信息的校正。请参见图8所示，图8为本发明实施例提供的另一种恢复预测栈的示意图，这样，在C函数返回类分支指令读写预测栈时，在确定C函数返回类分支指令预测错误时，根据C函数返回类分支指令对应的恢复栈编号RAS\_REC=3，将恢复栈编号RAS\_REC=3的恢复栈中的地址信息复制到栈顶指针对应的预测栈中。这样，通过为预测栈构建相应的恢复栈，使得在待执行分支指令在预测错误时，可以根据恢复栈编号，将恢复栈编号对应的恢复栈中的地址信息恢复至栈顶指针对应的预测栈中，从而提高了返回地址预测栈的准确率。

[0083] 示例的，若继续执行E函数调用类分支指令，在E函数调用类分支指令读写预测栈时，将E函数调用类分支指令对应的PC+8的地址写入到预测栈中，并将预测栈的栈顶指针RAS\_TOS加1，此时预测栈中包括的是A函数调用类分支指令的PC+8的地址信息和E函数调用类分支指令的PC+8的地址信息；将预测栈中已有的地址信息复制到恢复栈编号RAS\_REC=4的恢复栈中，则恢复栈编号RAS\_REC=4的恢复栈中也顺序备份了A函数调用类分支指令的PC+8的地址信息和E函数调用类分支指令的PC+8的地址信息，则E函数调用类分支指令对应的恢复栈为恢复栈编号RAS\_REC=4的恢复栈。示例的，请参见图9所示，图9为本发明实施例提供的一种E函数调用类分支指令读写预测栈的示意图。

[0084] 一段时间之后，若获取到D其他类分支指令的预测错误信息，该D其他类分支指令的预测错误信息包括D其他类分支指令对应的预测栈的栈顶指针和D其他类分支指令对应的恢复栈编号RAS\_REC=3，则将恢复栈编号RAS\_REC=3的恢复栈中的地址信息复制到栈顶指针对应的预测栈中，从而完成对预测栈中的地址信息的校正。请参见图10所示，图10为本发明实施例提供的再一种恢复预测栈的示意图，这样，在D其他类分支指令读写预测栈时，在确定D其他类分支指令预测错误时，根据D其他类分支指令对应的恢复栈编号RAS\_REC=3，将恢复栈编号RAS\_REC=3的恢复栈中的地址信息复制到栈顶指针对应的预测栈中。这样，通过为预测栈构建相应的恢复栈，使得在待执行分支指令在预测错误时，可以根据恢复栈编号，将恢复栈编号对应的恢复栈中的地址信息恢复至栈顶指针对应的预测栈中，从而提高了返回地址预测栈的准确率。

[0085] 图11为本发明实施例提供的一种预测栈的错误恢复装置11的结构示意图，请参见图11所示，配置有对预测栈中的地址信息进行备份的至少一个恢复栈；该预测栈的错误恢复装置11可以包括：

[0086] 复制单元1101，用于在根据待执行分支指令的类型读写预测栈后，将预测栈中已有的分支指令的地址信息复制到其中一个恢复栈。

[0087] 校正单元1102，用于若待执行分支指令根据预测栈中的地址信息发生预测错误，则根据恢复栈中的地址信息校正预测栈中的地址信息。

[0088] 可选的，至少一个恢复栈中的每个恢复栈有相应的恢复栈编号。

[0089] 复制单元1101,具体用于在待执行分支指令的类型为函数调用类分支指令时,在预测栈的栈顶指针中写入待执行分支指令的地址信息;将预测栈中已有的分支指令的地址信息复制到对应的恢复栈,待执行分支指令携带预测栈的栈顶指针和对应的恢复栈的恢复栈编号;

[0090] 复制单元1101,具体用于在待执行分支指令的类型为函数返回类分支指令时,将预测栈的栈顶指针的地址信息确定为待执行分支指令的预测地址信息,并将栈顶指针中的地址信息从预测栈中读出;将预测栈中已有的分支指令的地址信息复制到对应的恢复栈,待执行分支指令携带预测栈的栈顶指针和对应的恢复栈的恢复栈编号;

[0091] 复制单元1101,具体用于在待执行分支指令的类型为其他类分支指令时,确定待执行分支指令的前一条分支指令对应的恢复栈;将前一条分支指令对应的恢复栈确定为待执行分支指令对应的恢复栈,待执行分支指令携带预测栈的栈顶指针和对应的恢复栈的恢复栈编号。

[0092] 可选的,校正单元1102,具体用于获取预测错误信息;预测错误信息包括待执行分支指令对应的预测栈的栈顶指针和待对应的恢复栈的恢复栈编号;并根据恢复栈编号对应的恢复栈中的地址信息校正栈顶指针对应的预测栈中的地址信息。

[0093] 可选的,该预测栈的错误恢复装置11还可以包括清空单元1103,请参见图12所示,图12为本发明实施例提供的另一种预测栈的错误恢复装置11的结构示意图。

[0094] 清空单元1103,用于在新的待执行分支指令读写预测栈时,根据预设规则将恢复栈中的地址信息清空,得到新的恢复栈。

[0095] 复制单元1101,还用于将预测栈中已有的分支指令的地址信息复制到新的恢复栈。

[0096] 可选的,清空单元1103,具体用于当待执行分支指令的数量大于预设阈值时,在新的待执行分支指令读写预测栈时,根据恢复栈的编号将恢复栈中的地址信息清空,得到新的恢复栈。

[0097] 上述预测栈的错误恢复装置11,对应地可执行任一实施例的预测栈的错误恢复方法的技术方案,其实现原理和技术效果类似,在此不再赘述。

[0098] 本发明实施例还提供一种计算机可读存储介质,计算机可读存储介质上存储有计算机程序,在计算机程序被处理器执行时,执行上述任一实施例所示的预测栈的错误恢复方法。

[0099] 上述计算机可读存储介质,对应地可执行任一实施例的预测栈的错误恢复方法的技术方案,其实现原理和技术效果类似,在此不再赘述。

[0100] 图13为本发明提供的一种电子设备13的结构示意图,请参见图13所示,该电子设备13可以包括:

[0101] 处理器1301和存储器1302,其中,存储器1302用于存储程序指令。

[0102] 处理器1301用于读取存储器1302中的程序指令,并根据存储器1302中的程序指令执行上述任一实施例所示的预测栈的错误恢复方法。

[0103] 应理解的是,处理器1301可以是中央处理单元(Central Processing Unit,CPU),还可以是其他通用处理器、数字信号处理器(Digital Signal Processor,DSP)、专用集成电路(Application Specific Integrated Circuit,ASIC)等。通用处理器可以是微处理器

或者该处理器1301也可以是任何常规的处理器等。结合本申请所公开的方法的步骤可以直接体现为硬件处理器执行完成,或者用处理器1301中的硬件及软件模块组合执行完成。

[0104] 实现上述各方法实施例的全部或部分步骤可以通过程序指令相关的硬件来完成。前述的程序可以存储于一可读取存储器中。该程序在执行时,执行包括上述各方法实施例的步骤;而前述的存储器(存储介质)包括:只读存储器(read-only memory,ROM)、RAM、快闪存储器、硬盘、固态硬盘、磁带(magnetic tape)、软盘(floppy disk)、光盘(optical disc)及其任意组合。

[0105] 上述电子设备13,对应地可执行任一实施例的预测栈的错误恢复方法的技术方案,其实现原理和技术效果类似,在此不再赘述。

[0106] 本领域技术人员在考虑说明书及实践这里公开的发明后,将容易想到本公开的其他实施方案。本发明旨在涵盖本公开的任何变型、用途或者适应性变化,这些变型、用途或者适应性变化遵循本公开的一般性原理并包括本公开未公开的本技术领域中的公知常识或惯用技术手段。说明书和实施例仅被视为示例性的,本公开的真正范围和精神由下面的权利要求书指出。

[0107] 应当理解的是,本公开并不局限于上面已经描述并在附图中示出的精确结构,并且可以在不脱离其范围进行各种修改和改变。本公开的范围仅由所附的权利要求书来限制。

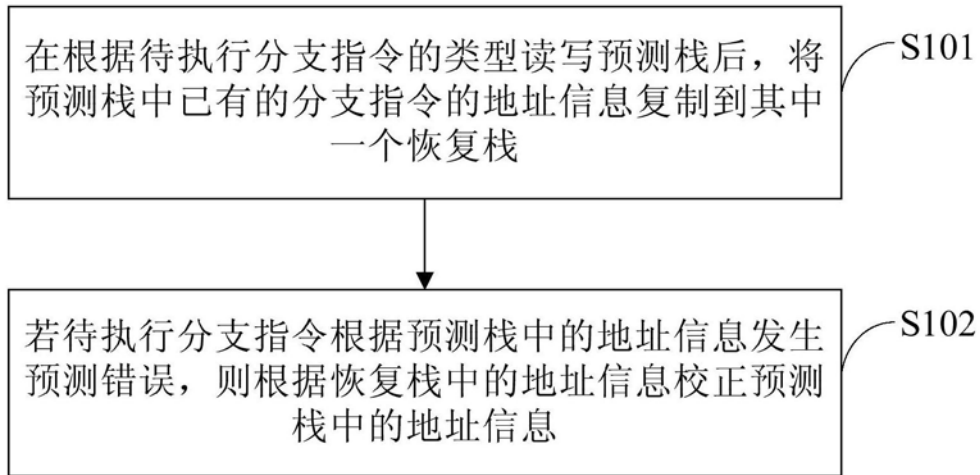


图1

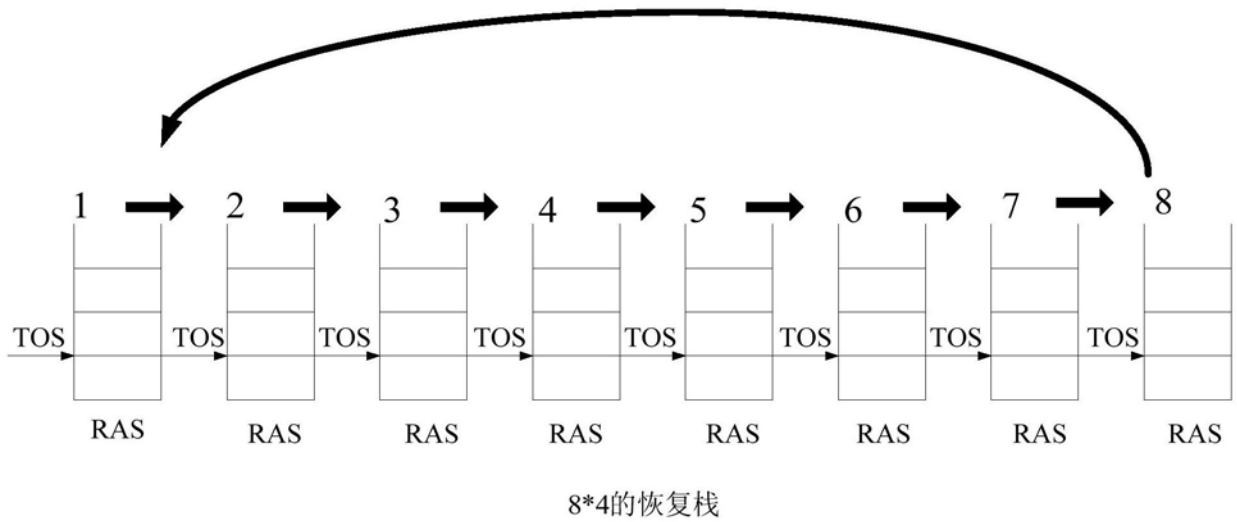


图2

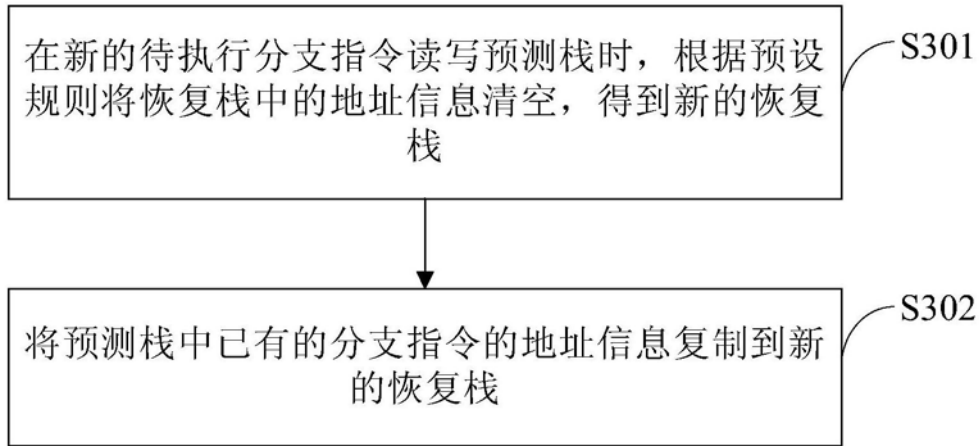


图3

A函数调用类分支指令



图4

B函数调用类分支指令



图5

C函数返回类分支指令



图6



恢复预测栈

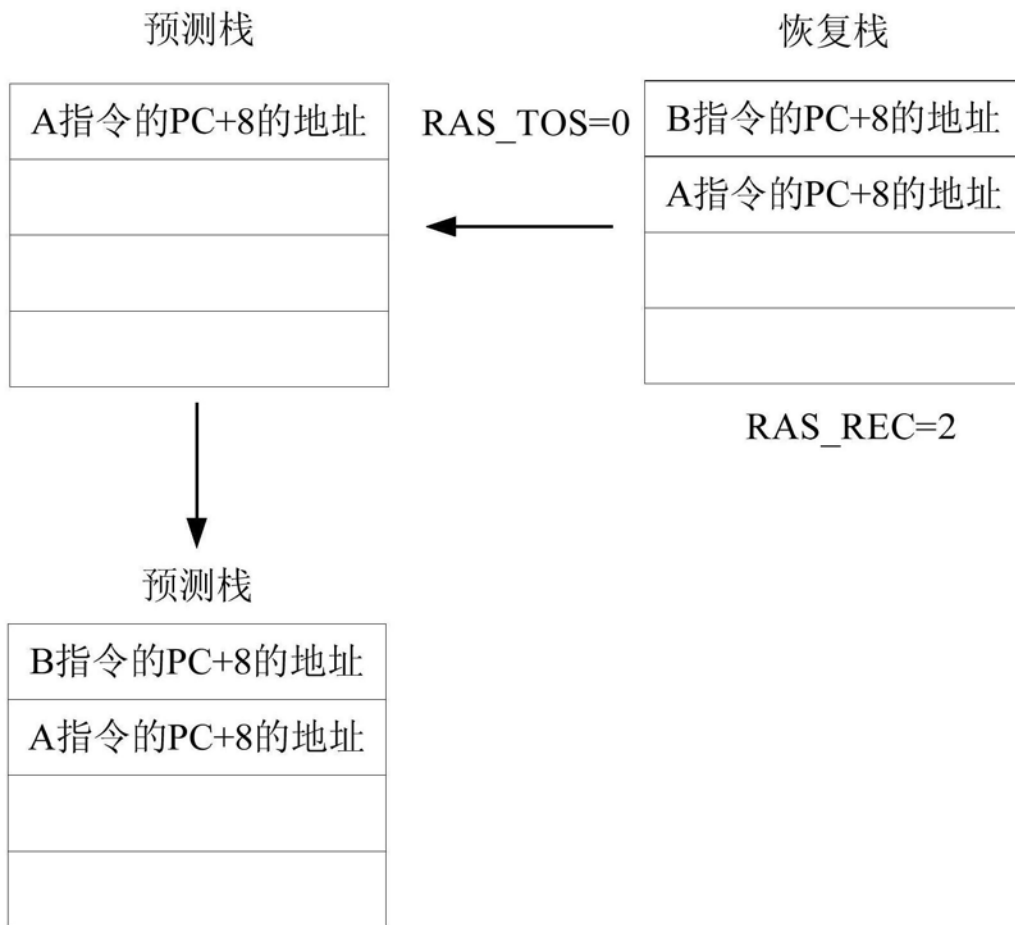


图7

恢复预测栈

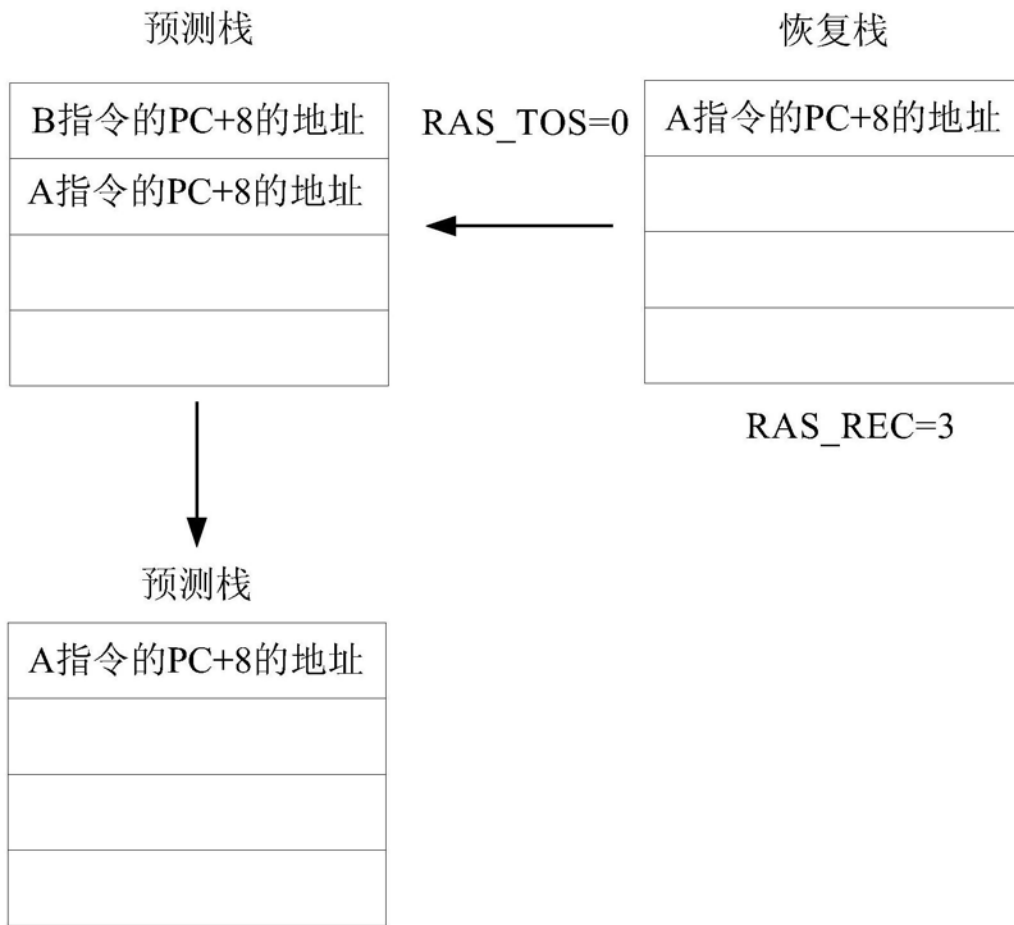


图8

E函数调用类指令经过

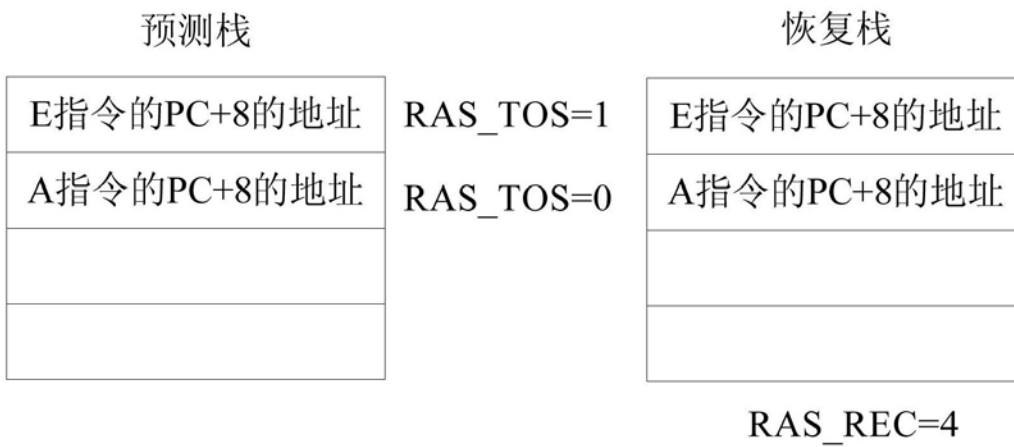


图9

恢复预测栈

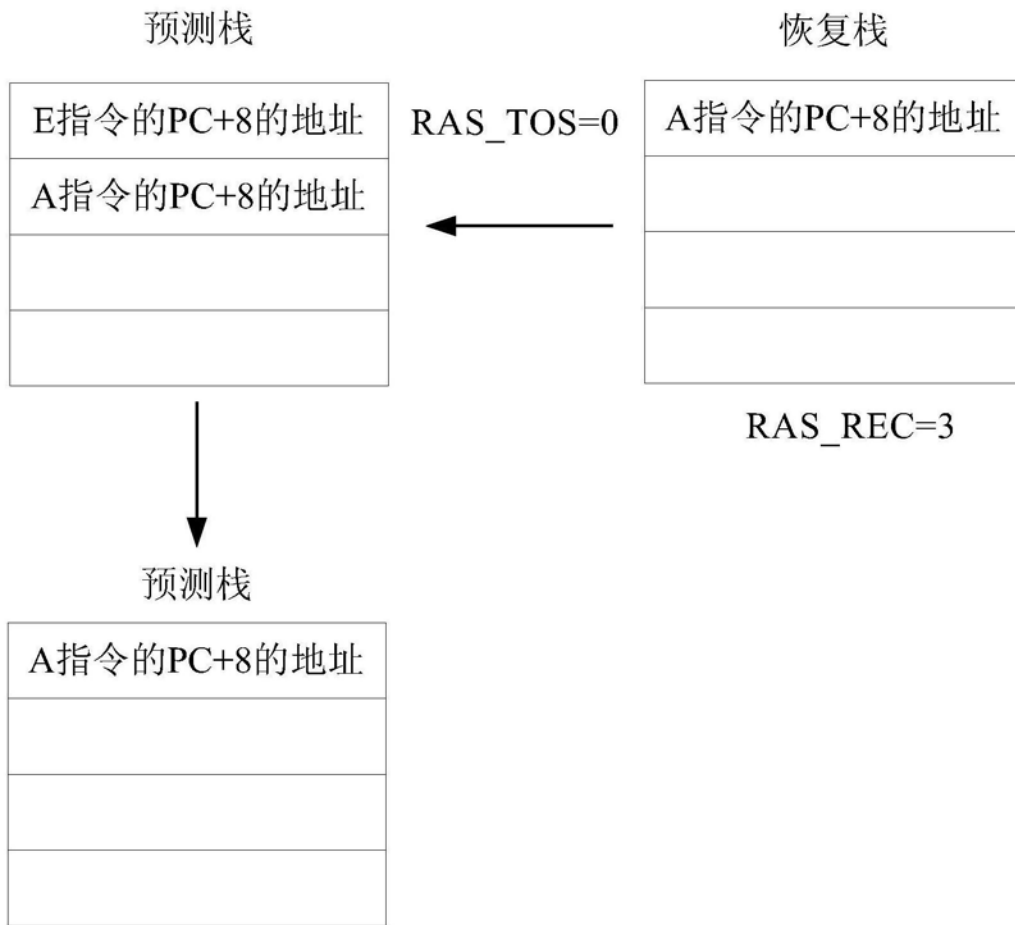


图10

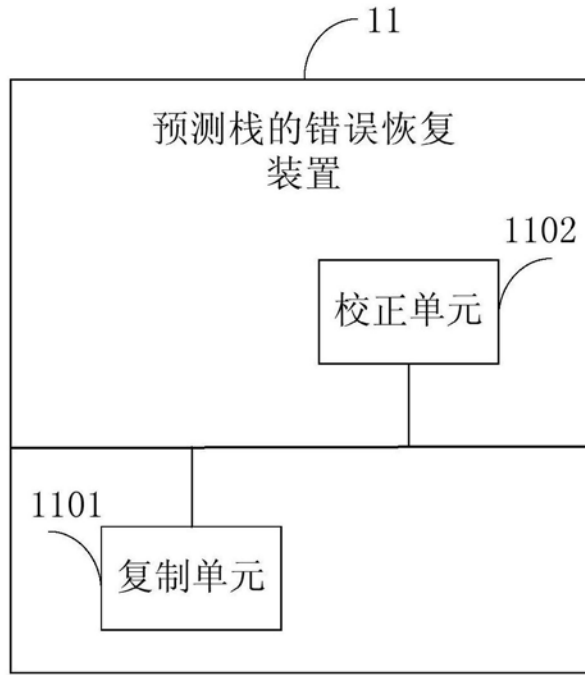


图11

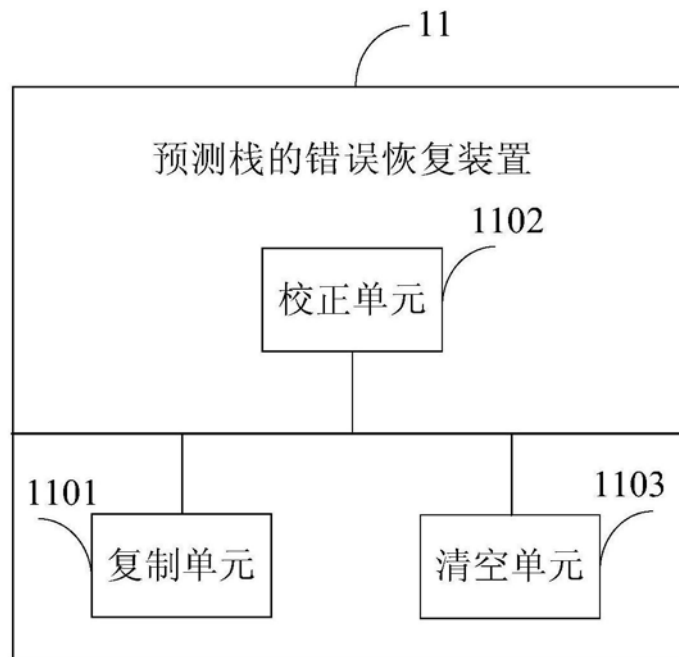


图12

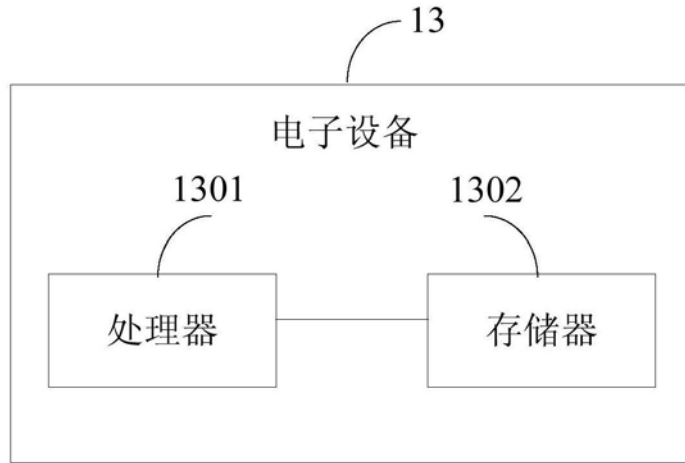


图13