(12) 按照专利合作条约所公布的国际申请

(19) 世界知识产权组织
国 际 局

(43) 国际公布日
2015 年 5 月 7 日 (07.05.2015)

WIPO | PCT

(10) 国际公布号
WO 2015/062199 A1

(51) 国际专利分类号:
G06F 9/48 (2006.01)

(21) 国际申请号: PCT/CN2014/074412

(22) 国际申请日: 2014 年 3 月 31 日 (31.03.2014)

(25) 申请语言: 中文

(26) 公布语言: 中文

(30) 优先权:
201310531026.4 2013 年 10 月 31 日 (31.10.2013) CN

(71) 申请人: **华为技术有限公司 (HUAWEI TECHNOLOGIES CO., LTD.)** [CN/CN]; 中国广东省深圳市龙岗区坂田华为总部办公楼, Guangdong 518129 (CN)。

(72) 发明人: **藏洪永 (ZANG, Hongyong)**; 中国广东省深圳市龙岗区坂田华为总部办公楼, Guangdong 518129 (CN)。 **张浩宇 (ZHANG, Haoyu)**; 中国广东省深圳市龙岗区坂田华为总部办公楼, Guangdong 518129 (CN)。

(74) 代理人: **广州三环专利代理有限公司 (GUANGZHOU SCIHEAD PATENT AGENT CO.. LTD)**; 中国广东省广州市越秀区先烈中路 80 号汇华商贸大厦 1508 室, Guangdong 510070 (CN)。

(81) 指定国 (除另有指明，要求每一种可提供的国家保护): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JP, KE, KG, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW。

(84) 指定国 (除另有指明，要求每一种可提供的地区保护): ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, SZ, TZ, UG, ZM, ZW), 欧亚 (AM, AZ, BY, KG, KZ, RU, TJ, TM), 欧洲 (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG)。

本国际公布:

— 包括国际检索报告(条约第 21 条(3))。

(54) **Title**: METHOD FOR IMPLEMENTING AN INTERRUPT BETWEEN VIRTUAL PROCESSORS, RELATED DEVICE, AND SYSTEM

(54) **发明名称** : 一种虚拟处理器之间的中断的实现方法、相关装置和系统

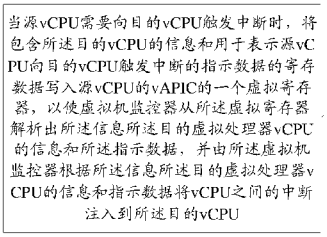


图 2 / FIG. 2

201 When source vCPU needs to trigger interrupt with target vCPU, writing register data, containing information of said target vCPU and instruction data used for representing source vCPU triggering interrupt with target vCPU, into virtual register of vAPIC of source vCPU, such that virtual machine monitor can parse from said virtual register said target vCPU information and said instruction data, and said virtual machine monitor injecting, according to said target vCPU information and instruction data, the interrupt between vCPUs into said target vCPU

(57) **Abstract**: The embodiments of the present invention disclose a method for implementing an interrupt between virtual processors (vCPU), comprising: when a source virtual processor needs to trigger an interrupt with a target virtual processor, writing the register data, containing information of said target virtual processor and the instruction data used for representing the source virtual processor triggering an interrupt with the target virtual processor, into a virtual register of the virtual advanced programmable interrupt controller (vAPIC) of the source virtual processor, such that a virtual machine monitor can parse from said virtual register said target virtual processor information and said instruction data, and said virtual machine monitor injecting, according to said target virtual processor information and instruction data, the interrupt between the virtual processors into said target virtual processor. Accordingly, the embodiments of the present invention also provide a related virtual machine, virtual controller, compute node, and system. The embodiments of the present invention improve the performance of a virtual machine.

(57) **摘要**:

[见续页]

WO 2015/062199 A1

本发明实施例公开了一种虚拟处理器之间中断的实现方法，包括：当源虚拟处理器需要向目的虚拟处理器触发中断时，将包含所述目的虚拟处理器的信息和用于表示源虚拟处理器向目的虚拟处理器触发中断的指示数据的寄存数据写入源虚拟处理器的虚拟高级可编程中断控制器 vAPIC 的一个虚拟寄存器，以使虚拟机监控器从所述虚拟寄存器解析出所述目的虚拟处理器的信息和所述指示数据，并由所述虚拟机监控器根据所述目的虚拟处理器的信息和指示数据将虚拟处理器之间的中断注入到所述目的虚拟处理器。相应地，本发明实施例还提供相关的虚拟机、虚拟控制器、计算节点和系统。本发明实施例可以提高虚拟机的性能。

一种虚拟处理器之间的中断的实现方法、相关装置和系统

本申请要求于 2013 年 10 月 31 日提交中国专利局、申请号为 201310531026.4 、发明名称为“一种虚拟处理器之间的中断的实现方法、相关装置和系统”的中国专利申请的优先权，其全部内容通过引用结合在本申请中。

技术领域

本发明涉及通信领域，尤其涉及一种虚拟处理器之间的中断的实现方法、相关装置和系统。

背景技术

目前虚拟机已经得到广泛应用，在虚拟机应用中经常用出现中断，例如:在多处理器系统（Symmetrical Multi-Processing，SMP）中当运行多任务且未绑定时，会由于负载均衡调度会频繁发生中断，如，发生处理器之间的中断（Inter-Processor Interrupt，IPI），当然还可以是其它中断。而中断都是通过访问高级可编程中断控制器（Advanced Programmable Interrupt Controller，APIC）的寄存器实现的。具体实现过程如下:

读取 APIC 的虚拟中断命令寄存器（virtual Interrupt Command Register，vICR）中的传送状态（delivery status）值;

根据所述 delivery status 值向 ICR 的高位寄存器中写入目的虚拟处理器的本地 APIC（virtual Central Processing Unit Local APIC，vLocal APIC）的标识;

向 ICR 的低位寄存器写入用于指示目的 vLocal APIC 所属的目的虚拟处理器（virtual Central Processing Unit，vCPU）发生中断的指示信息，使虚拟机监控器（Virtual Machine Monitor，VMM）解析出所述 ICR 的高位寄存器寄存的上述标识，以及解析出 ICR 的低位寄存器寄存指示信息，并将该标识和指示信息转发至目的 vLocal APIC，当上目的 vLocal APIC 接收到上述中断信息时就可以控制目的 vCPU 产生中断。

可见，上述中断实现过程由于需要向 APIC 执行三次操作，即导致三次虚拟机退出（Virtual Machine exit，VM-Exit），其中，VM-Exit 具体可以理解虚

拟机与 Hypervisor之间的上下文切换。而在实际应用中虚拟处理器是会频繁的出现中断，即虚拟机会更频繁地出现 VM-Exit，从而导致虚拟机的性能下降。

发明内容

本发明实施例提供了一种 vCPU 之间的中断的实现方法、相关装置和系统，以提高虚拟机的性能。

第一方面，本发明实施例提供的一种 vCPU之间中断的实现方法，包括:

当源 vCPU需要向目的 vCPU触发中断时，

将包含所述目的vCPU的信息和用于表示源 vCPU向目的vCPU触发中断的指示数据的寄存数据写入源 vCPU 的虚拟高级可编程中断控制器 vAPIC 的一个虚拟寄存器，以使虚拟机监控器从所述虚拟寄存器解析出所述目的 vCPU的信息和所述指示数据，并由所述虚拟机监控器根据所述目的 vCPU的信息和指示数据将 vCPU之间的中断注入到所述目的 vCPU。

在第一方面的第一种可能的实现方式中，所述虚拟寄存器为虚拟中断命令寄存器 vICR 的高位寄存器或低位寄存器；所述将包含所述目的 vCPU的信息和用于表示源 vCPU 向目的 vCPU 触发中断的指示数据的寄存数据写入源 vCPU的 vAPIC 的一个虚拟寄存器，包括:

将包含所述目的vCPU的信息和用于表示源 vCPU向目的vCPU触发中断的指示数据的寄存数据写入源 vCPU的 vAPIC 的所述 vICR 的高位寄存器或者低位寄存器。

结合第一方面的第一种可能的实现方式，在第一方面的第二种可能的实现方式中，所述将包含所述目的 vCPU的信息和用于表示源 vCPU向目的 vCPU触发中断的指示数据的寄存数据写入源 vCPU的 vAPIC 的所述 vICR 的高位寄存器或者低位寄存器，包括:

将包含所述目的vCPU的信息和用于表示源 vCPU向目的vCPU触发中断的指示数据的寄存数据写入源 vCPU的 vAPIC 的所述 vICR 的低位寄存器，其中，所述目的 vCPU的信息被填入在所述低位寄存器的保留位；或者

将包含所述目的vCPU的信息和用于表示源 vCPU向目的vCPU触发中断的指示数据的寄存数据写入源 vCPU的 vAPIC 的所述 vICR 的高位寄存器，其

中，所述指示数据被填入在所述高位寄存器的保留位。

结合第一方面，在第一方面的第三种可能的实现方式中，所述将包含所述目的 vCPU 的信息和用于表示源 vCPU 向目的 vCPU 触发中断的指示数据的寄存数据写入源 vCPU 的 vAPIC 的一个虚拟寄存器，包括:

将包含所述目的 vCPU 的信息和用于表示源 vCPU 向目的 vCPU 触发中断的指示数据和优化数据的寄存数据写入源 vCPU 的 vAPIC 的一个虚拟寄存器，其中，所述优化数据被填入在所述虚拟寄存器的保留位，所述优化数据用于指示所述 VMM 从所述虚拟寄存器中解析出所述目的 vCPU 的信息和所述指示数据。

结合第一方面，在第一方面的第四种可能的实现方式中，所述将包含所述目的 vCPU 的信息和用于表示源 vCPU 向目的 vCPU 触发中断的指示数据的寄存数据写入源 vCPU 的 vAPIC 的一个虚拟寄存器，包括:

将包含所述目的 vCPU 的信息和用于表示源 vCPU 向目的 vCPU 触发中断的指示数据的寄存数据写入源 vCPU 的 vAPIC 的一个虚拟寄存器，其中，所述目的 vCPU 的信息被填入在所述虚拟寄存器的保留位，所述指示数据包括被填入在所述虚拟寄存器的航向域、传送模域、目的模型域、等级域、触发模型域和目的速记域的数据。

第二方面，本发明实施例提供一种虚拟机，包括：一个或多个 vCPU，所述 vCPU 包括虚拟高级可编程中断控制器 vAPIC，所述 vAPIC 包括虚拟寄存器，其中:

所述一个或多个 vCPU 中当前 vCPU 用于:

当所述当前 vCPU 需要向目的 vCPU 触发中断时，包含所述目的 vCPU 的信息和用于表示所述当前 vCPU 向所述目的 vCPU 触发中断的指示数据的寄存数据包写入所述当前 vCPU 的 vAPIC 的一个虚拟寄存器，以使虚拟机监控器从该虚拟寄存器解析出所述目的 vCPU 的信息和所述指示数据，并由所述虚拟机监控器根据所述目的 vCPU 的信息和指示数据将 vCPU 之间的中断注入所述目的 vCPU。

在第二方面的第一种可能的实现方式中，所述虚拟寄存器为虚拟中断命令寄存器 vICR 的高位寄存器或低位寄存器；所述包含所述目的 vCPU 的信息和

用于表示所述当前vCPU向所述目的vCPU触发中断的指示数据的寄存数据包写入所述当前 vCPU 的 vAPIC 的一个虚拟寄存器的操作上，所述当前 vCPU 具体用于:

将包含所述目的vCPU的信息和用于表示所述当前vCPU向目的vCPU触发中断的指示数据的寄存数据写入所述当前vCPU的vAPIC的vICR的高位寄存器或者低位寄存器。

结合第二方面的第一种可能的实现方式，在第二方面的第二种可能的实现方式中，所述将包含所述目的 vCPU 的信息和用于表示所述当前 vCPU 向目的 vCPU 触发中断的指示数据的寄存数据写入所述当前 vCPU 的 vAPIC 的 vICR 的高位寄存器或者低位寄存器的操作上，所述 vCPU 具体用于:

将包含所述目的vCPU的信息和用于表示所述当前vCPU向目的vCPU触发中断的指示数据的寄存数据写入所述当前vCPU的vAPIC的vICR的低位寄存器，其中，所述目的 vCPU 的信息被填入在所述低位寄存器的保留位; 或者

将包含所述目的vCPU的信息和用于表示所述当前vCPU向目的vCPU触发中断的指示数据的寄存数据写入所述当前vCPU的vAPIC的vICR的高位寄存器，其中，所述指示数据被填入在所述高位寄存器的保留位。

结合第二方面，在第二方面的第三种可能的实现方式中，所述包含所述目的 vCPU 的信息和用于表示所述当前 vCPU 向所述目的 vCPU 触发中断的指示数据的寄存数据包写入所述当前 vCPU 的 vAPIC 的一个虚拟寄存器的操作上，所述当前 vCPU 具体用于:

将包含所述目的vCPU的信息和用于表示所述当前vCPU向目的vCPU触发中断的指示数据和优化数据的寄存数据写入所述当前 vCPU 的 vAPIC 的一个虚拟寄存器，其中，所述优化数据被填入在该虚拟寄存器的保留位，所述优化数据用于指示所述 VMM 从该虚拟寄存器中解析出所述目的 vCPU 的信息和所述指示数据。

结合第二方面，在第二方面的第四种可能的实现方式中，所述包含所述目的 vCPU 的信息和用于表示所述当前 vCPU 向所述目的 vCPU 触发中断的指示数据的寄存数据包写入所述当前 vCPU 的 vAPIC 的一个虚拟寄存器的操作上，所述当前 vCPU 具体用于:

将包含所述目的vCPU的信息和用于表示所述当前vCPU向目的vCPU触发中断的指示数据的寄存数据写入所述当前vCPU的vAPIC的一个虚拟寄存器，其中，所述目的vCPU的信息被填入在该虚拟寄存器的保留位，所述指示数据包括被填入在该虚拟寄存器的航向域、传送模域、目的模型域、等级域、触发模型域和目的速记域的数据。

第三方面，本发明实施例提供一种计算节点，包括：硬件层、运行在所述硬件层之上的VMM、以及运行在所述VMM之上的至少一个虚拟机，其中，所述至少一个虚拟机包括第一虚拟机和第二虚拟机，第一虚拟机包括第一vCPU，所述第一vCPU包括第一虚拟高级可编程中断控制器vAPIC，所述第一vAPIC包括第一虚拟寄存器，第二虚拟机包括第二vCPU，其中：

所述第一vCPU，用于当第一vCPU需要向第二vCPU触发中断时，将包含所述第二vCPU的信息和用于表示所述第一vCPU向所述第二vCPU触发中断的指示数据的寄存数据包写入所述第一虚拟寄存器；

所述VMM，用于从所述第一虚拟寄存器解析出所述第二vCPU的信息和所述指示数据，并根据所述第二vCPU的信息和指示数据将vCPU之间的中断注入到所述第二vCPU，其中所述vCPU之间的中断包括第一vCPU与第二vCPU之间的中断。

在第三方面的第一种可能的实现方式中，所述第一虚拟寄存器为为虚拟中断命令寄存器vICR的高位寄存器或低位寄存器；

所述第一vCPU具体用于当第一vCPU需要向第二vCPU触发中断时，将包含所述第二vCPU的信息和用于表示所述第一vCPU向所述第二vCPU触发中断的指示数据的寄存数据包写入所述高位寄存器或低位寄存器；

所述VMM具体用于从所述高位寄存器或低位寄存器解析出所述第二vCPU的信息和所述指示数据，并根据所述第二vCPU的信息和指示数据将vCPU之间的中断注入到所述第二vCPU，其中所述vCPU之间的中断包括第一vCPU与第二vCPU之间的中断。

结合第三方面的第一种可能的实现方式，在第三方面的第二种可能的实现方式中，所述第一vCPU具体用于当第一vCPU需要向第二vCPU触发中断时，将包含所述第二vCPU的信息和用于表示所述第一vCPU向所述第二vCPU触

发中断的指示数据的寄存数据包写入所述低位寄存器，其中，所述第二 vCPU 的信息被填入在所述低位寄存器的保留位;

所述 VMM 具体用于从所述低位寄存器的保留位解析出所述第二 vCPU 的信息，以及从所述低位寄存器解析出所述指示数据，并根据所述第二 vCPU 的信息和指示数据将 vCPU 之间的中断注入到所述第二 vCPU，其中所述 vCPU 之间的中断包括第一 vCPU 与第二 vCPU 之间的中断；或者

所述第一 vCPU 具体用于当第一 vCPU 需要向第二 vCPU 触发中断时，将包含所述第二 vCPU 的信息和用于表示所述第一 vCPU 向所述第二 vCPU 触发中断的指示数据的寄存数据包写入所述高位寄存器，其中，所述指示数据被填入在所述高位寄存器的保留位;

所述虚拟机监控器具体用于从所述高位寄存器的保留位解析出所述指示数据，以及从所述高位寄存器解析出所述第二 vCPU 的信息，并根据所述第二 vCPU 的信息和指示数据将 vCPU 之间的中断注入到所述第二 vCPU，其中所述 vCPU 之间的中断包括第一 vCPU 与第二 vCPU 之间的中断。

结合第三方面，在第三方面的第三种可能的实现方式中，所述第一 vCPU 具体用于当第一 vCPU 需要向第二 vCPU 触发中断时，将包含所述第二 vCPU 的信息和用于表示所述第一 vCPU 向所述第二 vCPU 触发中断的指示数据和优化数据的寄存数据包写入所述第一虚拟寄存器，其中，所述优化数据被填入在该虚拟寄存器的保留位，所述优化数据用于指示所述 VMM 从所述第一虚拟寄存器中解析出所述第二 vCPU 的信息和所述指示数据;

所述 VMM 具体用于从所述第一虚拟寄存器解析出所述优化数据，并根据所述优化数据从所述第一虚拟寄存器解析出所述第二 vCPU 的信息和所述指示数据，并根据所述第二 vCPU 的信息和指示数据将 vCPU 之间的中断注入到所述第二 vCPU，其中所述 vCPU 之间的中断包括第一 vCPU 与第二 vCPU 之间的中断。

结合第三方面，在第三方面的第四种可能的实现方式中，所述第一 vCPU 具体用于当第一 vCPU 需要向第二 vCPU 触发中断时，将包含所述第二 vCPU 的信息和用于表示所述第一 vCPU 向所述第二 vCPU 触发中断的指示数据的寄存数据包写入所述第一虚拟寄存器，其中，所述第二 vCPU 的信息被填入在所

述虚拟寄存器的保留位，所述指示数据包括被填入在所述虚拟寄存器的航向域、传送模域、目的模型域、等级域、触发模型域和目的速记域的数据；

所述 VMM 具体用于从所述第一虚拟寄存器的保留位解析出所述第二 vCPU 的信息，以及从所述第一虚拟寄存器的航向域、传送模域、目的模型域、等级域、触发模型域和目的速记域中解析出所述指示数据，并根据所述第二 vCPU 的信息和指示数据将 vCPU 之间的中断注入到所述第二 vCPU，其中所述 vCPU 之间的中断包括第一 vCPU 与第二 vCPU 之间的中断。

第四方面，本发明实施例提供一种计算机系统，包括：第三方面包括的任一计算节点。

上述技术方案中，当源 vCPU 需要向目的 vCPU 触发中断时，将包含所述目的 vCPU 的信息和用于表示源 vCPU 向目的 vCPU 触发中断的指示数据的寄存数据写入源 vCPU 的虚拟高级可编程中断控制器 vAPIC 的一个虚拟寄存器，以使虚拟机监控器从所述虚拟寄存器解析出所述目的 vCPU 的信息和所述指示数据，并由所述虚拟机监控器根据所述目的 vCPU 的信息和指示数据将 vCPU 之间的中断注入到所述目的 vCPU。在上述中断过程只需要执行一次对 vAPIC 的操作，即一次中断只导致一次 VM-Exit，从而可以提高虚拟机的性能。

附图说明

为了更清楚地说明本发明实施例或现有技术中的技术方案，下面将对实施例或现有技术描述中所需要使用的附图作简单地介绍，显而易见地，下面描述中的附图仅仅是本发明的一些实施例，对于本领域普通技术人员来讲，在不付出创造性劳动的前提下，还可以根据这些附图获得其他的附图。

图 1 是本发明实施例可应用的场景示意图；

图 2 是本发明实施例提供的一种 vCPU 之间的中断的实现方法的流程示意图；

图 3 是本发明实施例提供的另一种 vCPU 之间的中断的实现方法的流程示意图；

图 4 和图 5 是本发明实施例提供的可选的寄存数据包示意图；

图 6 是本发明实施例提供的一种虚拟机的结构示意图；

图 7 是本发明实施例提供的另一种虚拟机的结构示意图;

图 8 是本发明实施例提供的一种计算节点的结构示意图;

图 9 是本发明实施例还提供一种计算机系统的结构示意图;

图 10 是本发明实施例提供的另一种计算节点的结构示意图。

具体实施方式

下面将结合本发明实施例中的附图，对本发明实施例中的技术方案进行清楚、完整地描述，显然，所描述的实施例仅仅是本发明一部分实施例，而不是全部的实施例。基于本发明中的实施例，本领域普通技术人员在没有作出创造性劳动前提下所获得的所有其他实施例，都属于本发明保护的范围。

为了方便理解本发明实施例，首先对虚拟化技术进行介绍，虚拟化技术是一种将底层硬件设备与上层操作系统、应用程序分离的去耦合技术，如图1所示，虚拟化技术应用场景中包括硬件层11、VMM层12和虚拟机层13，硬件层11、VMM层12和虚拟机层13定义可以如下:

虚拟机 VM:

通过虚拟机软件可以在一台物理计算机上模拟出一台或者多台虚拟的计算机，而这些虚拟机就像真正的计算机那样进行工作，虚拟机上可以安装操作系统和应用程序，虚拟机还可访问网络资源。对于在虚拟机中运行的应用程序而言，虚拟机就像是在真正的计算机中进行工作。

硬件层:

虚拟化环境运行的硬件平台。其中，硬件层可包括多种硬件，例如某计算节点的硬件层可包括 CPU 和内存，还可以包括网卡、存储器等等高速/低速输入/输出（ I/O，Input/Output）设备，及具有特定处理功能的其它设备，如输入输出内存管理单元（ IOMMU，Input/Output Memory Management Unit ），其中 IOMMU 可用于虚拟机物理地址和 Host 物理地址的转换。

VMM（例如：Hypervisor）:

作为管理层，用以完成硬件资源的管理、分配；为虚拟机呈现虚拟硬件平台；实现虚拟机的调度和隔离。此外，有时 VMM 和 1 个特权虚拟机配合，两者结合组成宿主机 Host。其中，虚拟硬件平台对其上运行的各个虚拟机提供

各种硬件资源，如提供虚拟 CPU、虚拟内存、虚拟磁盘、虚拟网卡等等。其中，该虚拟磁盘可对应 Host 的一个文件或者一个逻辑块设备。虚拟机则运行在 VMM 或 Host 为其准备的虚拟硬件平台上，VMM 或 Host 上运行一个或多个虚拟机。

本发明实施例中的中断具体可以是任何通过访问 APIC 的虚拟寄存器而产生的中断，例如：IPI 中断和其他类型中断。这些中断具体可以应用于多个 vCPU 之间的负载均衡的场景，例如，将进程从一个 vCPU 的等待队列迁移至其它 vCPU 时产生的中断。这些中断还可以是当某个 vCPU 收到处理中断后，将该中断应交给另一 vCPU 处理的中断。

本发明实施例中，vCPU 之间的中断具体可以是指 IPI。vCPU 具体可以是 vCPU，即 vCPU 处理的进程或者中断都可以是等同于该虚拟机处理的进程或者中断。例如：vCPU 中断就可以理解该 vCPU 所在的虚拟机中断。另外，在 SMP 中每个虚拟机都可以配置对应的 vAPIC，例如，vLocal APIC。

图 2 是本发明实施例提供的一种 vCPU 之间的中断的实现方法的流程示意图，如图 2 所示，包括：

201、当源 vCPU 需要向目的 vCPU 触发中断时，将包含所述目的 vCPU 的信息和用于表示源 vCPU 向目的 vCPU 触发中断的指示数据的寄存数据写入源 vCPU 的 vAPIC 的一个虚拟寄存器，以使虚拟机监控器从所述虚拟寄存器解析出所述目的 vCPU 的信息和所述指示数据，并由所述虚拟机监控器根据所述目的 vCPU 的信息和指示数据将 vCPU 之间的中断注入到所述目的 vCPU。

可选的，步骤 201 具体可以是当源 vCPU 的负载过重，需要将某一些负载均衡调度至目的 vCPU 时，或者当源 vCPU 接收到某一中断时，但该源 vCPU 不想执行该中断，而是触发目的 vCPU 中断。或者接收到用户输入的命令指示源 vCPU 需要向目的 vCPU 触发中断时，或者接收 VMM 发送的命令指示源 vCPU 需要向目的 vCPU 触发中断时。本发明实施中对此不作限定。

其中，上述寄存数据包具体可以是用于写入 vAPIC 的一个虚拟寄存器的数据，该寄存数据的格式与该虚拟寄存器的格式对应。当该寄存数据写入 vAPIC 的虚拟寄存器时，该虚拟寄存器就存储该寄存数据，以便 VMM 读取该寄存数据，执行相关的操作。

可选的，上述目的 vCPU 的信息具体可以是用于唯一表示目的 vCPU 的信息，具体可以是用于标识所述目的 vCPU 的标识数据，或者目的 vCPU 的信息（例如：目标 vCPU 的名称和标识等信息）。另外，上述目的 vCPU 的信息具体可以占用上述寄存数据的一个或者多个数据位，即该信息写入上述虚拟寄存器中时，该信息占用虚拟寄存器的一个或者多个数据位。上述指示数据具体可以是占用上述寄存数据一个多个数据位，即该指示数据写入上述虚拟寄存器中时，该指示数据占用虚拟寄存器的一个或者多个数据位。

可选的，上述 VMM 将 vCPU 之间的中断注入到所述目的 vCPU 具体可以是 VMM 向目的 vCPU 发送用于产生 vCPU 之间的中断的命令；或者向 VMM 向目的 vAPIC 发送用于产生 vCPU 之间的中断的命令，由该目的 vAPIC 控制目的 vCPU 产生 vCPU 之间的中断；或者是 VMM 控制目的 vCPU 产生 vCPU 之间的中断。

可选的，上述目的 vCPU 具体可以是指一个或者多个 vCPU，上述寄存数据包填入的信息具体还可以是一个或者多个信息，即一个信息对应一个 vCPU。当然还可以用一个信息标识一个或者多个 vCPU。例如，上述指示数据包括用于表示信息的类型的目的模型数据，该目的模型数据具体可以是一位二进制数，例如：该目的模型数据为 1 时表示上述目的 vCPU 的信息用于标识多个 vCPU，当该目的模型数据为 1 时表示上述目的 vCPU 的信息用于标识一个 vCPU，具体可以如下：

假设 SMP 包括 8 个 vCPU，那么上述目的 vCPU 的信息就可以为 8 位二进制数据，且预先设置好这 8 位二进制数与上述 8 个 vCPU 的映射关系，例如，第 1 位二进制数据与第 1 个 vCPU 对应，第 2 位二进制数据与第 2 个 vCPU 对应等。

这样当上述目的 vCPU 的信息为 11000000 时，且目的模型数据为 1 时，该信息就用于表示第 1 和第 2 个 vCPU。

假设 SMP 包括 8 个 vCPU，那么上述目的 vCPU 的信息可以为 8 位二进制数据，而 8 位二进制数据可以表示 256 个值，而 SMP 包括 8 个 vCPU，这样就可以预先选择上述 256 个值中的 8 个值与上述 8 个 vCPU 的映射关系，选择最小的 8 个值 0-7，这样就可以将 8 位二进制数据的值为 7 与第 1 个 vCPU

对应，8位二进制数据的值为6与第2个vCPU对应等。

这样当上述目的vCPU的信息为00000110，即3位二进制数据的值为6时，且目的模型数据为0时，该信息就用于表示第2个vCPU。

当还可以将上述目的vCPU的信息还可以为3位二进制数据，而3位二进制数据可以表示8个值，即可以预先设置好这8值与上述8个vCPU的映射关系，例如，3位二进制数据的值为7与第1个vCPU对应，3位二进制数据的值为6与第2个vCPU对应等。

这样当上述目的vCPU的信息为110，即3位二进制数据的值为6时，且目的模型数据为0时，该信息就用于表示第2个vCPU。

在另一个实施例中，上述虚拟寄存器为虚拟中断命令寄存器vICR的高位寄存器或低位寄存器；步骤201具体可以包括:

当源vCPU需要向目的vCPU触发中断时，将包含所述目的vCPU的信息和用于表示源vCPU向目的vCPU触发中断的指示数据和优化数据的寄存数据写入源vCPU的vAPIC的一个虚拟寄存器，其中，所述优化数据被填入在所述虚拟寄存器的保留位，所述优化数据用于指示所述VMM从所述虚拟寄存器中解析出所述目的vCPU的信息和所述指示数据。

其中，上述优化数据具体可以是一个一位或者多个位的二进制数，例如，上述优化数据为“1”表示所述VMM从所述虚拟寄存器中解析出所述目的vCPU的信息和所述指示数据。

该实施例中，由于增加了上述优化数据，这样可以实现当上述寄存数据包含该优化数据时，VMM实现中断的方式就按照本方法实现中断的方式进行，当上述寄存数据不包含该优化数据时，VMM实现中断的方式就可以采用现有技术的方式实现中断，从而保证本实施例中寄存数据或者上述虚拟寄存器可以兼容不同的VMM。

在另一个实施例中，步骤201具体可以包括:

当源vCPU需要向目的vCPU触发中断时，将包含所述目的vCPU的信息和用于表示源vCPU向目的vCPU触发中断的指示数据的寄存数据写入源vCPU的vAPIC的一个虚拟寄存器，其中，所述目的vCPU的信息被填入在所述虚拟寄存器的保留位，所述指示数据包括被填入在所述虚拟寄存器的航向

域、传送模域、目的模型域、等级域、触发模型域和目的速记域的数据。

其中，上述指示数据包括被填入在所述虚拟寄存器的航向域、传送模域、目的模型域、等级域、触发模型域和目的速记域的数据具体可以是指，该指示数据是用于填入在所述虚拟寄存器的航向域、传送模域、目的模型域、等级域、触发模型域和目的速记域的数据。

可选的，上述寄存数据包含上述目的 vCPU 的信息和指示数据具体可以是在执行步骤 201 之前，将信息和指示数据填入至寄存数据中，或者还可以是在执行在步骤 201 之前将信息填入至寄存数据，当在执行步骤 201 中将指示数据填入至该寄存数据中，或者在执行步骤 201 中将信息和指示数据填入至该寄存数据中。本实施例对此不作限定。

可选的，上述方法具体可以是应用于虚拟机，通过步骤 201 将寄存数据写入 vAPIC 的一个虚拟寄存器，从而产生 VM-Exit，即虚拟机进行一次虚拟机与 VMM 之间的上下文切换。当上述 VMM 从所述虚拟寄存器解析出所述目的 vCPU 的信息和所述指示数据后，就可以由该 VMM 根据所述目的 vCPU 的信息和指示数据将 vCPU 之间的中断注入到所述目的 vCPU。从而在实现中断过程中上述虚拟机只产生了一次 VM-Exit，从而可以提高虚拟机的性能。

可选的，上述 vAPIC 具体可以是上述虚拟机的 vLocal APIC。

上述技术方案中，当源 vCPU 需要向目的 vCPU 触发中断时，将包含所述目的 vCPU 的信息和用于表示源 vCPU 向目的 vCPU 触发中断的指示数据的寄存数据写入源 vCPU 的虚拟高级可编程中断控制器 vAPIC 的一个虚拟寄存器，以使虚拟机监控器从所述虚拟寄存器解析出所述目的 vCPU 的信息和所述指示数据，并由所述虚拟机监控器根据所述目的 vCPU 的信息和指示数据将 vCPU 之间的中断注入到所述目的 vCPU。这样一次中断只需要执行一次对 vAPIC 的操作，即一次中断只导致一次 VM-Exit，从而可以提高虚拟机的性能。

图 3 是本发明实施例提供的另一种 vCPU 之间的中断的实现方法的流程示意图，如图 3 所示，包括：

301、当源 vCPU 需要向目的 vCPU 触发中断时，将包含所述目的 vCPU 的信息和用于表示源 vCPU 向目的 vCPU 触发中断的指示数据的寄存数据写入

源 vCPU 的 vAPIC 的所述 vICR 的高位寄存器或者低位寄存器，以使虚拟机监控器从高位寄存器或者低位寄存器解析出所述目的 vCPU 的信息和所述指示数据，并由所述虚拟机监控器根据所述目的 vCPU 的信息和指示数据将 vCPU 之间的中断注入到所述目的 vCPU。

这样可以实现只需要一次操作就可以将上述目的 vCPU 的信息和指示数据写入 vICR 的高位寄存器或者低位寄存器，从而实现一次中断过程中只产生一次 VM-Exit。

可选的，上述 vICR 具体可以为 64 位的虚拟寄存器，且上述 vICR 中的 32 位为上述高位寄存器，低 32 位为上述低位寄存器，即上述寄存数据为与高位寄存器对应的 32 位寄存数据。如图 4 所示的 vICR 的高位寄存器中的 56-63 位数据为目的域，32-55 位数据为保留位； vICR 的低位寄存器中第 0-7 位为航向（Vector）域，第 8-10 位为传送模型（Delivery Mode）域，第 11 位为目的模型（Destination Mode）域，第 12 位为传送状态（Delivery Status）域，第 13 位为保留位，第 14 位为等级（Level）域，第 15 位触发模型（Trigger Mode）域，第 16 和 17 位为保留位，第 18 和 19 位为目的速记（Destination Shorthand）域，第 20-31 位数据为保留位。图 5 所示的 vICR 的高位寄存器中的 32-63 位数据为目的域； vICR 的低位寄存器中第 0-7 位为航向（Vector）域，第 8-10 位为传送模型（Delivery Mode）域，第 11 位为目的模型（Destination Mode）域，第 12 和 13 位为保留位，第 14 位为等级（Level）域，第 15 位触发模型（Trigger Mode）域，第 16 和 17 位为保留位，第 18 和 19 位为目的速记（Destination Shorthand）域，第 20-31 位数据为保留位。其中，上述高低寄存器和低位寄存器包括各个域都可以是采用现有技术中 vICR 对各域的定义，此处不作详细说明。

可选的，上述将包含所述目的 vCPU 的信息和用于表示源 vCPU 向目的 vCPU 触发中断的指示数据的寄存数据写入源 vCPU 的 vAPIC 的所述 vICR 的高位寄存器或者低位寄存器，具体可以包括：

将包含所述目的 vCPU 的信息和用于表示源 vCPU 向目的 vCPU 触发中断的指示数据的寄存数据写入源 vCPU 的 vAPIC 的所述 vICR 的低位寄存器，其中，所述目的 vCPU 的信息被填入在所述低位寄存器的保留位。

例如，如图 4 所示，上述寄存数据的信息被填入在低位寄存器的第 20-31 位、第 16-17 位或者第 13 位的保留位中，而上述指示数据被填入在低位寄存器的第 0-7 位为航向（Vector）域、第 8-10 位为传送模型（Delivery Mode）域、第 11 位为目的模型（Destination Mode）域、第 12 位为传送状态（Delivery Status）域、第 14 位为等级（Level）域、第 15 位触发模型（Trigger Mode）域、第 18 和 19 位为目的速记（Destination Shorthand）域中，即上述指示数据可以是包括被填入在上述低位寄存器的航向（Vector）域、传送模型（Delivery Mode）域、目的模型（Destination Mode）域、传送状态（Delivery Status）域、等级（Level）域、触发模型（Trigger Mode）域和目的速记（Destination Shorthand）域的数据。

例如，如图 5 所示，上述寄存数据的信息被填入在低位寄存器的第 20-31 位、第 16 和 7 位或者第 12-13 位的保留位中，而上述指示数据被填入在低位寄存器的第 0-7 位为航向（Vector）域、第 8-10 位为传送模型（Delivery Mode）域、第 11 位为目的模型（Destination Mode）域、第 14 位为等级（Level）域、第 15 位触发模型（Trigger Mode）域、第 18 和 19 位为目的速记（Destination Shorthand）域中，即上述指示数据可以是包括被填入在上述低位寄存器的航向（Vector）域、传送模型（Delivery Mode）域、目的模型（Destination Mode）域、等级（Level）域、触发模型（Trigger Mode）域和目的速记（Destination Shorthand）域的数据。

可选的，上述将包含所述目的 vCPU 的信息和用于表示源 vCPU 向目的 vCPU 触发中断的指示数据的寄存数据写入源 vCPU 的 vAPIC 的所述 vICR 的高位寄存器或者低位寄存器，具体可以包括：

将包含所述目的 vCPU 的信息和用于表示源 vCPU 向目的 vCPU 触发中断的指示数据的寄存数据写入源 vCPU 的 vAPIC 的所述 vICR 的高位寄存器，其中，所述指示数据被填入在所述高位寄存器的保留位。

例如，如图 4 所示，上述寄存数据的信息被填入在高位寄存器的第 56-63 位的目的域，上述指示数据被填入在高位寄存器的第 32-55 位的保留位。

可选的，步骤 301 具体可以包括：

当源 vCPU 需要向目的 vCPU 触发中断时，将包含所述目的 vCPU 的信息

和用于表示源 vCPU向目的 vCPU触发中断的指示数据和优化数据的寄存数据写入源 vCPU 的 vAPIC 的所述 vICR 的高位寄存器或者低位寄存器，其中，所述优化数据写入所述高位寄存器或者低位寄存器的保留位，所述优化数据用于指示所述VMM从所述虚拟寄存器中解析出所述目的vCPU的信息和所述指示数据。

这样可以实现上述寄存数据和虚拟寄存器兼容不同的虚拟器监控器。

可选的，通过上述分析可以得出图 5 所示的寄存数据包比图 4 所示的寄存数据包多了一位保留位，即第 12 位，而图 4 所示的寄存数据包的第 12 位为传送状态（Delivery Status）域。这样在采用图 5 所示的寄存数据包时，本实施例中在中断实现过程中，就不需要读取的 vICR 传送状态（Delivery Status），直接执行步骤 301。当在采用图 4 所示的寄存数据包时，本实施例中在中断实现过程中，在步骤 301 之前，还可以读取的 vICR 传送状态（Delivery Status），再根据该传送状态执行步骤 301。即在步骤 301 之前，所述方法可以包括:

读取的 vICR 中的传送状态。

步骤 301 具体可以包括:

当源 vCPU 需要向目的 vCPU 触发中断时，再根据该传送状态将包含所述目的 vCPU的信息和用于表示源 vCPU向目的 vCPU触发中断的指示数据的寄存数据写入源 vCPU 的 vAPIC 的所述 vICR 的高位寄存器或者低位寄存器。

需要说明的是，在实际应用中 vICR 的传送状态都为空闲（Idle）状态，所以本实施例中不执行读取的 vICR 中的传送状态也是可以实现的，且可以减少一次 VM-Exit。

上述技术方案中，当源 vCPU 需要向目的 vCPU 触发中断时，将包含所述目的 vCPU的信息和用于表示源 vCPU向目的 vCPU触发中断的指示数据的寄存数据写入源 vCPU 的 vAPIC 的所述 vICR 的高位寄存器或者低位寄存器。这样一次中断只需要执行一次对 vAPIC 的操作，即一次中断只导致一次 VM-Exit，从而可以提高虚拟机的性能。且由于是在 vICR 写入上述寄存数据包，而 vICR 为专门用于控制中断的虚拟寄存器，这样 VMM 可以快速实现将 vCPU之间的中断注入到所述目的 vCPU。

下面为本发明装置实施例，本发明装置实施例用于执行本发明方法实施例一至二实现的方法，为了便于说明，主要示出了与本发明实施例相关的部分，具体技术细节未揭示的，请参照本发明实施例一和实施例二。

图 6 是本发明实施例提供的一种虚拟机的结构示意图，如图 6 所示，包括：一个或者多个 vCPU61，vCPU 包括 vAPIC611，所述 vAPIC611 包括虚拟寄存器 6111，其中：

所述一个或者多个 vCPU61 中的当前 vCPU61 用于：

当所述当前 vCPU61 需要向目的 vCPU 触发中断时，将包含所述目的 vCPU 的信息和用于表示所述当前 vCPU61 向所述目的 vCPU 触发中断的指示数据的寄存数据包写入所述当前 vCPU 的 vAPIC611 的一个虚拟寄存器 6111，以使虚拟机监控器从该虚拟寄存器 6111 解析出所述目的 vCPU 的信息和所述指示数据，并由所述虚拟机监控器根据所述目的 vCPU 的信息和指示数据将 vCPU 之间的中断注入所述目的 vCPU。

可选的，vCPU61 将寄存数据写入 vAPIC611 的虚拟寄存器 6111，从而产生 VM-Exit，即虚拟机进行一次虚拟机与 VMM（例如：Hypervisor）之间的上下文切换。当上述 VMM 从所述虚拟寄存器 6111 解析出所述目的 vCPU 的信息和所述指示数据后，就可以由该 VMM 根据所述目的 vCPU 的信息和指示数据将 vCPU 之间的中断注入到所述目的 vCPU。从而在实现中断过程中上述虚拟机只产生了一次 VM-Exit，从而可以提高虚拟机的性能。

可选的，上述包含所述目的 vCPU 的信息和用于表示所述当前 vCPU61 向所述目的 vCPU 触发中断的指示数据的寄存数据包写入所述当前 vCPU 的 vAPIC611 的一个虚拟寄存器 6111 的操作上，所述当前 vCPU 具体用于：

将包含所述目的 vCPU 的信息和用于表示所述当前 vCPU61 向目的 vCPU 触发中断的指示数据和优化数据的寄存数据写入所述当前 vCPU61 的 vAPIC611 的一个虚拟寄存器 6111，其中，所述优化数据被填入在该虚拟寄存器 6111 的保留位，所述优化数据用于指示所述 VMM 从该虚拟寄存器中解析出所述目的 vCPU 的信息和所述指示数据。

这样通过上述优化数据可以提高上述寄存数据或者提高上述虚拟寄存器

的兼容性。

可选的，上述包含所述目的 vCPU 的信息和用于表示所述当前 vCPU61 向所述目的 vCPU 触发中断的指示数据的寄存数据包写入所述当前 vCPU 的 vAPIC611 的一个虚拟寄存器 6111 的操作上，所述当前 vCPU 具体用于：

将包含所述目的 vCPU 的信息和用于表示所述当前 vCPU61 向目的 vCPU 触发中断的指示数据的寄存数据写入所述当前 vCPU61 的 vAPIC611 的一个虚拟寄存器 611，其中，所述目的 vCPU 的信息被填入在该虚拟寄存器 6111 的保留位，所述指示数据包括被填入在该虚拟寄存器 6111 的航向域、传送模域、目的模型域、等级域、触发模型域和目的速记域的数据。

可选的，上述 vAPIC611 具体可以是上述虚拟机的 vLocal vAPIC。

可选的，所述虚拟机上还运行有操作子系统（Guest OS）62 和应用程序 63，其中：

操作子系统（Guest OS）62，用于运行应用程序 63;

应用程序 63，用于执行虚拟机中业务。

可选的，上述目的 vCPU 可以是所述虚拟机的 vCPU，也可以是另一个虚拟机的 vCPU。

上述技术方案中，当所述当前 vCPU 需要向目的 vCPU 触发中断时，包含所述目的 vCPU 的信息和用于表示所述当前 vCPU 向所述目的 vCPU 触发中断的指示数据的寄存数据包写入所述当前 vCPU 的 vAPIC 的一个虚拟寄存器，以使虚拟机监控器从该虚拟寄存器解析出所述目的 vCPU 的信息和所述指示数据，并由所述虚拟机监控器根据所述目的 vCPU 的信息和指示数据将 vCPU 之间的中断注入到所述目的 vCPU。这样一次中断只需要执行一次对 vAPIC 的操作，即一次中断只导致一次 VM-Exit，从而可以提高虚拟机的性能。

图 7 是本发明实施例提供的另一种虚拟机的结构示意图，如图 7 所示，包括：一个或者多个 vCPU71，vVPU71 包括 vAPIC711，所述 vAPIC711 包括 vICR7111，其中：

一个或者多个 vCPU71 中的当前 vCPU71 用于：

当所述当前 vCPU71 需要向目的 vCPU 触发中断时，将包含所述目的 vCPU

的信息和用于表示所述当前 vCPU71 向目的 vCPU触发中断的指示数据的寄存数据写入所述当前 vCPU71 的 vAPIC711 的 vICR7111 的高位寄存器或者低位寄存器，以使 VMM 从所述高位寄存器或者低位寄存器解析出所述目的 vCPU 的信息和所述指示数据，并由所述 VMM 根据所述目的 vCPU 的信息和指示数据将 vCPU 之间的中断注入到所述目的 vCPU。

可选的，上述 vICR7111 包括的高位寄存器和低位寄存器的结构可以参考图 4 和图 5 所示的寄存器结构，此处不作重复说明。

可选的，上述将包含所述目的 vCPU 的信息和用于表示所述当前 vCPU71 向目的 vCPU 触发中断的指示数据的寄存数据写入所述当前 vCPU71 的 vAPIC711 的 vICR7111 的高位寄存器或者低位寄存器的操作上，所述当前 vCPU71 具体可以用于:

将包含所述目的 vCPU 的信息和用于表示当前 vCPU71 向目的 vCPU 触发中断的指示数据的寄存数据写入当前 vCPU71 的 vAPIC711 的所述 vICR7111 的低位寄存器，其中，所述目的 vCPU 的信息被填入在所述低位寄存器的保留位。

例如，如图 4 所示，上述寄存数据的信息写入低位寄存器的第 20-31 位、第 16-17 位或者第 13 位的保留位中，而上述指示数据写入低位寄存器的第 0-7 位为航向（Vector）域、第 8-10 位为传送模型（Delivery Mode）域、第 11 位为目的模型（Destination Mode）域、第 12 位为传送状态（Delivery Status）域、第 14 位为等级（Level）域、第 15 位触发模型（Trigger Mode）域、第 18 和 19 位为目的速记（Destination Shorthand）域中，即上述指示数据可以是包括写入上述低位寄存器的航向（Vector）域、传送模型（Delivery Mode）域、目的模型（Destination Mode）域、传送状态（Delivery Status）域、等级（Level）域、触发模型（Trigger Mode）域和目的速记（Destination Shorthand）域的数据。

例如，如图 5 所示，上述寄存数据的信息写入低位寄存器的第 20-31 位、第 16 和 7 位或者第 12-13 位的保留位中，而上述指示数据写入低位寄存器的第 0-7 位为航向（Vector）域、第 8-10 位为传送模型（Delivery Mode）域、第 11 位为目的模型（Destination Mode）域、第 14 位为等级（Level）域、第 15

位触发模型(Trigger Mode)域、第18和19位为目的速记(Destination Shorthand)域中，即上述指示数据可以是包括写入上述低位寄存器的航向（Vector）域、传送模型（Delivery Mode）域、目的模型（Destination Mode）域、等级（Level）域、触发模型（Trigger Mode）域和目的速记（Destination Shorthand）域的数据。

可选的，上述将包含所述目的 vCPU 的信息和用于表示所述当前 vCPU71 向目的 vCPU 触发中断的指示数据的寄存数据写入所述当前 vCPU71 的 vAPIC711 的 vICR7111 的高位寄存器或者低位寄存器的操作上，所述当前 vCPU71 具体可以用于:

将包含所述目的 vCPU 的信息和用于表示所述当前 vCPU71 向目的 vCPU 触发中断的指示数据的寄存数据写入所述当前 vCPU71 的 vAPIC711 的 vICR7111 的高位寄存器，其中，所述指示数据被填入在所述高位寄存器的保留位。

例如，如图 4 所示，上述寄存数据包含的信息被填入在高位寄存器的第 56-63 位的目的域，上述指示数据被填入高位寄存器的第 32-55 位的保留位。

可选的，上述将包含所述目的 vCPU 的信息和用于表示所述当前 vCPU71 向目的 vCPU 触发中断的指示数据的寄存数据写入所述当前 vCPU71 的 vAPIC711 的 vICR7111 的高位寄存器或者低位寄存器的操作上，所述当前 vCPU71 具体可以用于:

将包含所述目的 vCPU 的信息和用于表示所述当前 vCPU71 向目的 vCPU 触发中断的指示数据和优化数据的寄存数据写入所述当前 vCPU71 的 vAPIC711 的 vICR7111 的高位寄存器或者低位寄存器，其中，所述优化数据被填入在该高位寄存器或者低位寄存器的保留位，所述优化数据用于指示所述 VMM 从该虚拟寄存器中解析出所述目的 vCPU 的信息和所述指示数据。

这样可以实现上述寄存数据兼容不同的虚拟器监控器。

可选的，当在采用图 4 所示的寄存数据包时，本实施例中在中断实现过程中，当前 vCPU71 在执行将包含所述目的 vCPU 的信息和用于表示所述当前 vCPU71 向目的 vCPU 触发中断的指示数据的寄存数据写入所述当前 vCPU71 的 vAPIC711 的 vICR7111 的高位寄存器或者低位寄存器的操作之前，还可以

用于:

读取的 vICR 中的传送状态。

上述将包含所述目的 vCPU 的信息和用于表示所述当前 vCPU71 向目的 vCPU 触发中断的指示数据的寄存数据写入所述当前 vCPU71 的 vAPIC711 的 vICR7111 的高位寄存器或者低位寄存器的操作上，所述当前 vCPU71 具体可以用于:

再根据该传送状态将包含所述目的 vCPU 的信息和用于表示所述当前 vCPU71 向目的 vCPU 触发中断的指示数据的寄存数据写入所述当前 vCPU71 的 vAPIC711 的 vICR7111 的高位寄存器或者低位寄存器。

需要说明的是，在实际应用中 vICR 的传送状态都为空闲（Idle）状态，所以本实施例中不执行读取的 vICR 中的传送状态也是可以实现的，且可以减少一次 VM-Exit。

可选的，所述虚拟机上还运行有操作子系统（Guest OS）72 和应用程序 73，其中:

操作子系统（Guest OS）72，用于运行应用程序 73;

应用程序 73，用于执行虚拟机中业务。

上述技术方案中，当所述当前 vCPU 需要向目的 vCPU 触发中断时，将包含所述目的 vCPU 的信息和用于表示所述当前 vCPU 向目的 vCPU 触发中断的指示数据的寄存数据写入所述当前 vCPU 的 vAPIC 的 vICR 的高位寄存器或者低位寄存器，以使 VMM 从所述高位寄存器或者低位寄存器解析出所述目的 vCPU 的信息和所述指示数据，并由所述 VMM 根据所述目的 vCPU 的信息和指示数据将 vCPU 之间的中断注入到所述目的 vCPU。这样一次中断只需要执行一次对 vAPIC 的操作，即一次中断只导致一次 VM-Exit，从而可以提高虚拟机的性能，且由于是在 vICR 写入上述寄存数据包，而 vICR 为专门用于控制中断的虚拟寄存器，这样 VMM 可以快速实现将 vCPU 之间的中断注入到所述目的 vCPU。

图 8 是本发明实施例提供的一种计算节点的结构示意图，如图 8 所示，包括: 硬件层 81、运行在所述硬件层 81 之上的 VMM82、以及运行在所述 VMM

之上的至少一个虚拟机，其中，所述至少一个虚拟机 VM 包括第一虚拟机 83 和第二虚拟机 81，第一虚拟机 83 包括第一 vCPU831、第一 vCPU831 包括第一 vAPIC8311，第一 vAPIC8311 包括第一虚拟寄存器 8311，第二虚拟机 84 包括第二 vCPU841，其中：

硬件层 81，用于提供 VMM82、第一虚拟机 83 和第二虚拟机 84 的虚拟化环境运行的硬件平台。其中，硬件层 81 可包括多种硬件，例如某计算节点的硬件层可包括 CPU 和内存，还可以包括网卡、存储器等等高速/低速输入/输出（I/O，Input/Output）设备，及具有特定处理功能的其它设备，如 IOMMU，其中 IOMMU 可用于虚拟机物理地址和 Host 物理地址的转换。

第一 vCPU831，用于当第一 vCPU831 需要向第二 vCPU 触发中断时，将包含所述第二 vCPU841 的信息和用于表示所述第一 vCPU831 向所述第二 vCPU841 触发中断的指示数据的寄存数据包写入所述第一虚拟寄存器 83111；

所述 VMM82，用于从所述第一虚拟寄存器 83111 解析出所述第二 vCPU 的信息和所述指示数据，并根据所述第二 vCPU 的信息和指示数据将 vCPU 之间的中断注入到所述第二 vCPU841，其中所述 vCPU 之间的中断包括第一 vCPU831 与第二 vCPU841 之间的中断。

可选的，当上述 VMM 上运行还有其它虚拟机（例如：第三虚拟机）时，本实施例还可以实现上述第一 vCPU 与其它虚拟机（例如：第三虚拟机）的 vCPU 之间的中断，具体可以实现过程请参考第一 vCPU 与第二 vCPU 之间的中断实现过程，此作不作重复说明。

可选的，上述第一虚拟寄存器 83111 为 vICR，该 vICR 包括高位寄存器和低位寄存器，其中，高位寄存器和低位寄存器的结构具体可以参考图 4 和图 5。

可选的，所述第一 vCPU831 具体可以用于当第一 vCPU831 需要向第二 vCPU 触发中断时，将包含所述第二 vCPU841 的信息和用于表示所述第一 vCPU831 向所述第二 vCPU841 触发中断的指示数据的寄存数据包写入所述高位寄存器或低位寄存器；

所述 VMM82 具体可以用于从所述高位寄存器或低位寄存器解析出所述第二 vCPU841 的信息和所述指示数据，并根据所述第二 vCPU841 的信息和指示数据将 vCPU 之间的中断注入到所述第二 vCPU841，其中所述 vCPU 之间的

中断包括第一 vCPU831 与第二 vCPU841 之间的中断。

可选的，所述第一 vCPU831 具体用于当第一 vCPU831 需要向第二 vCPU841 触发中断时，将包含所述第二 vCPU841 的信息和用于表示所述第一 vCPU831 向所述第二 vCPU841 触发中断的指示数据的寄存数据包写入所述低位寄存器，其中，所述第二 vCPU841 的信息被填入在所述低位寄存器的保留位;

所述 VMM82 具体用于从所述低位寄存器的保留位解析出所述第二 vCPU841 的信息，以及从所述低位寄存器解析出所述指示数据，并根据所述第二 vCPU841 的信息和指示数据将 vCPU 之间的中断注入到所述第二 vCPU841，其中所述 vCPU 之间的中断包括第一 vCPU831 与第二 vCPU841 之间的中断。

其中，所述指示数据可以包括被填入在所述虚拟寄存器的航向域、传送模域、目的模型域、等级域、触发模型域和目的速记域的数据。

可选的，所述第一 vCPU831 具体用于当第一 vCPU831 需要向第二 vCPU841 触发中断时，将包含所述第二 vCPU841 的信息和用于表示所述第一 vCPU831 向所述第二 vCPU841 触发中断的指示数据的寄存数据包写入所述高位寄存器，其中，所述指示数据被填入在所述高位寄存器的保留位;

所述 VMM82 具体用于从所述高位寄存器的保留位解析出所述指示数据，以及从所述高位寄存器解析出所述第二 vCPU841 的信息，并根据所述第二 vCPU841 的信息和指示数据将 vCPU 之间的中断注入到所述第二 vCPU841，其中所述 vCPU 之间的中断包括第一 vCPU831 与第二 vCPU841 之间的中断。

可选的，所述第一 vCPU831 具体可以用于当第一 vCPU831 需要向第二 vCPU841 触发中断时，将包含所述第二 vCPU841 的信息和用于表示所述第一 vCPU831 向所述第二 vCPU841 触发中断的指示数据和优化数据的寄存数据包写入所述第一虚拟寄存器 83111，其中，所述优化数据被填入在该第一虚拟寄存器 83111 的保留位，所述优化数据用于指示所述 VMM 从所述第一虚拟寄存器 83111 中解析出所述第二 vCPU 的信息和所述指示数据;

所述 VMM 具体可以用于从所述第一虚拟寄存器 83111 解析出所述优化数据，并根据所述优化数据从所述第一虚拟寄存器 83111 解析出所述第二 vCPU

的信息和所述指示数据，并根据所述第二 vCPU 的信息和指示数据将 vCPU 之间的中断注入到所述第二 vCPU841，其中所述 vCPU 之间的中断包括第一 vCPU831 与第二 vCPU841 之间的中断。

可选的，所述第一 vCPU831 具体可以用于当第一 vCPU831 需要向第二 vCPU 触发中断时，将包含所述第二 vCPU841 的信息和用于表示所述第一 vCPU831 向所述第二 vCPU841 触发中断的指示数据的寄存数据包写入所述第一虚拟寄存器 83111，其中，所述第二 vCPU 的信息被填入在所述虚拟寄存器的保留位，所述指示数据包括被填入在所述虚拟寄存器的航向域、传送模域、目的模型域、等级域、触发模型域和目的速记域的数据；

所述 VMM82 具体可以用于从所述第一虚拟寄存器 83111 的保留位解析出所述第二 vCPU 的信息，以及从所述第一虚拟寄存器 83111 的航向域、传送模域、目的模型域、等级域、触发模型域和目的速记域中解析出所述指示数据，并根据所述第二 vCPU 的信息和指示数据将 vCPU 之间的中断注入到所述第二 vCPU841，其中所述 vCPU 之间的中断包括第一 vCPU831 与第二 vCPU841 之间的中断。

上述技术方案中，所述第一 vCPU 当第一 vCPU 需要向第二 vCPU 触发中断时，将包含所述第二 vCPU 的信息和用于表示所述第一 vCPU 向所述第二 vCPU 触发中断的指示数据的寄存数据包写入所述第一虚拟寄存器；所述 VMM 从所述第一虚拟寄存器解析出所述第二 vCPU 的信息和所述指示数据，并根据所述第二 vCPU 的信息和指示数据将 vCPU 之间的中断注入到所述第二 vCPU，其中所述 vCPU 之间的中断包括第一 vCPU 与第二 vCPU 之间的中断。这样一次中断只需要执行一次对 vAPIC 的操作，即一次中断只导致一次 VM-Exit，从而可以提高虚拟机的性能。

图 9 是本发明实施例提供的另一种计算节点的结构示意图，如图 9 所示，该计算节点包括：至少一个处理器 91，例如 CPU，至少一个网络接口 92 或者其他用户接口 93，存储器 95，至少一个通信总线 92。通信总线 92 用于实现这些组件之间的连接通信。该计算节点 90 可选的包含用户接口 93，包括显示器，键盘或者点击设备（例如，鼠标，轨迹球（trackball），触感板或者触感

显示屏)。存储器 95 可能包含高速 RAM 存储器，也可能还包括非不稳定的存储器（non-volatile memory），例如至少一个磁盘存储器。存储器 605 可选的可以包含至少一个位于远离前述处理器 91 的存储装置。

在一些实施方式中，存储器 95 存储了如下的元素，可执行模块或者数据结构，或者他们的子集，或者他们的扩展集:

操作系统 951，包含各种系统程序，用于实现各种基础业务以及处理基于硬件的任务;

应用程序模块 952，包含各种应用程序，用于实现各种应用业务。

应用程序模块 952 中包括但不限于一个或者多个 vCPU61，vCPU61 包括 vAPIC611，所述 vAPIC611 包括虚拟寄存器 6111。

应用程序模块 952 中各模块的具体实现参见图 7和图 8 所示实施例中的相应模块，在此不赘述。

在本发明实施例中，通过调用存储器 95 存储的程序或指令，处理器 91 用于:

当源 vCPU 需要向目的 vCPU 触发中断时，

将包含所述目的 vCPU 的信息和用于表示源 vCPU 向目的 vCPU 触发中断的指示数据的寄存数据写入源 vCPU 的虚拟高级可编程中断控制器 vAPIC 的一个虚拟寄存器，以使虚拟机监控器从所述虚拟寄存器解析出所述目的 vCPU 的信息和所述指示数据，并由所述虚拟机监控器根据所述目的 vCPU 的信息和指示数据将 vCPU 之间的中断注入到所述目的 vCPU。

在上述各个实施例中，进一步地，所述处理器 91 还用于:

当源 vCPU 需要向目的 vCPU 触发中断时，将包含所述目的 vCPU 的信息和用于表示源 vCPU 向目的 vCPU 触发中断的指示数据的寄存数据写入源 vCPU 的 vAPIC 的所述 vICR 的高位寄存器或者低位寄存器，以使 VMM 从所述高位寄存器或者低位寄存器解析出所述目的 vCPU 的信息和所述指示数据，并由所述 VMM 根据所述目的 vCPU 的信息和指示数据将 vCPU 之间的中断注入到所述目的 vCPU。

可选的，处理器 91 执行的将包含所述目的 vCPU 的信息和用于表示源 vCPU 向目的 vCPU 触发中断的指示数据的寄存数据写入源 vCPU 的 vAPIC 的

所述 vICR 的高位寄存器或者低位寄存器的操作，可以包括:

将包含所述目的 vCPU 的信息和用于表示源 vCPU 向目的 vCPU 触发中断的指示数据的寄存数据写入源 vCPU 的 vAPIC 的所述 vICR 的低位寄存器，其中，所述目的 vCPU 的信息被填入在所述低位寄存器的保留位。

可选的，处理器 91 执行的将包含所述目的 vCPU 的信息和用于表示源 vCPU 向目的 vCPU 触发中断的指示数据的寄存数据写入源 vCPU 的 vAPIC 的所述 vICR 的高位寄存器或者低位寄存器的操作，可以包括:

将包含所述目的 vCPU 的信息和用于表示源 vCPU 向目的 vCPU 触发中断的指示数据的寄存数据写入源 vCPU 的 vAPIC 的所述 vICR 的高位寄存器，其中，所述指示数据被填入在所述高位寄存器的保留位。

在上述各个实施例中，进一步地，所述处理器 91 还用于:

当源 vCPU 需要向目的 vCPU 触发中断时，将包含所述目的 vCPU 的信息和用于表示源 vCPU 向目的 vCPU 触发中断的指示数据和优化数据的寄存数据写入源 vCPU 的 vAPIC 的一个虚拟寄存器，其中，所述优化数据被填入在所述虚拟寄存器的保留位，所述优化数据用于指示所述 VMM 从所述虚拟寄存器中解析出所述目的 vCPU 的信息和所述指示数据。以使 VMM 根据所述优化数据从所述虚拟寄存器解析出所述目的 vCPU 的信息和所述指示数据，并由所述 VMM 根据所述目的 vCPU 的信息和指示数据将 vCPU 之间的中断注入到所述目的 vCPU。

在上述各个实施例中，进一步地，所述处理器 91 还用于:

当源 vCPU 需要向目的 vCPU 触发中断时，将包含所述目的 vCPU 的信息和用于表示源 vCPU 向目的 vCPU 触发中断的指示数据的寄存数据写入源 vCPU 的 vAPIC 的一个虚拟寄存器，其中，所述目的 vCPU 的信息被填入在所述虚拟寄存器的保留位，所述指示数据包括被填入在所述虚拟寄存器的航向域、传送模域、目的模型域、等级域、触发模型域和目的速记域的数据。以使 VMM 从所述虚拟寄存器解析出所述目的 vCPU 的信息和所述指示数据，并由所述 VMM 根据所述目的 vCPU 的信息和指示数据将 vCPU 之间的中断注入到所述目的 vCPU。

可见，采用上述方案后，这样一次中断只需要执行一次对 vAPIC 的操作，

即一次中断只导致一次 VM-Exit，从而可以提高虚拟机的性能。且如果将上述寄存数据包写入虚拟寄存器，这样由于中断命令寄存为专门用于控制中断的虚拟寄存器，VMM 可以快速实现将 vCPU 之间的中断注入到所述目的 vCPU。

图 10 是本发明实施例还提供一种计算机系统的结构示意图，如图 10 所示，可包括：至少一个计算节点 1000。

本发明实施例的计算节点的描述可参考前述实施例，这里不再赘述。

综上，本发明实施例中，中断过程只需要执行一次对 vAPIC 的操作，即一次中断只导致一次 VM-Exit，从而可以提高虚拟机的性能。且由于是在 vICR 写入上述寄存数据包，而 vICR 为专门用于控制中断的虚拟寄存器，这样 VMM 可以快速实现将 vCPU 之间的中断注入到所述目的 vCPU。另外，本发明实施例中，是在寄存数据包的保留位中填入信息这样在执行本方法过程中，不会修改寄存数据现有的功能。另外，本发明实施例中在寄存数据增加了优化数据，从而可以提高寄存数据和虚拟寄存器的兼容性。

具体的，为了克服现有技术中，IPI 中断是系统常见中断，IPI 中断处理中的 APIC 虚拟寄存器访问会引起 VM-Exit，从而导致虚拟机性能下降的问题，在一种实现方式下，本发明实施例通过虚拟 APIC 的扩展位，将三次 APIC 访问合并为一次，从而减少 APIC 访问次数，达到优化 IPI 中断性能的效果，从而提升了虚拟机和系统的性能。

应当理解的是，对中断处理过程中，其他类型中断对 APIC 虚拟寄存器多次访问造成的 VM-Exit 场景也可以采用本发明实施例类似方式实现以优化性能。

本领域普通技术人员可以理解实现上述实施例方法中的全部或部分流程，是可以通过计算机程序来指令相关的硬件（例如处理器）来完成，所述的程序可存储于一计算机可读取存储介质中，该程序在执行时，可包括如上述各方法的实施例的流程。其中，所述的存储介质可为磁碟、光盘、只读存储记忆体（Read-Only Memory，ROM）或随机存取存储器（Random Access Memory，简称 RAM）等。

以上所揭露的仅为本发明较佳实施例而已，当然不能以此来限定本发明之

权利范围，因此依本发明权利要求所作的等同变化，仍属本发明所涵盖的范围。

权 利 要 求

1、一种虚拟处理器之间的中断的实现方法，其特征在于，包括:

当源虚拟处理器需要向目的虚拟处理器触发中断时，

将包含所述目的虚拟处理器的信息和用于表示源虚拟处理器向目的虚拟处理器触发中断的指示数据的寄存数据写入源虚拟处理器的虚拟高级可编程中断控制器 vAPIC 的一个虚拟寄存器，以使虚拟机监控器从所述虚拟寄存器解析出所述目的虚拟处理器的信息和所述指示数据，并由所述虚拟机监控器根据所述目的虚拟处理器的信息和指示数据将虚拟处理器之间的中断注入到所述目的虚拟处理器。

2、如权利要求 1 所述的方法，其特征在于，所述虚拟寄存器为虚拟中断命令寄存器 vICR 的高位寄存器或低位寄存器；所述将包含所述目的虚拟处理器的信息和用于表示源虚拟处理器向目的虚拟处理器触发中断的指示数据的寄存数据写入源虚拟处理器的 vAPIC 的一个虚拟寄存器，包括:

将包含所述目的虚拟处理器的信息和用于表示源虚拟处理器向目的虚拟处理器触发中断的指示数据的寄存数据写入源虚拟处理器的 vAPIC 的所述 vICR 的高位寄存器或者低位寄存器。

3、如权利要求 2 所述的方法，其特征在于，所述将包含所述目的虚拟处理器的信息和用于表示源虚拟处理器向目的虚拟处理器触发中断的指示数据的寄存数据写入源虚拟处理器的 vAPIC 的所述 vICR 的高位寄存器或者低位寄存器，包括:

将包含所述目的虚拟处理器的信息和用于表示源虚拟处理器向目的虚拟处理器触发中断的指示数据的寄存数据写入源虚拟处理器的 vAPIC 的所述 vICR 的低位寄存器，其中，所述目的虚拟处理器的信息被填入在所述低位寄存器的保留位；或者

将包含所述目的虚拟处理器的信息和用于表示源虚拟处理器向目的虚拟处理器触发中断的指示数据的寄存数据写入源虚拟处理器的 vAPIC 的所述 vICR 的高位寄存器，其中，所述指示数据被填入在所述高位寄存器的保留位。

4、如权利要求 1 所述的方法，其特征在于，所述将包含所述目的虚拟处理器的信息和用于表示源虚拟处理器向目的虚拟处理器触发中断的指示数据的寄存数据写入源虚拟处理器的 vAPIC 的一个虚拟寄存器，包括:

将包含所述目的虚拟处理器的信息和用于表示源虚拟处理器向目的虚拟处理器触发中断的指示数据和优化数据的寄存数据写入源虚拟处理器的 vAPIC 的一个虚拟寄存器，其中，所述优化数据被填入在所述虚拟寄存器的保留位，所述优化数据用于指示所述虚拟机监控器从所述虚拟寄存器中解析出所述目的虚拟处理器的信息和所述指示数据。

5、如权利要求 1 所述的方法，其特征在于，所述将包含所述目的虚拟处理器的信息和用于表示源虚拟处理器向目的虚拟处理器触发中断的指示数据的寄存数据写入源虚拟处理器的 vAPIC 的一个虚拟寄存器，包括:

将包含所述目的虚拟处理器的信息和用于表示源虚拟处理器向目的虚拟处理器触发中断的指示数据的寄存数据写入源虚拟处理器的 vAPIC 的一个虚拟寄存器，其中，所述目的虚拟处理器的信息被填入在所述虚拟寄存器的保留位，所述指示数据包括被填入在所述虚拟寄存器的航向域、传送模域、目的模型域、等级域、触发模型域和目的速记域的数据。

6、一种虚拟机，其特征在于，包括：一个或多个虚拟处理器，所述虚拟处理器包括虚拟高级可编程中断控制器 vAPIC，所述 vAPIC 包括虚拟寄存器，其中:

所述一个或多个虚拟处理器中当前虚拟处理器用于:

当所述当前虚拟处理器需要向目的虚拟处理器触发中断时，将包含所述目的虚拟处理器的信息和用于表示所述当前虚拟处理器向所述目的虚拟处理器触发中断的指示数据的寄存数据包写入所述当前虚拟处理器的 vAPIC 的一个虚拟寄存器，以使虚拟机监控器从该虚拟寄存器解析出所述目的虚拟处理器的信息和所述指示数据，并由所述虚拟机监控器根据所述目的虚拟处理器的信息和指示数据将虚拟处理器之间的中断注入所述目的虚拟处理器。

7、如权利要求 6 所述的虚拟机，其特征在于，所述虚拟寄存器为虚拟中断命令寄存器 vICR 的高位寄存器或低位寄存器；所述包含所述目的虚拟处理器的信息和用于表示所述当前虚拟处理器向所述目的虚拟处理器触发中断的指示数据的寄存数据包写入所述当前虚拟处理器的 vAPIC 的一个虚拟寄存器的操作上，所述当前虚拟处理器具体用于：

将包含所述目的虚拟处理器的信息和用于表示所述当前虚拟处理器向目的虚拟处理器触发中断的指示数据的寄存数据写入所述当前虚拟处理器的 vAPIC 的 vICR 的高位寄存器或者低位寄存器。

8、如权利要求 7 所述的虚拟机，其特征在于，所述将包含所述目的虚拟处理器的信息和用于表示所述当前虚拟处理器向目的虚拟处理器触发中断的指示数据的寄存数据写入所述当前虚拟处理器的 vAPIC 的 vICR 的高位寄存器或者低位寄存器的操作上，所述虚拟处理器具体用于：

将包含所述目的虚拟处理器的信息和用于表示所述当前虚拟处理器向目的虚拟处理器触发中断的指示数据的寄存数据写入所述当前虚拟处理器的 vAPIC 的 vICR 的低位寄存器，其中，所述目的虚拟处理器的信息被填入在所述低位寄存器的保留位；或者

将包含所述目的虚拟处理器的信息和用于表示所述当前虚拟处理器向目的虚拟处理器触发中断的指示数据的寄存数据写入所述当前虚拟处理器的 vAPIC 的 vICR 的高位寄存器，其中，所述指示数据被填入在所述高位寄存器的保留位。

9、如权利要求 6 所述的虚拟机，其特征在于，所述包含所述目的虚拟处理器的信息和用于表示所述当前虚拟处理器向所述目的虚拟处理器触发中断的指示数据的寄存数据包写入所述当前虚拟处理器的 vAPIC 的一个虚拟寄存器的操作上，所述当前虚拟处理器具体用于：

将包含所述目的虚拟处理器的信息和用于表示所述当前虚拟处理器向目的虚拟处理器触发中断的指示数据和优化数据的寄存数据写入所述当前虚拟

处理器的 vAPIC 的一个虚拟寄存器，其中，所述优化数据被填入在该虚拟寄存器的保留位，所述优化数据用于指示所述虚拟机监控器从该虚拟寄存器中解析出所述目的虚拟处理器的信息和所述指示数据。

10、如权利要求 6 所述的虚拟机，其特征在于，所述包含所述目的虚拟处理器的信息和用于表示所述当前虚拟处理器向所述目的虚拟处理器触发中断的指示数据的寄存数据包写入所述当前虚拟处理器的 vAPIC 的一个虚拟寄存器的操作上，所述当前虚拟处理器具体用于：

将包含所述目的虚拟处理器的信息和用于表示所述当前虚拟处理器向目的虚拟处理器触发中断的指示数据的寄存数据写入所述当前虚拟处理器的 vAPIC 的一个虚拟寄存器，其中，所述目的虚拟处理器的信息被填入在该虚拟寄存器的保留位，所述指示数据包括被填入在该虚拟寄存器的航向域、传送模域、目的模型域、等级域、触发模型域和目的速记域的数据。

11、一种计算节点，其特征在于，包括：硬件层、运行在所述硬件层之上的虚拟机监控器、以及运行在所述虚拟机监控器之上的至少一个虚拟机，其中，所述至少一个虚拟机包括第一虚拟机和第二虚拟机，第一虚拟机包括第一虚拟处理器，所述第一虚拟处理器包括第一虚拟高级可编程中断控制器 vAPIC，所述第一 vAPIC 包括第一虚拟寄存器，第二虚拟机包括第二虚拟处理器，其中：

所述第一虚拟处理器，用于当第一虚拟处理器需要向第二虚拟处理器触发中断时，将包含所述第二虚拟处理器的信息和用于表示所述第一虚拟处理器向所述第二虚拟处理器触发中断的指示数据的寄存数据包写入所述第一虚拟寄存器；

所述虚拟机监控器，用于从所述第一虚拟寄存器解析出所述第二虚拟处理器的信息和所述指示数据，并根据所述第二虚拟处理器的信息和指示数据将虚拟处理器之间的中断注入到所述第二虚拟处理器，其中所述虚拟处理器之间的中断包括第一虚拟处理器与第二虚拟处理器之间的中断。

12、如权利要求 11 所述的计算节点，其特征在于，所述第一虚拟寄存器

为为虚拟中断命令寄存器 vICR 的高位寄存器或低位寄存器;

所述第一虚拟处理器具体用于当第一虚拟处理器需要向第二虚拟处理器触发中断时,将包含所述第二虚拟处理器的信息和用于表示所述第一虚拟处理器向所述第二虚拟处理器触发中断的指示数据的寄存数据包写入所述高位寄存器或低位寄存器;

所述虚拟机监控器具体用于从所述高位寄存器或低位寄存器解析出所述第二虚拟处理器的信息和所述指示数据,并根据所述第二虚拟处理器的信息和指示数据将虚拟处理器之间的中断注入到所述第二虚拟处理器,其中所述虚拟处理器之间的中断包括第一虚拟处理器与第二虚拟处理器之间的中断。

13、如权利要求 12 所述的计算节点,其特征在于,

所述第一虚拟处理器具体用于当第一虚拟处理器需要向第二虚拟处理器触发中断时,将包含所述第二虚拟处理器的信息和用于表示所述第一虚拟处理器向所述第二虚拟处理器触发中断的指示数据的寄存数据包写入所述低位寄存器,其中,所述第二虚拟处理器的信息被填入在所述低位寄存器的保留位;

所述虚拟机监控器具体用于从所述低位寄存器的保留位解析出所述第二虚拟处理器的信息,以及从所述低位寄存器解析出所述指示数据,并根据所述第二虚拟处理器的信息和指示数据将虚拟处理器之间的中断注入到所述第二虚拟处理器,其中所述虚拟处理器之间的中断包括第一虚拟处理器与第二虚拟处理器之间的中断;或者

所述第一虚拟处理器具体用于当第一虚拟处理器需要向第二虚拟处理器触发中断时,将包含所述第二虚拟处理器的信息和用于表示所述第一虚拟处理器向所述第二虚拟处理器触发中断的指示数据的寄存数据包写入所述高位寄存器,其中,所述指示数据被填入在所述高位寄存器的保留位;

所述虚拟机监控器具体用于从所述高位寄存器的保留位解析出所述指示数据,以及从所述高位寄存器解析出所述第二虚拟处理器的信息,并根据所述第二虚拟处理器的信息和指示数据将虚拟处理器之间的中断注入到所述第二虚拟处理器,其中所述虚拟处理器之间的中断包括第一虚拟处理器与第二虚拟处理器之间的中断。

14、如权利要求 11 所述的计算节点，其特征在于，

所述第一虚拟处理器具体用于当第一虚拟处理器需要向第二虚拟处理器触发中断时，将包含所述第二虚拟处理器的信息和用于表示所述第一虚拟处理器向所述第二虚拟处理器触发中断的指示数据和优化数据的寄存数据包写入所述第一虚拟寄存器，其中，所述优化数据被填入在该虚拟寄存器的保留位，所述优化数据用于指示所述虚拟机监控器从所述第一虚拟寄存器中解析出所述第二虚拟处理器的信息和所述指示数据；

所述虚拟机监控器具体用于从所述第一虚拟寄存器解析出所述优化数据，并根据所述优化数据从所述第一虚拟寄存器解析出所述第二虚拟处理器的信息和所述指示数据，并根据所述第二虚拟处理器的信息和指示数据将虚拟处理器之间的中断注入到所述第二虚拟处理器，其中所述虚拟处理器之间的中断包括第一虚拟处理器与第二虚拟处理器之间的中断。

15、如权利要求 11 所述的计算节点，其特征在于，

所述第一虚拟处理器具体用于当第一虚拟处理器需要向第二虚拟处理器触发中断时，将包含所述第二虚拟处理器的信息和用于表示所述第一虚拟处理器向所述第二虚拟处理器触发中断的指示数据的寄存数据包写入所述第一虚拟寄存器，其中，所述第二虚拟处理器的信息被填入在所述虚拟寄存器的保留位，所述指示数据包括被填入在所述虚拟寄存器的航向域、传送模域、目的模型域、等级域、触发模型域和目的速记域的数据；

所述虚拟机监控器具体用于从所述第一虚拟寄存器的保留位解析出所述第二虚拟处理器的信息，以及从所述第一虚拟寄存器的航向域、传送模域、目的模型域、等级域、触发模型域和目的速记域中解析出所述指示数据，并根据所述第二虚拟处理器的信息和指示数据将虚拟处理器之间的中断注入到所述第二虚拟处理器，其中所述虚拟处理器之间的中断包括第一虚拟处理器与第二虚拟处理器之间的中断。

16、一种计算机系统，其特征在于，包括：至少一个如权利要求 11-15 任

一项所述的计算节点。

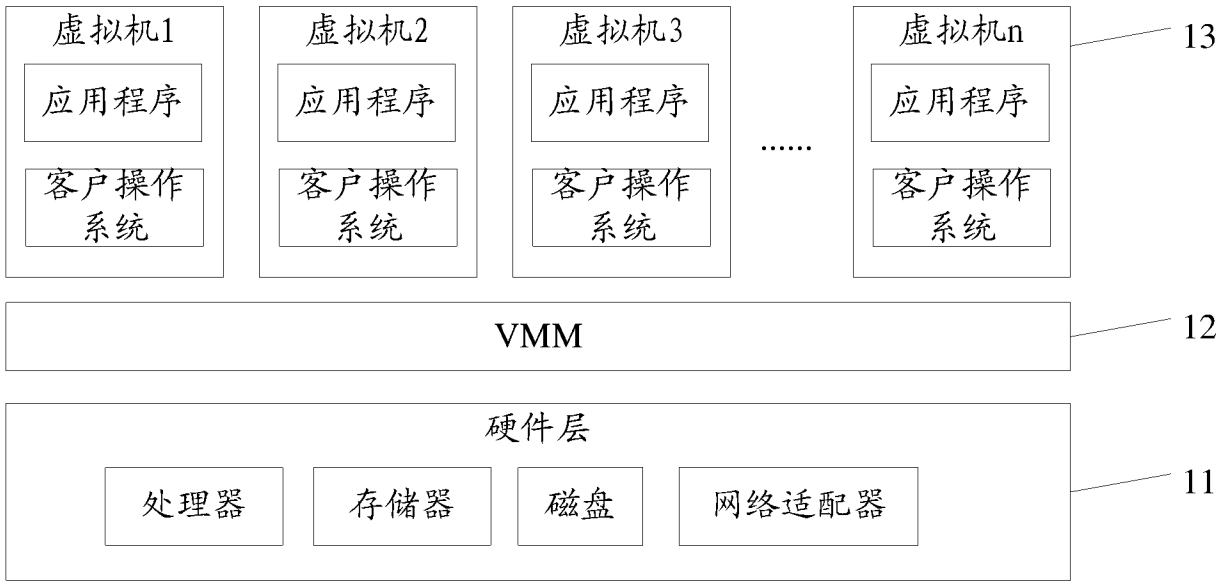

虚拟机1
应用程序
客户操作系统
虚拟机2
应用程序
客户操作系统
虚拟机3
应用程序
客户操作系统
.....
虚拟机n
应用程序
客户操作系统
13
VMM
12
硬件层
处理器
存储器
磁盘
网络适配器
11

图 1

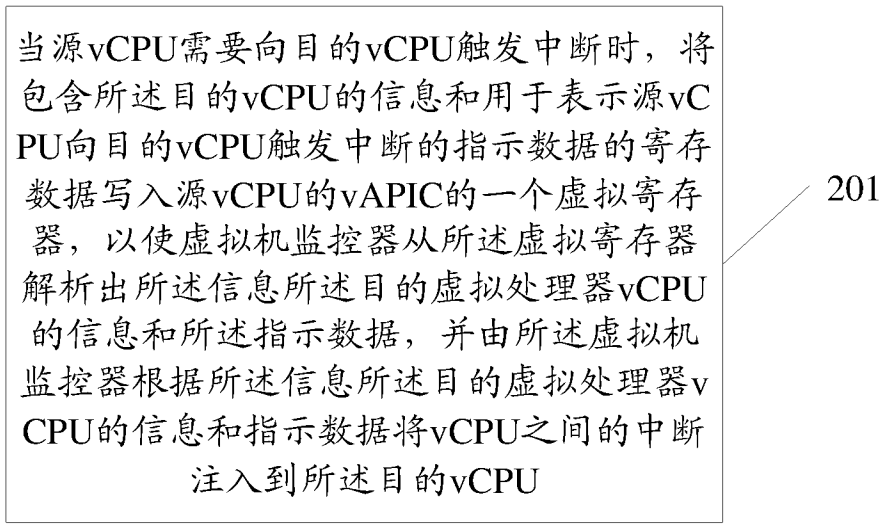

当源vCPU需要向目的vCPU触发中断时，将包含所述目的vCPU的信息和用于表示源vCPU向目的vCPU触发中断的指示数据的寄存数据写入源vCPU的vAPIC的一个虚拟寄存器，以使虚拟机监控器从所述虚拟寄存器解析出所述信息所述目的虚拟处理器vCPU的信息和所述指示数据，并由所述虚拟机监控器根据所述信息所述目的虚拟处理器vCPU的信息和指示数据将vCPU之间的中断注入到所述目的vCPU
201

图 2

当源vCPU需要向目的vCPU触发中断时，将包含所述目的vCPU的信息和用于表示源vCPU向目的vCPU触发中断的指示数据的寄存数据写入源vCPU的vAPIC的所述vICR的高位寄存器或者低位寄存器，以使虚拟机监控器从高位寄存器或者低位寄存器解析出所述信息所述目的虚拟处理器vCPU的信息和所述指示数据，并由所述虚拟机监控器根据所述信息所述目的虚拟处理器vCPU的信息和指示数据将vCPU之间的中断注入到所述目的vCPU

301

图 3

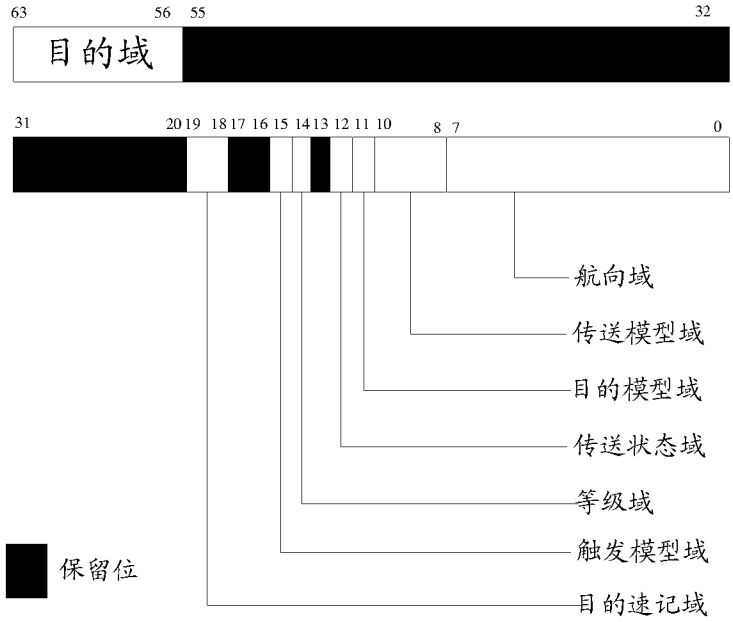


图 4

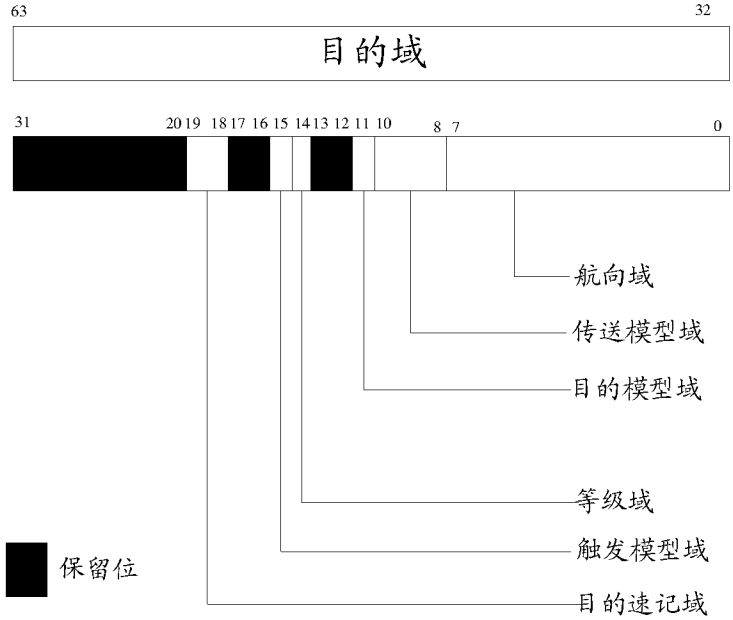

63
32
目的域
31
20 19 18 17 16 15 14 13 12 11 10
8 7
0
航向域
传送模型域
目的模型域
等级域
触发模型域
保留位
目的速记域

图 5

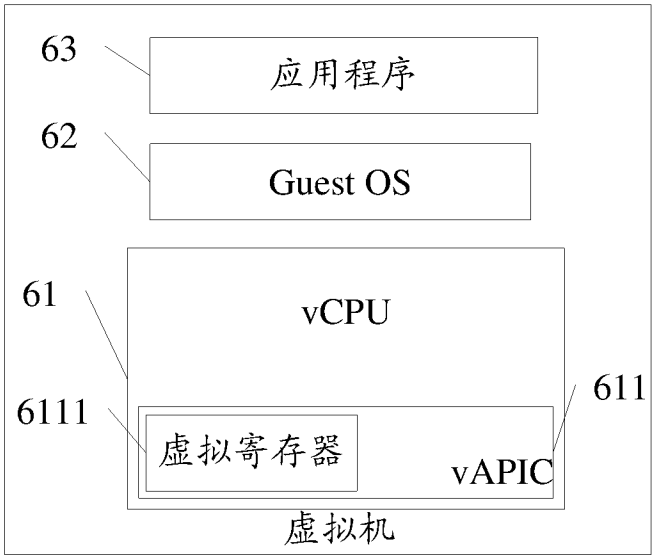

63
应用程序
62
Guest OS
61
vCPU
611
6111
虚拟寄存器
vAPIC
虚拟机

图 6

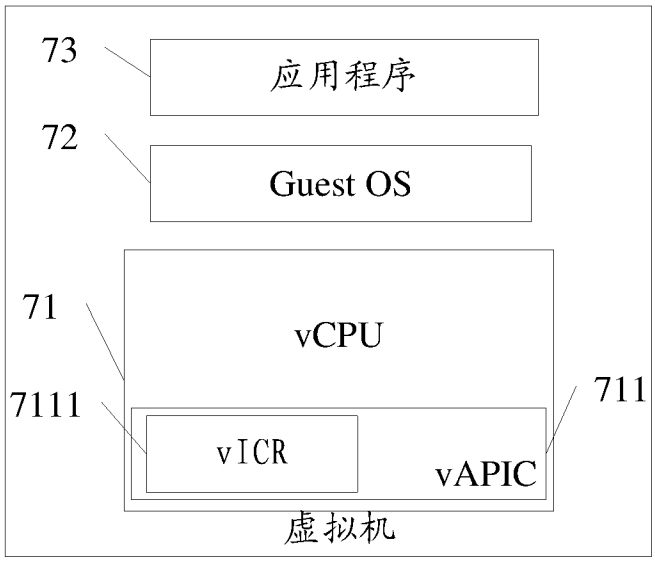

73
应用程序
72
Guest OS
71
vCPU
7111
711
vICR
vAPIC
虚拟机

图 7

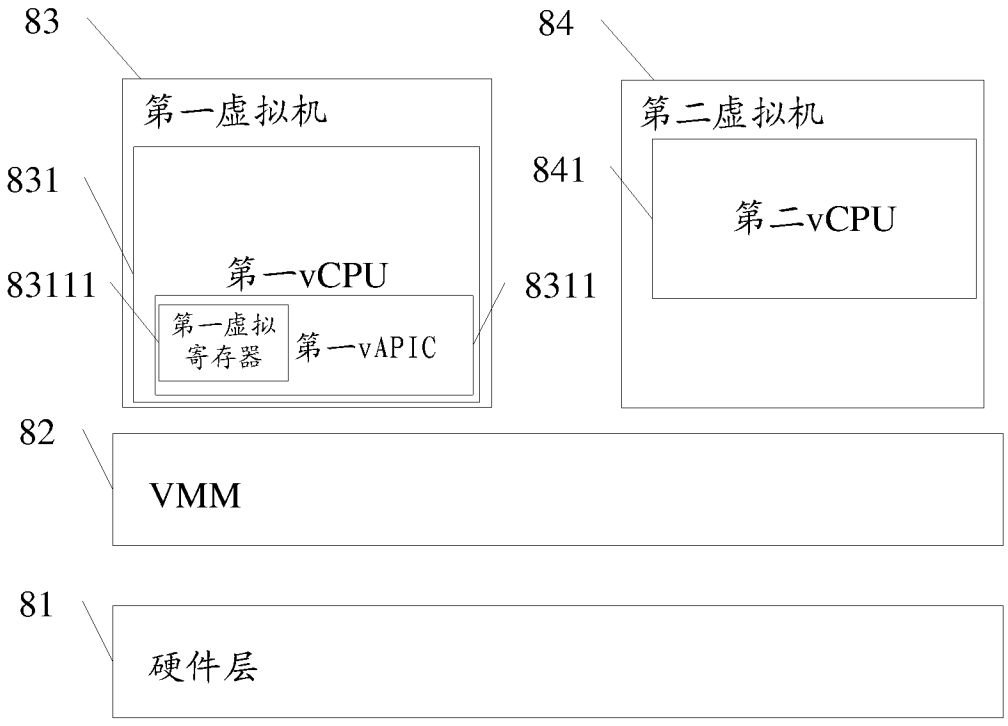

83
84
第一虚拟机
第二虚拟机
831
841
第二vCPU
第一vCPU
83111
8311
第一虚拟寄存器
第一vAPIC
82
VMM
81
硬件层

图 8

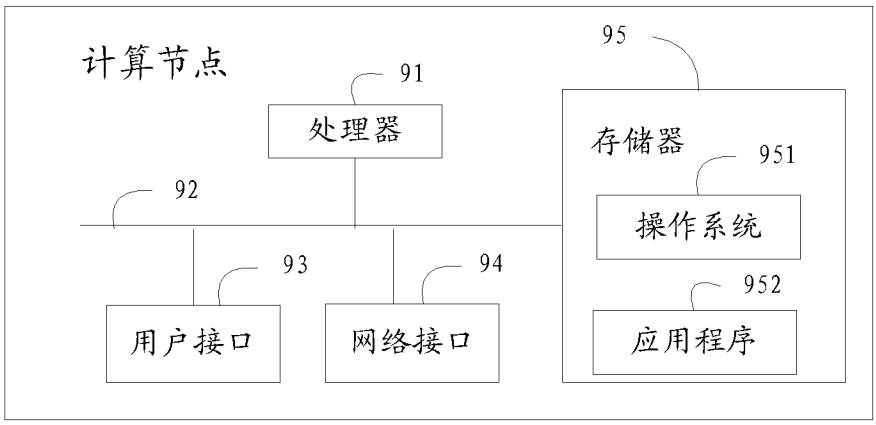

计算节点
91
处理器
95
存储器
951
操作系统
92
93
94
952
用户接口
网络接口
应用程序

图 9

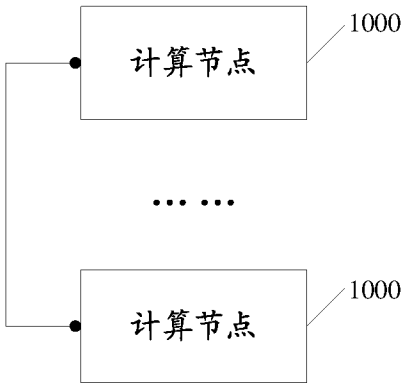

计算节点
1000
……
计算节点
1000

图 10

INTERNATIONAL SEARCH REPORT

International application No.
PCT/CN2014/074412

A. CLASSIFICATION OF SUBJECT MATTER

G06F 9/48 (2006.01) i

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

H04L; H04M; G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

WPI; EPODOC; CNKI; IEEE; CNPAT: virtual machine, virtual processor, recognize, advanced programmable interrupt controller, virtual machine monitor, interrupt command register, virtual, processor, interrupt, APIC, VMM, ICR, register, ID, identifier, index, data, information, indicate, machine, monitor

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	CN 101727414 A (INTEL CORPORATION), 09 June 2010 (09.06.2010), description, paragraphs [0019]-[0021], and figure 4	1-16
Y	CN 101751284 A (BEIJING UNIVERSITY OF AERONAUTICS AND ASTRONAUTICS), 23 June 2010 (23.06.2010), description, paragraphs [0049]-[0057]	1-16
PX	CN 103559087 A (HUAWEI TECHNOLOGIES CO., LTD.), 05 February 2014 (05.02.2014), claims 1-16, and description, paragraphs [0058]-[0193]	1-16

☐ Further documents are listed in the continuation of Box C. ☒ See patent family annex.

* Special categories of cited documents:

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
02 July 2014 (02.07.2014)	**01 August 2014 (01.08.2014)**
Name and mailing address of the ISA/CN: State Intellectual Property Office of the P. R. China No. 6, Xitucheng Road, Jimenqiao Haidian District, Beijing 100088, China Facsimile No.: (86-10) 62019451	Authorized officer **HE, Linlin** Telephone No.: (86-10) **61648280**

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.
PCT/CN2014/074412

Patent Documents referred in the Report	Publication Date	Patent Family	Publication Date
CN 101727414 A	09 June 2010	US 2010106875 A1	29 April 2010
		WO 2010062514 A2	03 June 2010
		DE 102009050983 A1	02 June 2010
		INDEL 200902212 A	18 June 2010
		TW 201032055 A	01 September 2010
		JP 2014029702 A	13 February 2014
		JP 2012507080 A	22 March 2012
		US 2012124264 A1	17 May 2012
		CN 103399833 A	20 November 2013
		INDELNP 201102342 E	06 January 2012
CN 101751284 A	23 June 2010	None	
CN 103559087 A	05 February 2014	None	

国际检索报告

国际申请号
PCT/CN2014/074412

A. 主题的分类

G06F 9/48 (2006.01) i

按照国际专利分类(IPC)或者同时按照国家分类和IPC两种分类

B. 检索领域

检索的最低限度文献(标明分类系统和分类号)

H04L; H04M; G06F

包含在检索领域中的除最低限度文献以外的检索文献

在国际检索时查阅的电子数据库(数据库的名称，和使用的检索词（如使用））

WPI; EPODOC; CNKI; IEEE; CNPAT: 虚拟机、虚拟处理器、虚机、中断、数据、信息、指示、识别、标识、高级可编程中断控制器、虚拟机监视器、中断命令寄存器、virtual、processor、interrupt、APIC、VMM、ICR、register、ID、identifier、index、data、information、indicate、machine、monitor

C. 相关文件

类 型*	引用文件，必要时，指明相关段落	相关的权利要求
Y	CN 101727414 A (英特尔公司) 2010年 6月 09日 (2010-06-09) 说明书第[0019]-[0021]段，附图4	1-16
Y	CN 101751284 A (北京航空航天大学) 2010年 6月 23日 (2010-06-23) 说明书第[0049]-[0057]段	1-16
PX	CN 103559087 A (华为技术有限公司) 2014年 2月 05日 (2014-02-05) 权利要求第1-16项，说明书第[0058]-[0193]段	1-16

☐ 其余文件在C栏的续页中列出。

☑ 见同族专利附件。

* 引用文件的具体类型：

"A" 认为不特别相关的表示了现有技术一般状态的文件

"E" 在国际申请日的当天或之后公布的在先申请或专利

"L" 可能对优先权要求构成怀疑的文件，或为确定另一篇引用文件的公布日而引用的或者因其他特殊理由而引用的文件（如具体说明的）

"O" 涉及口头公开、使用、展览或其他方式公开的文件

"P" 公布日先于国际申请日但迟于所要求的优先权日的文件

"T" 在申请日或优先权日之后公布，与申请不相抵触，但为了理解发明之理论或原理的在后文件

"X" 特别相关的文件，单独考虑该文件，认定要求保护的发明不是新颖的或不具有创造性

"Y" 特别相关的文件，当该文件与另一篇或者多篇该类文件结合并且这种结合对于本领域技术人员为显而易见时，要求保护的发明不具有创造性

"&" 同族专利的文件

国际检索实际完成的日期	国际检索报告邮寄日期
2014年 7月 02日	2014年 8月 01日

ISA/CN的名称和邮寄地址	受权官员
中华人民共和国国家知识产权局(ISA/CN) 北京市海淀区蓟门桥西土城路6号 100088 中国	何琳琳
传真号 (86-10)62019451	电话号码 (86-10)61648280

国际检索报告
关于同族专利的信息

国际申请号
PCT/CN2014/074412

检索报告引用的专利文件	公布日（年/月/日）	同族专利	公布日（年/月/日）
CN 101727414 A	2010年 6月 09日	US 2010106875A1	2010年 4月 29日
		WO 2010062514A2	2010年 6月 03日
		DE 102009050983A1	2010年 6月 02日
		IND EL200902212A	2010年 6月 18日
		TW 201032055A	2010年 9月 01日
		JP 2014029702A	2014年 2月 13日
		JP 2012507080A	2012年 3月 22日
		US 2012124264A1	2012年 5月 17日
		CN 103399833A	2013年 11月 20日
		INDE LNP201102342E	2012年 1月 06日
CN 101751284 A	2010年 6月 23日	无	
CN 103559087 A	2014年 2月 05日	无	