

进程处理方法、装置、电子设备及计算机可读存储介质

本申请要求 2019 年 09 月 24 日递交的申请号为 201910907661.5、发明名称为“进程处理方法、装置、电子设备及计算机可读存储介质”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。

5

技术领域

本发明实施例涉及进程处理技术领域，具体涉及一种进程处理方法、装置、电子设备及计算机可读存储介质。

10 背景技术

随着数据技术的发展，为了提高数据的使用效率和安全性，虚拟机（Virtual Machine，VM）的概念应运而生，虚拟机是指通过软件模拟的具有完整硬件系统功能的、运行在一个完全隔离环境中的完整计算机系统。在 KVM（Kernel-based Virtual Machine，基于内核的虚拟机）架构下，虚拟处理器（vcpu）作为普通进程运行，当然宿主机上也会存在一些系统和运维程序。在虚拟处理器运行的同时，如果系统和运维程序也同时运行，将会对虚拟处理器资源造成争夺，并且如果系统和运维程序运行于内核态，不主动出让资源，则会导致出现虚拟处理器单次长时间抖动现象。现有技术通常使用开启内核抢占来解决资源争夺的问题，但内核抢占只是提供了更多的抢占时间点，并不能实现任意时刻的资源抢占，另外，内核抢占的实施还会带来整个内核的改动，进而引入更多的稳定性和安全性挑战。

20

发明内容

本发明实施例提供一种进程处理方法、装置、电子设备及计算机可读存储介质。

第一方面，本发明实施例中提供了一种进程处理方法。

25

具体的，所述进程处理方法，包括：

响应于中断队列中当前中断事件处理结束，检测内核进程的阻塞时间；

当所述内核进程的阻塞时间超过预设时间阈值时，触发虚拟处理器软中断事件；

响应于所述虚拟处理器软中断事件被触发，向所述虚拟处理器发送中间启动指令，

以使所述虚拟处理器在预设时间段内运行。

30

结合上述实现方式，所述响应于硬中断事件处理结束，检测内核进程的阻塞时间之

前，还包括：

响应于检测到中断事件被触发，处理中断队列中的当前中断事件。

结合上述实现方式，所述响应于检测到中断事件被触发，处理中断队列中的当前中断事件之前，还包括：

- 5 获取与所述内核进程对应的运行队列，并按照队列内预设运行单元排列顺序以所述虚拟处理器的预设运行单元为单位运行，直至接收到进程调度命令，其中，所述运行队列中放置有一个或多个虚拟处理器预设运行单元。

结合上述实现方式，所述获取与所述内核进程对应的运行队列，并按照队列内预设运行单元排列顺序以所述虚拟处理器的预设运行单元为单位运行，直至接收到进程调度

- 10 命令之前，还包括：

对于内核进程进行初始化。

结合上述实现方式，所述对于内核进程进行初始化，包括：

创建内核进程；

- 15 响应于接收到所述虚拟处理器发送的挂载请求，将所述虚拟处理器的一个或多个预设运行单元放置于所述内核进程对应的运行队列中，向所述虚拟处理器发送挂载成功消息；

响应于接收到所述虚拟处理器发送的内核进程唤醒请求，启动所述内核进程。

结合上述实现方式，还包括：

- 20 响应于接收到预设运行请求，向所述虚拟处理器发送中间唤醒请求，并在接收到所述虚拟处理器发送的唤醒成功消息后进入睡眠状态，其中，所述中间唤醒请求携带有所述预设运行请求信息；

响应于接收到所述虚拟处理器发送的预设运行请求处理结束消息，进入运行状态。

结合上述实现方式，还包括：

响应于所述预设时间段结束，处理所述中断队列中的下一中断事件。

- 25 第二方面，本发明实施例中提供了一种进程处理方法。

具体的，所述进程处理方法，包括：

对于虚拟处理器进行初始化；

响应于接收到虚拟机监视器发送的中间启动指令，启动中间运行，其中，所述中间运行的运行时长为预设时间段；

- 30 响应于所述预设时间段结束，停止所述中间运行。

结合上述实现方式，所述对于虚拟处理器进行初始化，包括：

生成一个或多个预设运行单元；

向所述虚拟机监视器发送挂载请求，以将所述一个或多个预设运行单元放置于所述虚拟机监视器内核进程对应的运行队列中；

- 5 响应于接收到所述虚拟机监视器发送的挂载成功消息，向所述虚拟机监视器发送内核进程唤醒请求并进入睡眠状态，以使所述虚拟机监视器启动所述内核进程。

结合上述实现方式，还包括：

响应于接收到所述虚拟机监视器发送的中间唤醒请求，由睡眠状态进入中间运行状态，并向所述虚拟机监视器发送唤醒成功消息，其中，所述中间唤醒请求携带有所述预设运行请求信息；

- 10 运行所述预设运行请求，并在所述预设运行请求处理结束后，向所述虚拟机监视器发送预设运行请求处理结束消息，进入睡眠状态。

第三方面，本发明实施例中提供了一种进程处理方法。

具体的，所述进程处理方法，包括：

- 15 对于虚拟机监视器的内核进程和虚拟处理器进行初始化；

当所述内核进程中断事件被触发且满足预设条件时，对于所述虚拟处理器进行中间启动，以使所述虚拟处理器在预设时间段内运行；

响应于所述预设时间段结束，所述虚拟机监视器处理中断队列中的下一中断事件。

结合上述实现方式，所述对于虚拟机监视器的内核进程进行初始化，被实施为：

- 20 创建内核进程；

响应于接收到所述虚拟处理器发送的挂载请求，将所述虚拟处理器的一个或多个预设运行单元放置于所述内核进程对应的运行队列中，向所述虚拟处理器发送挂载成功消息；

响应于接收到所述虚拟处理器发送的内核进程唤醒请求，启动所述内核进程。

- 25 结合上述实现方式，所述对于虚拟处理器进行初始化，被实施为：

生成一个或多个预设运行单元；

向所述虚拟机监视器发送挂载请求，以将所述一个或多个预设运行单元放置于所述虚拟机监视器内核进程对应的运行队列中；

- 30 响应于接收到所述虚拟机监视器发送的挂载成功消息，向所述虚拟机监视器发送内核进程唤醒请求并进入睡眠状态，以使所述虚拟机监视器启动所述内核进程。

结合上述实现方式，所述当所述内核进程中断事件被触发且满足预设条件时，对于所述虚拟处理器进行中间启动，包括：

响应于中断队列中当前中断事件处理结束，所述虚拟机监视器检测内核进程的阻塞时间；

- 5 当所述内核进程的阻塞时间超过预设时间阈值时，所述虚拟机监视器触发虚拟处理器软中断事件；

响应于所述虚拟处理器软中断事件被触发，所述虚拟机监视器向所述虚拟处理器发送中间启动指令；

- 10 响应于接收到所述虚拟机监视器发送的中间启动指令，所述虚拟处理器启动中间运行，其中，所述中间运行的运行时长为预设时间段；

响应于所述预设时间段结束，所述虚拟处理器停止所述中间运行。

结合上述实现方式，所述响应于中断队列中当前中断事件处理结束，所述虚拟机监视器检测内核进程的阻塞时间之前，还包括：

响应于检测到中断事件被触发，所述虚拟机监视器处理中断队列中的当前中断事件。

- 15 结合上述实现方式，所述响应于检测到中断事件被触发，所述虚拟机监视器处理中断队列中的当前中断事件之前，还包括：

所述虚拟机监视器获取与所述内核进程对应的运行队列，并按照队列内预设运行单元排列顺序以所述虚拟处理器的预设运行单元为单位运行，直至接收到进程调度命令，其中，所述运行队列中放置有一个或多个虚拟处理器预设运行单元。

- 20 结合上述实现方式，还包括：

响应于接收到预设运行请求，所述虚拟机监视器对于所述虚拟处理器进行中间唤醒。

结合上述实现方式，所述响应于接收到预设运行请求，所述虚拟机监视器对于所述虚拟处理器进行中间唤醒，包括：

- 25 响应于接收到预设运行请求，所述虚拟机监视器向所述虚拟处理器发送中间唤醒请求，其中，所述中间唤醒请求携带有所述预设运行请求信息；

响应于接收到所述虚拟机监视器发送的中间唤醒请求，所述虚拟处理器由睡眠状态进入中间运行状态，并向所述虚拟机监视器发送唤醒成功消息；

所述虚拟机监视器在接收到所述虚拟处理器发送的唤醒成功消息后进行睡眠状态；

- 30 所述虚拟处理器运行所述预设运行请求，并在所述预设运行请求处理结束后，向所述虚拟机监视器发送预设运行请求处理结束消息，进入睡眠状态；

响应于接收到所述虚拟处理器发送的预设运行请求处理结束消息，所述虚拟机监视器进入运行状态。

第四方面，本发明实施例中提供了一种进程处理装置。

具体的，所述进程处理装置，包括：

5 检测模块，被配置为响应于中断队列中当前中断事件处理结束，检测内核进程的阻塞时间；

触发模块，被配置为当所述内核进程的阻塞时间超过预设时间阈值时，触发虚拟处理器软中断事件；

10 第一发送模块，被配置为响应于所述虚拟处理器软中断事件被触发，向所述虚拟处理器发送中间启动指令，以使所述虚拟处理器在预设时间段内运行。

结合上述实现方式，所述检测模块之前，还包括：

第一处理模块，被配置为响应于检测到中断事件被触发，处理中断队列中的当前中断事件。

结合上述实现方式，所述第一处理模块之前，还包括：

15 第一运行模块，被配置为获取与所述内核进程对应的运行队列，并按照队列内预设运行单元排列顺序以所述虚拟处理器的预设运行单元为单位运行，直至接收到进程调度命令，其中，所述运行队列中放置有一个或多个虚拟处理器预设运行单元。

结合上述实现方式，所述第一运行模块之前，还包括：

第一初始化模块，被配置为对于内核进程进行初始化。

20 结合上述实现方式，所述第一初始化模块包括：

创建子模块，被配置为创建内核进程；

挂载子模块，被配置为响应于接收到所述虚拟处理器发送的挂载请求，将所述虚拟处理器的一个或多个预设运行单元放置于所述内核进程对应的运行队列中，向所述虚拟处理器发送挂载成功消息；

25 第一启动子模块，被配置为响应于接收到所述虚拟处理器发送的内核进程唤醒请求，启动所述内核进程。

结合上述实现方式，还包括：

30 第二发送模块，被配置为响应于接收到预设运行请求，向所述虚拟处理器发送中间唤醒请求，并在接收到所述虚拟处理器发送的唤醒成功消息后进入睡眠状态，其中，所述中间唤醒请求携带有所述预设运行请求信息；

第二运行模块，被配置为响应于接收到所述虚拟处理器发送的预设运行请求处理结束消息，进入运行状态。

结合上述实现方式，还包括：

第二处理模块，被配置为响应于所述预设时间段结束，处理所述中断队列中的下一中断事件。

第五方面，本发明实施例中提供了一种进程处理装置。

具体的，所述进程处理装置，包括：

第二初始化模块，被配置为对于虚拟处理器进行初始化；

第一启动模块，被配置为响应于接收到虚拟机监视器发送的中间启动指令，启动中间运行，其中，所述中间运行的运行时长为预设时间段；

停止模块，被配置为响应于所述预设时间段结束，停止所述中间运行。

结合上述实现方式，所述第二初始化模块包括：

生成子模块，被配置为生成一个或多个预设运行单元；

第一发送子模块，被配置为向所述虚拟机监视器发送挂载请求，以将所述一个或多个预设运行单元放置于所述虚拟机监视器内核进程对应的运行队列中；

第二发送子模块，被配置为响应于接收到所述虚拟机监视器发送的挂载成功消息，向所述虚拟机监视器发送内核进程唤醒请求并进入睡眠状态，以使所述虚拟机监视器启动所述内核进程。

结合上述实现方式，还包括：

第三发送模块，被配置为响应于接收到所述虚拟机监视器发送的中间唤醒请求，由睡眠状态进入中间运行状态，并向所述虚拟机监视器发送唤醒成功消息，其中，所述中间唤醒请求携带有所述预设运行请求信息；

第三运行模块，被配置为运行所述预设运行请求，并在所述预设运行请求处理结束后，向所述虚拟机监视器发送预设运行请求处理结束消息，进入睡眠状态。

第六方面，本发明实施例中提供了一种进程处理装置。

具体的，所述进程处理装置，包括：

第三初始化模块，被配置为对于虚拟机监视器的内核进程和虚拟处理器进行初始化；

第二启动模块，被配置为当所述内核进程中断事件被触发且满足预设条件时，对于所述虚拟处理器进行中间启动，以使所述虚拟处理器在预设时间段内运行；

第三处理模块，被配置为响应于所述预设时间段结束，处理中断队列中的下一中断

事件。

结合上述实现方式，所述第三初始化模块中对于虚拟机监视器的内核进程进行初始化的部分被配置为：

创建内核进程；

- 5 响应于接收到所述虚拟处理器发送的挂载请求，将所述虚拟处理器的一个或多个预设运行单元放置于所述内核进程对应的运行队列中，向所述虚拟处理器发送挂载成功消息；

响应于接收到所述虚拟处理器发送的内核进程唤醒请求，启动所述内核进程。

- 10 结合上述实现方式，所述第三初始化模块中对于虚拟处理器进行初始化的部分被配置为：

生成一个或多个预设运行单元；

向所述虚拟机监视器发送挂载请求，以将所述一个或多个预设运行单元放置于所述虚拟机监视器内核进程对应的运行队列中；

- 15 响应于接收到所述虚拟机监视器发送的挂载成功消息，向所述虚拟机监视器发送内核进程唤醒请求并进入睡眠状态，以使所述虚拟机监视器启动所述内核进程。

结合上述实现方式，所述第二启动模块包括：

检测子模块，被配置为响应于中断队列中当前中断事件处理结束，检测内核进程的阻塞时间；

- 20 触发子模块，被配置为当所述内核进程的阻塞时间超过预设时间阈值时，触发虚拟处理器软中断事件；

第三发送子模块，被配置为响应于所述虚拟处理器软中断事件被触发，向所述虚拟处理器发送中间启动指令；

第二启动子模块，被配置为响应于接收到所述虚拟机监视器发送的中间启动指令，启动中间运行，其中，所述中间运行的运行时长为预设时间段；

- 25 停止子模块，被配置为响应于所述预设时间段结束，停止所述中间运行。

结合上述实现方式，所述检测子模块之前，还包括：

处理子模块，被配置为响应于检测到中断事件被触发，处理中断队列中的当前中断事件。

结合上述实现方式，所述处理子模块之前，还包括：

- 30 第一运行子模块，被配置为获取与所述内核进程对应的运行队列，并按照队列内预

设运行单元排列顺序以所述虚拟处理器的预设运行单元为单位运行，直至接收到进程调度命令，其中，所述运行队列中放置有一个或多个虚拟处理器预设运行单元。

结合上述实现方式，还包括：

5 唤醒模块，被配置为响应于接收到预设运行请求，对于所述虚拟处理器进行中间唤醒。

结合上述实现方式，所述唤醒模块包括：

第四发送子模块，被配置为响应于接收到预设运行请求，向所述虚拟处理器发送中间唤醒请求，其中，所述中间唤醒请求携带有所述预设运行请求信息；

10 第五发送子模块，被配置为响应于接收到所述虚拟机监视器发送的中间唤醒请求，由睡眠状态进入中间运行状态，并向所述虚拟机监视器发送唤醒成功消息；

第一状态改变子模块，被配置为在接收到所述虚拟处理器发送的唤醒成功消息后进入睡眠状态；

第二运行子模块，被配置为运行所述预设运行请求，并在所述预设运行请求处理结束后，向所述虚拟机监视器发送预设运行请求处理结束消息，进入睡眠状态；

15 第二状态改变子模块，被配置为响应于接收到所述虚拟处理器发送的预设运行请求处理结束消息，进入运行状态。

20 第七方面，本发明实施例提供了一种电子设备，包括存储器和处理器，所述存储器用于存储一条或多条支持进程处理装置执行上述进程处理方法的计算机指令，所述处理器被配置为用于执行所述存储器中存储的计算机指令。所述进程处理装置还可以包括通信接口，用于进程处理装置与其他设备或通信网络通信。

第八方面，本发明实施例提供了一种计算机可读存储介质，用于存储进程处理装置所用的计算机指令，其包含用于执行上述进程处理方法为进程处理装置所涉及的计算机指令。

本发明实施例提供的技术方案可包括以下有益效果：

25 上述技术方案通过在处理中断事件时检测内核进程的阻塞时间，来判断是否需要对于虚拟处理器进行中间启动，并在满足预设条件时，对于虚拟处理器进行中间启动。该技术方案能够实现在软中断上下文中对于虚拟处理器进行补偿运行，从而在保证数据和资源的稳定性和安全性的前提下，避免了虚拟处理器出现长时间阻塞和单次长时间抖动现象。

30 应当理解的是，以上的一般描述和后文的细节描述仅是示例性和解释性的，并不能

限制本发明实施例。

附图说明

- 结合附图，通过以下非限制性实施方式的详细描述，本发明实施例的其它特征、目的和优点将变得更加明显。在附图中：
- 图 1 示出根据本发明一实施方式的进程处理方法的流程图；
- 图 2 示出根据本发明不同实施方式的进程处理方法的流程图；
- 图 3 示出根据本发明不同实施方式的进程处理方法的流程图；
- 图 4 示出根据本发明不同实施方式的进程处理方法的流程图；
- 10 图 5 示出根据图 4 所示实施方式的进程处理方法的步骤 S401 的流程图；
- 图 6 示出根据本发明不同实施方式的进程处理方法的流程图；
- 图 7 示出根据本发明不同实施方式的进程处理方法的流程图；
- 图 8 示出根据本发明不同实施方式的进程处理方法的流程图；
- 图 9 示出根据图 8 所示实施方式的进程处理方法的步骤 S801 的流程图；
- 15 图 10 示出根据本发明不同实施方式的进程处理方法的流程图；
- 图 11 示出根据本发明不同实施方式的进程处理方法的流程图；
- 图 12 示出根据图 11 所示实施方式的进程处理方法的步骤 S1102 的流程图；
- 图 13 示出根据图 11 所示另一实施方式的进程处理方法的步骤 S1102 的流程图；
- 图 14 示出根据图 11 所示再一实施方式的进程处理方法的步骤 S1102 的流程图；
- 20 图 15 示出根据本发明再一实施方式的进程处理方法的流程图；
- 图 16 示出根据图 15 所示实施方式的进程处理方法的步骤 S1504 的流程图；
- 图 17 示出根据本发明一实施方式的进程处理装置的结构框图；
- 图 18 示出根据本发明不同实施方式的进程处理装置的结构框图；
- 图 19 示出根据本发明不同实施方式的进程处理装置的结构框图；
- 25 图 20 示出根据本发明不同实施方式的进程处理装置的结构框图；
- 图 21 示出根据图 20 所示实施方式的第一初始化模块 2001 的结构框图；
- 图 22 示出根据本发明不同实施方式的进程处理装置的结构框图；
- 图 23 示出根据本发明不同实施方式的进程处理装置的结构框图；
- 图 24 示出根据本发明不同实施方式的进程处理装置的结构框图；
- 30 图 25 示出根据图 24 所示实施方式的第二初始化模块 2401 的结构框图；

- 图 26 示出根据本发明不同实施方式的进程处理装置的结构框图；
图 27 示出根据本发明不同实施方式的进程处理装置的结构框图；
图 28 示出根据图 27 所示实施方式的第二启动模块 2702 的结构框图；
图 29 示出根据图 27 所示另一实施方式的第二启动模块 2702 的结构框图；
5 图 30 示出根据图 27 所示再一实施方式的第二启动模块 2702 的结构框图；
图 31 示出根据本发明再一实施方式的进程处理装置的结构框图；
图 32 示出根据图 31 所示实施方式的唤醒模块 3104 的结构框图；
图 33 示出一应用场景中的进程处理方法的流程图；
图 34 示出根据本发明一实施方式的电子设备的结构框图；
10 图 35 是适于用来实现根据本发明一实施方式的进程处理方法的计算机系统的结构示意图。

具体实施方式

下文中，将参考附图详细描述本发明实施例的示例性实施方式，以使本领域技术人员可容易地实现它们。此外，为了清楚起见，在附图中省略了与描述示例性实施方式无关的部分。

在本发明实施例中，应理解，诸如“包括”或“具有”等的术语旨在指示本说明书中所公开的特征、数字、步骤、行为、部件、部分或其组合的存在，并且不欲排除一个或多个其他特征、数字、步骤、行为、部件、部分或其组合存在或被添加的可能性。

20 另外还需要说明的是，在不冲突的情况下，本发明中的实施例及实施例中的特征可以相互组合。下面将参考附图并结合实施例来详细说明本发明实施例。

本发明实施例提供的技术方案通过在处理中断事件时检测内核进程的阻塞时间，来判断是否需要对于虚拟处理器进行中间启动，并在满足预设条件时，对于虚拟处理器进行中间启动。该技术方案能够实现在软中断上下文中对于虚拟处理器进行补偿运行，从而在保证数据和资源的稳定性和安全性的前提下，避免了虚拟处理器出现长时间阻塞和
25 单次长时间抖动现象。

图 1 示出根据本发明一实施方式的进程处理方法的流程图，适用于虚拟机监视器，如图 1 所示，所述进程处理方法包括以下步骤 S101-S103：

在步骤 S101 中，响应于中断队列中当前中断事件处理结束，检测内核进程的阻塞时
30 间；

在步骤 S102 中,当所述内核进程的阻塞时间超过预设时间阈值时,触发虚拟处理器软中断事件;

在步骤 S103 中,响应于所述虚拟处理器软中断事件被触发,向所述虚拟处理器发送中间启动指令,以使所述虚拟处理器在预设时间段内运行。

- 5 上文提及,随着数据技术的发展,为了提高数据的使用效率和安全性,虚拟机的概念应运而生,虚拟机是指通过软件模拟的具有完整硬件系统功能的、运行在一个完全隔离环境中的完整计算机系统。在 KVM 架构下,虚拟处理器作为普通进程运行,当然宿主主机上也会存在一些系统和运维程序。在虚拟处理器运行的同时,如果系统和运维程序也同时运行,将会对虚拟处理器资源造成争抢,并且如果系统和运维程序运行于内核态,
- 10 不主动出让资源,则会导致出现虚拟处理器单次长时间抖动现象。现有技术通常使用开启内核抢占来解决资源争夺的问题,但内核抢占只是提供了更多的抢占时间点,并不能实现任意时刻的资源抢占,另外,内核抢占的实施还会带来整个内核的改动,进而引入更多的稳定性和安全性挑战。

- 考虑到上述问题,在该实施方式中,提出一种进程处理方法,该方法通过在处理中
- 15 断事件时检测内核进程的阻塞时间,来判断是否需要对于虚拟处理器进行中间启动,并在满足预设条件时,对于虚拟处理器进行中间启动。该技术方案能够实现在软中断上下文中对于虚拟处理器进行补偿运行,从而在保证数据和资源的稳定性和安全性的前提下,避免了虚拟处理器出现长时间阻塞和单次长时间抖动现象。

- 在本发明一实施方式中,所述中断队列指的是虚拟机监视器获取的中断事件组成的
- 20 队列,处于所述中断队列中的中断事件通常被按照在队列中的顺序顺次执行。

- 为了避免虚拟处理器出现长时间阻塞和单次长时间抖动现象,在本发明一实施方式中,每当中断队列中的某一中断事件处理结束,就检测一下内核进程的阻塞时间,若所述内核进程的阻塞时间超过预设时间阈值,则认为虚拟处理器有可能出现长时间阻塞现象,需要对于虚拟处理器执行补偿运行操作,则触发虚拟处理器软中断事件,之后向所
- 25 述虚拟处理器发送中间启动指令,以使所述虚拟处理器在预设时间段内运行,其中,所述预设时间段指的是虚拟处理器执行补偿运行的时长,其值可根据实际应用的需要进行设置,本公开对其不作具体限定,比如可以将所述预设时间段设置为 0.1-0.5ms 等等,例如,若所述预设时间段设置为 0.1ms,则意味着所述虚拟处理器补偿运行 0.1ms。

- 其中,所述预设时间阈值用于判断所述内核进程的阻塞时间长短,其值可根据实际
- 30 应用的需要进行设置,本公开对其不作具体限定。

在本发明一实施方式中，所述步骤 S101，即响应于中断队列中当前中断事件处理结束，检测内核进程的阻塞时间的步骤之前，还包括响应于检测到中断事件被触发，处理中断队列中的当前中断事件的步骤，即如图 2 所示，所述进程处理方法包括以下步骤 S201-S204：

- 5 在步骤 S201 中，响应于检测到中断事件被触发，处理中断队列中的当前中断事件；
 在步骤 S202 中，响应于中断队列中当前中断事件处理结束，检测内核进程的阻塞时间；

 在步骤 S203 中，当所述内核进程的阻塞时间超过预设时间阈值时，触发虚拟处理器软中断事件；

- 10 在步骤 S204 中，响应于所述虚拟处理器软中断事件被触发，向所述虚拟处理器发送中间启动指令，以使所述虚拟处理器在预设时间段内运行。

 上文提及，所述中断队列中有可能存在一个或多个中断事件，因此，在该实施方式中，在检测到中断事件被触发后，按照当前时间以及中断队列中中断事件的排列顺序来处理所述中断队列中的当前中断事件，并在所述当前中断事件处理结束之后，再检测内核进程由于中断事件的处理而产生的阻塞时间。

15 在本发明一实施方式中，所述步骤 S201，即响应于检测到中断事件被触发，处理中断队列中的当前中断事件的步骤之前，还包括获取与所述内核进程对应的运行队列，并按照队列内预设运行单元排列顺序以所述虚拟处理器的预设运行单元为单位运行，直至接收到进程调度命令的步骤，即如图 3 所示，所述进程处理方法包括以下步骤 S301-S305：

- 20 在步骤 S301 中，获取与所述内核进程对应的运行队列，并按照队列内预设运行单元排列顺序以所述虚拟处理器的预设运行单元为单位运行，直至接收到进程调度命令，其中，所述运行队列中放置有一个或多个虚拟处理器预设运行单元；

 在步骤 S302 中，响应于检测到中断事件被触发，处理中断队列中的当前中断事件；
 在步骤 S303 中，响应于中断队列中当前中断事件处理结束，检测内核进程的阻塞时间；

25 在步骤 S304 中，当所述内核进程的阻塞时间超过预设时间阈值时，触发虚拟处理器软中断事件；

 在步骤 S305 中，响应于所述虚拟处理器软中断事件被触发，向所述虚拟处理器发送中间启动指令，以使所述虚拟处理器在预设时间段内运行。

- 30 为了虚拟处理器数据处理的完整性，在该实施方式中，在运行虚拟处理器数据时，

以所述虚拟处理器的预设运行单元为单位运行，即首先获取、确定与所述内核进程对应的运行队列，并按照队列内预设运行单元排列顺序以所述虚拟处理器的预设运行单元为单位运行，直至接收到进程调度命令停止所述虚拟处理器的运行。其中，所述运行队列中放置有一个或多个虚拟处理器预设运行单元。在本发明一实施方式中，所述虚拟处理器预设运行单元可以由所述虚拟处理器运行的最小循环封装而成。

在本发明一实施方式中，所述虚拟机监视器中可设置一个内核进程也可以设置多个内核进程，其中，所述内核进程的数量可由物理处理器的数量来决定，比如，存在几个物理处理器就设置几个内核进程。其中，每个内核进程负责维护一个运行队列。

在本发明一实施方式中，所述步骤 S301，即获取与所述内核进程对应的运行队列，
10 并按照队列内预设运行单元排列顺序以所述虚拟处理器的预设运行单元为单位运行，直至接收到进程调度命令的步骤之前，还包括对于内核进程进行初始化的步骤，即如图 4 所示，所述进程处理方法包括以下步骤 S401-S406：

在步骤 S401 中，对于内核进程进行初始化；

在步骤 S402 中，获取与所述内核进程对应的运行队列，并按照队列内预设运行单元
15 排列顺序以所述虚拟处理器的预设运行单元为单位运行，直至接收到进程调度命令，其中，所述运行队列中放置有一个或多个虚拟处理器预设运行单元；

在步骤 S403 中，响应于检测到中断事件被触发，处理中断队列中的当前中断事件；

在步骤 S404 中，响应于中断队列中当前中断事件处理结束，检测内核进程的阻塞时间；

20 在步骤 S405 中，当所述内核进程的阻塞时间超过预设时间阈值时，触发虚拟处理器软中断事件；

在步骤 S406 中，响应于所述虚拟处理器软中断事件被触发，向所述虚拟处理器发送中间启动指令，以使所述虚拟处理器在预设时间段内运行。

为了保证内核进程的正常工作的，在该实施方式中，在内核进程开始工作之前，还需
25 对其执行初始化操作。

在本发明一实施方式中，如图 5 所示，所述步骤 S401，即对于内核进程进行初始化的步骤，包括以下步骤 S501-S503：

在步骤 S501 中，创建内核进程；

在步骤 S502 中，响应于接收到所述虚拟处理器发送的挂载请求，将所述虚拟处理器
30 的一个或多个预设运行单元放置于所述内核进程对应的运行队列中，向所述虚拟处理器

发送挂载成功消息；

在步骤 S503 中，响应于接收到所述虚拟处理器发送的内核进程唤醒请求，启动所述内核进程。

在该实施方式中，在对于内核进程进行初始化时，首先创建一个或多个内核进程，
5 对于每个内核进程来说，在接收到所述虚拟处理器发送的挂载请求后，将所述虚拟处理器的一个或多个预设运行单元放置于所述内核进程对应的运行队列中，并向所述虚拟处理器发送挂载成功消息，其中，所述挂载请求可包括内核进程标识信息和请求本体信息，以使所述虚拟机监视器根据所述内核进程标识信息将所述虚拟处理器的一个或多个预设运行单元放置于与所述内核进程标识信息对应的内核进程的运行队列中，当然，所述挂
10 载请求也可仅包括请求本体信息，此时，所述虚拟机监视器随机选择一个内核进程来放置所述虚拟处理器的一个或多个预设运行单元；之后在接收到所述虚拟处理器发送的内核进程唤醒请求后，启动所述内核进程运行。

在本发明一实施方式中，所述方法还包括与所述虚拟处理器交互以执行预设运行请求的步骤，即如图 6 所示，所述进程处理方法包括以下步骤 S601-S605：

15 在步骤 S601 中，响应于中断队列中当前中断事件处理结束，检测内核进程的阻塞时间；

在步骤 S602 中，当所述内核进程的阻塞时间超过预设时间阈值时，触发虚拟处理器软中断事件；

20 在步骤 S603 中，响应于所述虚拟处理器软中断事件被触发，向所述虚拟处理器发送中间启动指令，以使所述虚拟处理器在预设时间段内运行；

在步骤 S604 中，响应于接收到预设运行请求，向所述虚拟处理器发送中间唤醒请求，并在接收到所述虚拟处理器发送的唤醒成功消息后进入睡眠状态，其中，所述中间唤醒请求携带有所述预设运行请求信息；

25 在步骤 S605 中，响应于接收到所述虚拟处理器发送的预设运行请求处理结束消息，进入运行状态。

若所述虚拟机监视器接收到其无法完成、只能由虚拟处理器来完成的运行请求，比如基于用户态的运行请求等，则需向所述虚拟处理器发送携带有所述预设运行请求信息的中间唤醒请求，并在接收到所述虚拟处理器发送的唤醒成功消息后进入睡眠状态，转由所述虚拟处理器来完成所述预设运行请求，并在接收到所述虚拟处理器发送的预设运行请求处理结束消息后，再次进入运行状态。
30

在本发明一实施方式中，所述方法还包括响应于所述预设时间段结束，处理所述中断队列中的下一中断事件的步骤，即如图 7 所示，所述进程处理方法包括以下步骤 S701-S704：

5 在步骤 S701 中，响应于中断队列中当前中断事件处理结束，检测内核进程的阻塞时间；

在步骤 S702 中，当所述内核进程的阻塞时间超过预设时间阈值时，触发虚拟处理器软中断事件；

在步骤 S703 中，响应于所述虚拟处理器软中断事件被触发，向所述虚拟处理器发送中间启动指令，以使所述虚拟处理器在预设时间段内运行；

10 在步骤 S704 中，响应于所述预设时间段结束，处理所述中断队列中的下一中断事件。

在该实施方式中，在所述虚拟处理器执行中间启动预设时间段，即中间启动结束后，所述虚拟机监视器继续处理所述中断队列中的下一中断事件，并在处理完下一中断事件之后，检测内核进程的阻塞时间。若所述中断队列中不存在下一中断事件，则所述虚拟机监视器继续处理其他进程。

15 图 8 示出根据本发明另一实施方式的进程处理方法的流程图，适用于虚拟处理器，如图 8 所示，所述进程处理方法包括以下步骤 S801-S803：

在步骤 S801 中，对于虚拟处理器进行初始化；

在步骤 S802 中，响应于接收到虚拟机监视器发送的中间启动指令，启动中间运行，其中，所述中间运行的运行时长为预设时间段；

20 在步骤 S803 中，响应于所述预设时间段结束，停止所述中间运行。

上文提及，在 KVM 架构下，虚拟处理器作为普通进程运行，当然宿主机上也会存在一些系统和运维程序。在虚拟处理器运行的同时，如果系统和运维程序也同时运行，将会对虚拟处理器资源造成争抢，并且如果系统和运维程序运行于内核态，不主动出让资源，则会导致出现虚拟处理器单次长时间抖动现象。现有技术通常使用开启内核抢占来解决资源争夺的问题，但内核抢占只是提供了更多的抢占时间点，并不能实现任意时刻的资源抢占，另外，内核抢占的实施还会带来整个内核的改动，进而引入更多的稳定性和安全性挑战。

考虑到上述问题，在该实施方式中，提出一种进程处理方法，该方法通过虚拟机监视器在处理中断事件时检测内核进程的阻塞时间，来判断是否需要对于虚拟处理器进行中间启动，并在满足预设条件时，对于虚拟处理器进行中间启动。该技术方案能够实现

30

在软中断上下文中对于虚拟处理器进行补偿运行，从而在保证数据和资源的稳定性和安全性的前提下，避免了虚拟处理器出现长时间阻塞和单次长时间抖动现象。

为了避免虚拟处理器出现长时间阻塞和单次长时间抖动现象，在该实施方式中，在对于虚拟处理器进行初始化，挂载在所述虚拟机监视器上之后，若接收到虚拟机监视器发送的中间启动指令，则启动中间运行，其中，所述虚拟处理器中间运行的运行时长为预设时间段；并在所述预设时间段结束后，停止所述中间运行。

其中，所述预设时间段指的是虚拟处理器执行补偿运行的时长，其值可根据实际应用的需要进行设置，本公开对其不作具体限定，比如可以将所述预设时间段设置为 0.1-0.5ms 等等，例如，若所述预设时间段设置为 0.1ms，则意味着所述虚拟处理器补偿运行 0.1ms。

在本发明一实施方式中，如图 9 所示，所述步骤 S801，即对于虚拟处理器进行初始化的步骤，包括以下步骤 S901-S903：

在步骤 S901 中，生成一个或多个预设运行单元；

在步骤 S902 中，向所述虚拟机监视器发送挂载请求，以将所述一个或多个预设运行单元放置于所述虚拟机监视器内核进程对应的运行队列中；

在步骤 S903 中，响应于接收到所述虚拟机监视器发送的挂载成功消息，向所述虚拟机监视器发送内核进程唤醒请求并进入睡眠状态，以使所述虚拟机监视器启动所述内核进程。

在该实施方式中，在对于虚拟处理器进行初始化时，首先根据所述虚拟处理器运行的最小循环封装生成一个或多个预设运行单元；然后向所述虚拟机监视器发送挂载请求，以将所述一个或多个预设运行单元放置于所述虚拟机监视器内核进程对应的运行队列中由内核进程执行运行操作；在接收到所述虚拟机监视器发送的挂载成功消息后，向所述虚拟机监视器发送内核进程唤醒请求以唤醒所述内核进程，所述虚拟处理器进而进入睡眠状态。

在本发明一实施方式中，所述方法还包括与所述虚拟机监视器交互以执行预设运行请求的步骤，即如图 10 所示，所述进程处理方法包括以下步骤 S1001-S1005：

在步骤 S1001 中，对于虚拟处理器进行初始化；

在步骤 S1002 中，响应于接收到虚拟机监视器发送的中间启动指令，启动中间运行，其中，所述中间运行的运行时长为预设时间段；

在步骤 S1003 中，响应于所述预设时间段结束，停止所述中间运行；

在步骤 S1004 中，响应于接收到所述虚拟机监视器发送的中间唤醒请求，由睡眠状态进入中间运行状态，并向所述虚拟机监视器发送唤醒成功消息，其中，所述中间唤醒请求携带有所述预设运行请求信息；

在步骤 S1005 中，运行所述预设运行请求，并在所述预设运行请求处理结束后，向
5 所述虚拟机监视器发送预设运行请求处理结束消息，进入睡眠状态。

若所述虚拟机监视器接收到其无法完成、只能由虚拟处理器来完成的运行请求，比如基于用户态的运行请求等，则需向所述虚拟处理器发送携带有所述预设运行请求信息的中间唤醒请求以唤醒所述虚拟处理器，此时，所述虚拟处理器接收到所述虚拟机监视器发送的中间唤醒请求后，由睡眠状态进入中间运行状态，并向所述虚拟机监视器发送
10 唤醒成功消息，然后所述虚拟处理器运行所述预设运行请求，并在所述预设运行请求处理结束后，向所述虚拟机监视器发送预设运行请求处理结束消息以重新唤醒所述虚拟机监视器，而所述虚拟处理器则再次进入睡眠状态。

图 11 示出根据本发明再一实施方式的进程处理方法的流程图，适用于进程处理系统，其中，所述进程处理系统包括虚拟机监视器和虚拟处理器，如图 11 所示，所述进程
15 处理方法包括以下步骤 S1101-S1103：

在步骤 S1101 中，对于虚拟机监视器的内核进程和虚拟处理器进行初始化；

在步骤 S1102 中，当所述内核进程中中断事件被触发且满足预设条件时，对于所述虚拟处理器进行中间启动，以使所述虚拟处理器在预设时间段内运行；

在步骤 S1103 中，响应于所述预设时间段结束，所述虚拟机监视器处理中断队列中的
20 的下一中断事件。

上文提及，在 KVM 架构下，虚拟处理器作为普通进程运行，当然宿主机上也会存在一些系统和运维程序。在虚拟处理器运行的同时，如果系统和运维程序也同时运行，将会对虚拟处理器资源造成争抢，并且如果系统和运维程序运行于内核态，不主动出让资源，则会导致出现虚拟处理器单次长时间抖动现象。现有技术通常使用开启内核抢占来解决资源争夺的问题，但内核抢占只是提供了更多的抢占时间点，并不能实现任意时刻
25 的资源抢占，另外，内核抢占的实施还会带来整个内核的改动，进而引入更多的稳定性和安全性挑战。

考虑到上述问题，在该实施方式中，提出一种进程处理方法，该方法通过虚拟机监视器在处理中断事件时检测内核进程的阻塞时间，来判断是否需要对于虚拟处理器进行
30 中间启动，并在满足预设条件时，对于虚拟处理器进行中间启动。该技术方案能够实现

在软中断上下文中对于虚拟处理器进行补偿运行，从而在保证数据和资源的稳定性和安全性的前提下，避免了虚拟处理器出现长时间阻塞和单次长时间抖动现象。

在该实施方式中，首先对于虚拟机监视器的内核进程和虚拟处理器进行初始化；当所述虚拟机监视器的内核进程中断事件被触发且满足预设条件时，认为需要对于所述虚拟处理器进行补偿启动，但为了不致对于内核进程中产生不利影响，所述虚拟处理器的补偿启动只持续预设时间段；在所述预设时间段结束后，所述虚拟机监视器转而处理中断队列中的下一中断事件。

在本发明一实施方式中，所述步骤 S1101 中对于虚拟机监视器的内核进程进行初始化的部分，可被实施为：

10 创建内核进程；

响应于接收到所述虚拟处理器发送的挂载请求，将所述虚拟处理器的一个或多个预设运行单元放置于所述内核进程对应的运行队列中，向所述虚拟处理器发送挂载成功消息；

响应于接收到所述虚拟处理器发送的内核进程唤醒请求，启动所述内核进程。

15 为了保证内核进程的正常工作的，在该实施方式中，在内核进程开始工作之前，还需对其执行初始化操作。具体地，首先创建一个或多个内核进程，对于每个内核进程来说，在接收到所述虚拟处理器发送的挂载请求后，将所述虚拟处理器的一个或多个预设运行单元放置于所述内核进程对应的运行队列中，并向所述虚拟处理器发送挂载成功消息，其中，所述挂载请求可包括内核进程标识信息和请求本体信息，以使所述虚拟机监视器根据所述内核进程标识信息将所述虚拟处理器的一个或多个预设运行单元放置于与所述内核进程标识信息对应的内核进程的运行队列中，当然，所述挂载请求也可仅包括请求本体信息，此时，所述虚拟机监视器随机选择一个内核进程来放置所述虚拟处理器的一个或多个预设运行单元；之后在接收到所述虚拟处理器发送的内核进程唤醒请求后，启动所述内核进程运行。

25 在本发明一实施方式中，所述步骤 S1101 中对于虚拟处理器进行初始化的部分，可被实施为：

生成一个或多个预设运行单元；

向所述虚拟机监视器发送挂载请求，以将所述一个或多个预设运行单元放置于所述虚拟机监视器内核进程对应的运行队列中；

30 响应于接收到所述虚拟机监视器发送的挂载成功消息，向所述虚拟机监视器发送内

核进程唤醒请求并进入睡眠状态，以使所述虚拟机监视器启动所述内核进程。

为了保证虚拟处理器的正常工作，在该实施方式中，在虚拟处理器开始工作之前，还需对其执行初始化操作。具体地，在对于虚拟处理器进行初始化时，首先根据所述虚拟处理器运行的最小循环封装生成一个或多个预设运行单元；然后向所述虚拟机监视器发送挂载请求，以将所述一个或多个预设运行单元放置于所述虚拟机监视器内核进程对应的运行队列中由内核进程执行运行操作；在接收到所述虚拟机监视器发送的挂载成功消息后，向所述虚拟机监视器发送内核进程唤醒请求以唤醒所述内核进程，所述虚拟处理器进而进入睡眠状态。

在本发明一实施方式中，如图 12 所示，所述步骤 S1102，即当所述内核进程中中断事件被触发且满足预设条件时，对于所述虚拟处理器进行中间启动的步骤，包括以下步骤 S1201-S1205：

在步骤 S1201 中，响应于中断队列中当前中断事件处理结束，所述虚拟机监视器检测内核进程的阻塞时间；

在步骤 S1202 中，当所述内核进程的阻塞时间超过预设时间阈值时，所述虚拟机监视器触发虚拟处理器软中断事件；

在步骤 S1203 中，响应于所述虚拟处理器软中断事件被触发，所述虚拟机监视器向所述虚拟处理器发送中间启动指令；

在步骤 S1204 中，响应于接收到所述虚拟机监视器发送的中间启动指令，所述虚拟处理器启动中间运行，其中，所述中间运行的运行时长为预设时间段；

在步骤 S1205 中，响应于所述预设时间段结束，所述虚拟处理器停止所述中间运行。

为了避免虚拟处理器出现长时间阻塞和单次长时间抖动现象，在本发明一实施方式中，每当中断队列中的某一中断事件处理结束，虚拟机监视器就检测一下内核进程的阻塞时间，若所述内核进程的阻塞时间超过预设时间阈值，则认为虚拟处理器有可能出现长时间阻塞现象，需要对于虚拟处理器执行补偿运行操作，则触发虚拟处理器软中断事件，之后向所述虚拟处理器发送中间启动指令，虚拟处理器接收到虚拟机监视器发送的中间启动指令后，启动中间运行，其中，所述虚拟处理器中间运行的运行时长为预设时间段；并在所述预设时间段结束后，所述虚拟处理器停止其中间运行。

在本发明一实施方式中，所述步骤 S1201，即响应于中断队列中当前中断事件处理结束，所述虚拟机监视器检测内核进程的阻塞时间之前，还包括响应于检测到中断事件被触发，所述虚拟机监视器处理中断队列中的当前中断事件的步骤，即如图 13 所示，所

述步骤 S1102, 即当所述内核进程中中断事件被触发且满足预设条件时, 对于所述虚拟处理器进行中间启动的步骤, 包括以下步骤 S1301-S1306:

在步骤 S1301 中, 响应于检测到中断事件被触发, 所述虚拟机监视器处理中断队列中的当前中断事件;

5 在步骤 S1302 中, 响应于中断队列中当前中断事件处理结束, 所述虚拟机监视器检测内核进程的阻塞时间;

在步骤 S1303 中, 当所述内核进程的阻塞时间超过预设时间阈值时, 所述虚拟机监视器触发虚拟处理器软中断事件;

在步骤 S1304 中, 响应于所述虚拟处理器软中断事件被触发, 所述虚拟机监视器向
10 所述虚拟处理器发送中间启动指令;

在步骤 S1305 中, 响应于接收到所述虚拟机监视器发送的中间启动指令, 所述虚拟处理器启动中间运行, 其中, 所述中间运行的运行时长为预设时间段;

在步骤 S1306 中, 响应于所述预设时间段结束, 所述虚拟处理器停止所述中间运行。

上文提及, 所述中断队列中有可能存在一个或多个中断事件, 因此, 在该实施方式
15 中, 在检测到中断事件被触发后, 所述虚拟机监视器按照当前时间以及中断队列中中断事件的排列顺序来处理所述中断队列中的当前中断事件, 并在所述当前中断事件处理结束之后, 再检测内核进程由于中断事件的处理而产生的阻塞时间。

在本发明一实施方式中, 所述步骤 S1301, 即响应于检测到中断事件被触发, 所述虚拟机监视器处理中断队列中的当前中断事件之前, 还包括所述虚拟机监视器获取与所
20 述内核进程对应的运行队列, 并按照队列内预设运行单元排列顺序以所述虚拟处理器的预设运行单元为单位运行, 直至接收到进程调度命令的步骤, 即如图 14 所示, 所述步骤 S1102, 即当所述内核进程中中断事件被触发且满足预设条件时, 对于所述虚拟处理器进行中间启动的步骤, 包括以下步骤 S1401-S1407:

在步骤 S1401 中, 所述虚拟机监视器获取与所述内核进程对应的运行队列, 并按照
25 队列内预设运行单元排列顺序以所述虚拟处理器的预设运行单元为单位运行, 直至接收到进程调度命令, 其中, 所述运行队列中放置有一个或多个虚拟处理器预设运行单元;

在步骤 S1402 中, 响应于检测到中断事件被触发, 所述虚拟机监视器处理中断队列中的当前中断事件;

在步骤 S1403 中, 响应于中断队列中当前中断事件处理结束, 所述虚拟机监视器检
30 测内核进程的阻塞时间;

在步骤 S1404 中，当所述内核进程的阻塞时间超过预设时间阈值时，所述虚拟机监视器触发虚拟处理器软中断事件；

在步骤 S1405 中，响应于所述虚拟处理器软中断事件被触发，所述虚拟机监视器向所述虚拟处理器发送中间启动指令；

5 在步骤 S1406 中，响应于接收到所述虚拟机监视器发送的中间启动指令，所述虚拟处理器启动中间运行，其中，所述中间运行的运行时长为预设时间段；

在步骤 S1407 中，响应于所述预设时间段结束，所述虚拟处理器停止所述中间运行。

为了虚拟处理器数据处理的完整性，在该实施方式中，在所述虚拟机监视器运行虚拟处理器数据时，以所述虚拟处理器的预设运行单元为单位运行，即所述虚拟机监视器
10 首先获取、确定与所述内核进程对应的运行队列，并按照队列内预设运行单元排列顺序以所述虚拟处理器的预设运行单元为单位运行，直至接收到进程调度命令停止所述虚拟处理器的运行。其中，所述运行队列中放置有一个或多个虚拟处理器预设运行单元。在本发明一实施方式中，所述虚拟处理器预设运行单元可以由所述虚拟处理器运行的最小循环封装而成。

15 在本发明一实施方式中，所述方法还包括响应于接收到预设运行请求，所述虚拟机监视器对于所述虚拟处理器进行中间唤醒的步骤，即如图 15 所示，所述进程处理方法包括以下步骤 S1501-S1504：

在步骤 S1501 中，对于虚拟机监视器的内核进程和虚拟处理器进行初始化；

20 在步骤 S1502 中，当所述内核进程中断事件被触发且满足预设条件时，对于所述虚拟处理器进行中间启动，以使所述虚拟处理器在预设时间段内运行；

在步骤 S1503 中，响应于所述预设时间段结束，所述虚拟机监视器处理中断队列中的下一中断事件；

在步骤 S1504 中，响应于接收到预设运行请求，所述虚拟机监视器对于所述虚拟处理器进行中间唤醒。

25 在该实施方式中，若所述虚拟机监视器接收到其无法完成、只能由虚拟处理器来完成的运行请求，比如基于用户态的运行请求等，则需唤醒所述虚拟处理器进入运行状态以处理所述预设运行请求。

在本发明一实施方式中，如图 16 所示，所述步骤 S1504，即响应于接收到预设运行请求，所述虚拟机监视器对于所述虚拟处理器进行中间唤醒的步骤，包括以下步骤
30 S1601-S1605：

在步骤 S1601 中，响应于接收到预设运行请求，所述虚拟机监视器向所述虚拟处理器发送中间唤醒请求，其中，所述中间唤醒请求携带有所述预设运行请求信息；

在步骤 S1602 中，响应于接收到所述虚拟机监视器发送的中间唤醒请求，所述虚拟处理器由睡眠状态进入中间运行状态，并向所述虚拟机监视器发送唤醒成功消息；

5 在步骤 S1603 中，所述虚拟机监视器在接收到所述虚拟处理器发送的唤醒成功消息后进行睡眠状态；

在步骤 S1604 中，所述虚拟处理器运行所述预设运行请求，并在所述预设运行请求处理结束后，向所述虚拟机监视器发送预设运行请求处理结束消息，进入睡眠状态；

10 在步骤 S1605 中，响应于接收到所述虚拟处理器发送的预设运行请求处理结束消息，所述虚拟机监视器进入运行状态。

当所述虚拟机监视器接收到其无法完成、只能由虚拟处理器来完成的运行请求时，需向所述虚拟处理器发送携带有所述预设运行请求信息的中间唤醒请求；所述虚拟处理器在接收到所述虚拟机监视器发送的中间唤醒请求后，其由睡眠状态进入中间运行状态，并向所述虚拟机监视器发送唤醒成功消息；所述虚拟机监视器在接收到所述虚拟处理器
15 发送的唤醒成功消息后进入睡眠状态，转由所述虚拟处理器来完成所述预设运行请求；所述虚拟处理器运行所述预设运行请求，并在所述预设运行请求处理结束后，向所述虚拟机监视器发送预设运行请求处理结束消息，进入睡眠状态；所述虚拟机监视器在接收到所述虚拟处理器发送的预设运行请求处理结束消息后，恢复进入运行状态。

下述为本发明装置实施例，可以用于执行本发明方法实施例。

20 图 17 示出根据本发明一实施方式的进程处理装置的结构框图，该装置可以通过软件、硬件或者两者的结合实现成为电子设备的部分或者全部，比如可实现为虚拟机监视器。如图 17 所示，所述进程处理装置包括：

检测模块 1701，被配置为响应于中断队列中当前中断事件处理结束，检测内核进程的阻塞时间；

25 触发模块 1702，被配置为当所述内核进程的阻塞时间超过预设时间阈值时，触发虚拟处理器软中断事件；

第一发送模块 1703，被配置为响应于所述虚拟处理器软中断事件被触发，向所述虚拟处理器发送中间启动指令，以使所述虚拟处理器在预设时间段内运行。

30 上文提及，随着数据技术的发展，为了提高数据的使用效率和安全性，虚拟机的概念应运而生，虚拟机是指通过软件模拟的具有完整硬件系统功能的、运行在一个完全隔

离环境中的完整计算机系统。在 KVM 架构下，虚拟处理器作为普通进程运行，当然宿主主机上也会存在一些系统和运维程序。在虚拟处理器运行的同时，如果系统和运维程序也同时运行，将会对虚拟处理器资源造成争抢，并且如果系统和运维程序运行于内核态，不主动出让资源，则会导致出现虚拟处理器单次长时间抖动现象。现有技术通常使用开
5 启内核抢占来解决资源争夺的问题，但内核抢占只是提供了更多的抢占时间点，并不能实现任意时刻的资源抢占，另外，内核抢占的实施还会带来整个内核的改动，进而引入更多的稳定性和安全性挑战。

考虑到上述问题，在该实施方式中，提出一种进程处理装置，该装置通过在处理中断事件时检测内核进程的阻塞时间，来判断是否需要对于虚拟处理器进行中间启动，并
10 在满足预设条件时，对于虚拟处理器进行中间启动。该技术方案能够实现在软中断上下文中对于虚拟处理器进行补偿运行，从而在保证数据和资源的稳定性和安全性的前提下，避免了虚拟处理器出现长时间阻塞和单次长时间抖动现象。

在本发明一实施方式中，所述中断队列指的是虚拟机监视器获取的中断事件组成的队列，处于所述中断队列中的中断事件通常被按照在队列中的顺序顺次执行。

为了避免虚拟处理器出现长时间阻塞和单次长时间抖动现象，在本发明一实施方式中，每当中断队列中的某一中断事件处理结束，检测模块 1701 就检测一下内核进程的阻塞时间，若所述内核进程的阻塞时间超过预设时间阈值，则认为虚拟处理器有可能出现长时间阻塞现象，需要对于虚拟处理器执行补偿运行操作，则触发模块 1702 触发虚拟处理器软中断事件，之后第一发送模块 1703 向所述虚拟处理器发送中间启动指令，以使所
20 述虚拟处理器在预设时间段内运行，其中，所述预设时间段指的是虚拟处理器执行补偿运行的时长，其值可根据实际应用的需要进行设置，本公开对其不作具体限定，比如可以将所述预设时间段设置为 0.1-0.5ms 等等，例如，若所述预设时间段设置为 0.1ms，则意味着所述虚拟处理器补偿运行 0.1ms。

其中，所述预设时间阈值用于判断所述内核进程的阻塞时间长短，其值可根据实际应用的需要进行设置，本公开对其不作具体限定。
25

在本发明一实施方式中，所述检测模块 1701 之前，还包括响应于检测到中断事件被触发，处理中断队列中的当前中断事件的部分，即如图 18 所示，所述进程处理装置包括：

第一处理模块 1801，被配置为响应于检测到中断事件被触发，处理中断队列中的当前中断事件；

30 检测模块 1802，被配置为响应于中断队列中当前中断事件处理结束，检测内核进程

的阻塞时间；

触发模块 1803，被配置为当所述内核进程的阻塞时间超过预设时间阈值时，触发虚拟处理器软中断事件；

5 第一发送模块 1804，被配置为响应于所述虚拟处理器软中断事件被触发，向所述虚拟处理器发送中间启动指令，以使所述虚拟处理器在预设时间段内运行。

上文提及，所述中断队列中有可能存在一个或多个中断事件，因此，在该实施方式中，在检测到中断事件被触发后，第一处理模块 1801 按照当前时间以及中断队列中中断事件的排列顺序来处理所述中断队列中的当前中断事件，并在所述当前中断事件处理结束之后，检测模块 1802 再检测内核进程由于中断事件的处理而产生的阻塞时间。

10 在本发明一实施方式中，所述第一处理模块 1801 之前，还包括获取与所述内核进程对应的运行队列，并按照队列内预设运行单元排列顺序以所述虚拟处理器的预设运行单元为单位运行，直至接收到进程调度命令的部分，即如图 19 所示，所述进程处理装置包括：

15 第一运行模块 1901，被配置为获取与所述内核进程对应的运行队列，并按照队列内预设运行单元排列顺序以所述虚拟处理器的预设运行单元为单位运行，直至接收到进程调度命令，其中，所述运行队列中放置有一个或多个虚拟处理器预设运行单元；

第一处理模块 1902，被配置为响应于检测到中断事件被触发，处理中断队列中的当前中断事件；

20 检测模块 1903，被配置为响应于中断队列中当前中断事件处理结束，检测内核进程的阻塞时间；

触发模块 1904，被配置为当所述内核进程的阻塞时间超过预设时间阈值时，触发虚拟处理器软中断事件；

第一发送模块 1905，被配置为响应于所述虚拟处理器软中断事件被触发，向所述虚拟处理器发送中间启动指令，以使所述虚拟处理器在预设时间段内运行。

25 为了虚拟处理器数据处理的完整性，在该实施方式中，第一运行模块 1901 在运行虚拟处理器数据时，以所述虚拟处理器的预设运行单元为单位运行，即首先获取、确定与所述内核进程对应的运行队列，并按照队列内预设运行单元排列顺序以所述虚拟处理器的预设运行单元为单位运行，直至接收到进程调度命令停止所述虚拟处理器的运行。其中，所述运行队列中放置有一个或多个虚拟处理器预设运行单元。在本发明一实施方式中，所述虚拟处理器预设运行单元可以由所述虚拟处理器运行的最小循环封装而成。

30

在本发明一实施方式中，所述虚拟机监视器中可设置一个内核进程也可以设置多个内核进程，其中，所述内核进程的数量可由物理处理器的数量来决定，比如，存在几个物理处理器就设置几个内核进程。其中，每个内核进程负责维护一个运行队列。

在本发明一实施方式中，所述第一运行模块 1901 之前，还包括对于内核进程进行初始化的部分，即如图 20 所示，所述进程处理装置包括：

第一初始化模块 2001，被配置为对于内核进程进行初始化；

第一运行模块 2002，被配置为获取与所述内核进程对应的运行队列，并按照队列内预设运行单元排列顺序以所述虚拟处理器的预设运行单元为单位运行，直至接收到进程调度命令，其中，所述运行队列中放置有一个或多个虚拟处理器预设运行单元；

10 第一处理模块 2003，被配置为响应于检测到中断事件被触发，处理中断队列中的当前中断事件；

检测模块 2004，被配置为响应于中断队列中当前中断事件处理结束，检测内核进程的阻塞时间；

15 触发模块 2005，被配置为当所述内核进程的阻塞时间超过预设时间阈值时，触发虚拟处理器软中断事件；

第一发送模块 2006，被配置为响应于所述虚拟处理器软中断事件被触发，向所述虚拟处理器发送中间启动指令，以使所述虚拟处理器在预设时间段内运行。

为了保证内核进程的正常工作的，在该实施方式中，在内核进程开始工作之前，还需设置第一初始化模块 2001 对其执行初始化操作。

20 在本发明一实施方式中，如图 21 所示，所述第一初始化模块 2001 包括：

创建子模块 2101，被配置为创建内核进程；

挂载子模块 2102，被配置为响应于接收到所述虚拟处理器发送的挂载请求，将所述虚拟处理器的一个或多个预设运行单元放置于所述内核进程对应的运行队列中，向所述虚拟处理器发送挂载成功消息；

25 第一启动子模块 2103，被配置为响应于接收到所述虚拟处理器发送的内核进程唤醒请求，启动所述内核进程。

在该实施方式中，在对于内核进程进行初始化时，创建子模块 2101 首先创建一个或多个内核进程，对于每个内核进程来说，在接收到所述虚拟处理器发送的挂载请求后，挂载子模块 2102 将所述虚拟处理器的一个或多个预设运行单元放置于所述内核进程对应的运行队列中，并向所述虚拟处理器发送挂载成功消息，其中，所述挂载请求可包括

30

内核进程标识信息和请求本体信息，以使所述虚拟机监视器根据所述内核进程标识信息将所述虚拟处理器的一个或多个预设运行单元放置于与所述内核进程标识信息对应的内核进程的运行队列中，当然，所述挂载请求也可仅包括请求本体信息，此时，所述虚拟机监视器随机选择一个内核进程来放置所述虚拟处理器的一个或多个预设运行单元；之后第一启动子模块 2103 在接收到所述虚拟处理器发送的内核进程唤醒请求后，启动所述内核进程运行。

在本发明一实施方式中，所述装置还包括与所述虚拟处理器交互以执行预设运行请求的部分，即如图 22 所示，所述进程处理装置包括：

检测模块 2201，被配置为响应于中断队列中当前中断事件处理结束，检测内核进程的阻塞时间；

触发模块 2202，被配置为当所述内核进程的阻塞时间超过预设时间阈值时，触发虚拟处理器软中断事件；

第一发送模块 2203，被配置为响应于所述虚拟处理器软中断事件被触发，向所述虚拟处理器发送中间启动指令，以使所述虚拟处理器在预设时间段内运行；

第二发送模块 2204，被配置为响应于接收到预设运行请求，向所述虚拟处理器发送中间唤醒请求，并在接收到所述虚拟处理器发送的唤醒成功消息后进入睡眠状态，其中，所述中间唤醒请求携带有所述预设运行请求信息；

第二运行模块 2205，被配置为响应于接收到所述虚拟处理器发送的预设运行请求处理结束消息，进入运行状态。

若所述虚拟机监视器接收到其无法完成、只能由虚拟处理器来完成的运行请求，比如基于用户态的运行请求等，则需设置第二发送模块 2204 向所述虚拟处理器发送携带有所述预设运行请求信息的中间唤醒请求，并在接收到所述虚拟处理器发送的唤醒成功消息后进入睡眠状态，转由所述虚拟处理器来完成所述预设运行请求，并在接收到所述虚拟处理器发送的预设运行请求处理结束消息后，借助第二运行模块 2205 再次进入运行状态。

在本发明一实施方式中，所述装置还包括响应于所述预设时间段结束，处理所述中断队列中的下一中断事件的部分，即如图 23 所示，所述进程处理装置包括：

检测模块 2301，被配置为响应于中断队列中当前中断事件处理结束，检测内核进程的阻塞时间；

触发模块 2302，被配置为当所述内核进程的阻塞时间超过预设时间阈值时，触发虚

拟处理器软中断事件；

第一发送模块 2303，被配置为响应于所述虚拟处理器软中断事件被触发，向所述虚拟处理器发送中间启动指令，以使所述虚拟处理器在预设时间段内运行；

第二处理模块 2304，被配置为响应于所述预设时间段结束，处理所述中断队列中的下一中断事件。

在该实施方式中，第二处理模块 2304 在所述虚拟处理器执行中间启动预设时间段，即中间启动结束后，所述虚拟机监视器继续处理所述中断队列中的下一中断事件，并在处理完下一中断事件之后，检测内核进程的阻塞时间。若所述中断队列中不存在下一中断事件，则所述虚拟机监视器继续处理其他进程。

图 24 示出根据本发明一实施方式的进程处理装置的结构框图，该装置可以通过软件、硬件或者两者的结合实现成为电子设备的部分或者全部，比如可实现为虚拟处理器。如图 24 所示，所述进程处理装置包括：

第二初始化模块 2401，被配置为对于虚拟处理器进行初始化；

第一启动模块 2402，被配置为响应于接收到虚拟机监视器发送的中间启动指令，启动中间运行，其中，所述中间运行的运行时长为预设时间段；

停止模块 2403，被配置为响应于所述预设时间段结束，停止所述中间运行。

上文提及，在 KVM 架构下，虚拟处理器作为普通进程运行，当然宿主机上也会存在一些系统和运维程序。在虚拟处理器运行的同时，如果系统和运维程序也同时运行，将会对虚拟处理器资源造成争抢，并且如果系统和运维程序运行于内核态，不主动出让资源，则会导致出现虚拟处理器单次长时间抖动现象。现有技术通常使用开启内核抢占来解决资源争夺的问题，但内核抢占只是提供了更多的抢占时间点，并不能实现任意时刻的资源抢占，另外，内核抢占的实施还会带来整个内核的改动，进而引入更多的稳定性和安全性挑战。

考虑到上述问题，在该实施方式中，提出一种进程处理装置，该装置通过虚拟机监视器在处理中断事件时检测内核进程的阻塞时间，来判断是否需要对于虚拟处理器进行中间启动，并在满足预设条件时，对于虚拟处理器进行中间启动。该技术方案能够在软中断上下文中对于虚拟处理器进行补偿运行，从而在保证数据和资源的稳定性和安全性的前提下，避免了虚拟处理器出现长时间阻塞和单次长时间抖动现象。

为了避免虚拟处理器出现长时间阻塞和单次长时间抖动现象，在该实施方式中，在第二初始化模块 2401 对于虚拟处理器进行初始化，挂载在所述虚拟机监视器上之后，若

第一启动模块 2402 接收到虚拟机监视器发送的中间启动指令，则启动中间运行，其中，所述虚拟处理器中间运行的运行时长为预设时间段；停止模块 2403 在所述预设时间段结束后，停止所述中间运行。

其中，所述预设时间段指的是虚拟处理器执行补偿运行的时长，其值可根据实际应用的需要进行设置，本公开对其不作具体限定，比如可以将所述预设时间段设置为 0.1-0.5ms 等等，例如，若所述预设时间段设置为 0.1ms，则意味着所述虚拟处理器补偿运行 0.1ms。

在本发明一实施方式中，如图 25 所示，所述第二初始化模块 2401 包括：

生成子模块 2501，被配置为生成一个或多个预设运行单元；

10 第一发送子模块 2502，被配置为向所述虚拟机监视器发送挂载请求，以将所述一个或多个预设运行单元放置于所述虚拟机监视器内核进程对应的运行队列中；

第二发送子模块 2503，被配置为响应于接收到所述虚拟机监视器发送的挂载成功消息，向所述虚拟机监视器发送内核进程唤醒请求并进入睡眠状态，以使所述虚拟机监视器启动所述内核进程。

15 在该实施方式中，在对于虚拟处理器进行初始化时，生成子模块 2501 首先根据所述虚拟处理器运行的最小循环封装生成一个或多个预设运行单元；第一发送子模块 2502 然后向所述虚拟机监视器发送挂载请求，以将所述一个或多个预设运行单元放置于所述虚拟机监视器内核进程对应的运行队列中由内核进程执行运行操作；第二发送子模块 2503 在接收到所述虚拟机监视器发送的挂载成功消息后，向所述虚拟机监视器发送内核进程
20 唤醒请求以唤醒所述内核进程，所述虚拟处理器进而进入睡眠状态。

在本发明一实施方式中，所述装置还包括与所述虚拟机监视器交互以执行预设运行请求的部分，即如图 26 所示，所述进程处理装置包括：

第二初始化模块 2601，被配置为对于虚拟处理器进行初始化；

25 第一启动模块 2602，被配置为响应于接收到虚拟机监视器发送的中间启动指令，启动中间运行，其中，所述中间运行的运行时长为预设时间段；

停止模块 2603，被配置为响应于所述预设时间段结束，停止所述中间运行；

第三发送模块 2604，被配置为响应于接收到所述虚拟机监视器发送的中间唤醒请求，由睡眠状态进入中间运行状态，并向所述虚拟机监视器发送唤醒成功消息，其中，所述中间唤醒请求携带有所述预设运行请求信息；

30 第三运行模块 2605，被配置为运行所述预设运行请求，并在所述预设运行请求处理

结束后，向所述虚拟机监视器发送预设运行请求处理结束消息，进入睡眠状态。

若所述虚拟机监视器接收到其无法完成、只能由虚拟处理器来完成的运行请求，比如基于用户态的运行请求等，则需设置第三发送模块 2604 向所述虚拟处理器发送携带有所述预设运行请求信息的中间唤醒请求以唤醒所述虚拟处理器，此时，所述虚拟处理器
5 接收到所述虚拟机监视器发送的中间唤醒请求后，由睡眠状态进入中间运行状态，并向所述虚拟机监视器发送唤醒成功消息，然后所述虚拟处理器的第三运行模块 2605 运行所述预设运行请求，并在所述预设运行请求处理结束后，向所述虚拟机监视器发送预设运行请求处理结束消息以重新唤醒所述虚拟机监视器，而所述虚拟处理器则再次进入睡眠状态。

10 图 27 示出根据本发明一实施方式的进程处理装置的结构框图，该装置可以通过软件、硬件或者两者的结合实现成为电子设备的部分或者全部，比如可实现为进程处理系统。如图 27 所示，所述进程处理装置包括：

第三初始化模块 2701，被配置为对于虚拟机监视器的内核进程和虚拟处理器进行初始化；

15 第二启动模块 2702，被配置为当所述内核进程中断事件被触发且满足预设条件时，对于所述虚拟处理器进行中间启动，以使所述虚拟处理器在预设时间段内运行；

第三处理模块 2703，被配置为响应于所述预设时间段结束，处理中断队列中的下一中断事件。

上文提及，在 KVM 架构下，虚拟处理器作为普通进程运行，当然宿主机上也会存在
20 一些系统和运维程序。在虚拟处理器运行的同时，如果系统和运维程序也同时运行，将会对虚拟处理器资源造成争抢，并且如果系统和运维程序运行于内核态，不主动出让资源，则会导致出现虚拟处理器单次长时间抖动现象。现有技术通常使用开启内核抢占来解决资源争夺的问题，但内核抢占只是提供了更多的抢占时间点，并不能实现任意时刻的资源抢占，另外，内核抢占的实施还会带来整个内核的改动，进而引入更多的稳定性和
25 和安全性挑战。

考虑到上述问题，在该实施方式中，提出一种进程处理装置，该装置通过虚拟机监视器在处理中断事件时检测内核进程的阻塞时间，来判断是否需要对于虚拟处理器进行中间启动，并在满足预设条件时，对于虚拟处理器进行中间启动。该技术方案能够在软中断上下文中对于虚拟处理器进行补偿运行，从而在保证数据和资源的稳定性和安
30 全性的前提下，避免了虚拟处理器出现长时间阻塞和单次长时间抖动现象。

在该实施方式中，首先第三初始化模块 2701 对于虚拟机监视器的内核进程和虚拟处理器进行初始化；当所述虚拟机监视器的内核进程中断事件被触发且满足预设条件时，认为需要对于所述虚拟处理器进行补偿启动，但为了不致对于内核进程中产生不利影响，所述虚拟处理器的补偿启动只持续预设时间段；在所述预设时间段结束后，所述虚拟机监视器的第三处理模块 2703 转而处理中断队列中的下一中断事件。

在本发明一实施方式中，所述第三初始化模块 2701 中对于虚拟机监视器的内核进程进行初始化的部分，可被配置为：

创建内核进程；

响应于接收到所述虚拟处理器发送的挂载请求，将所述虚拟处理器的一个或多个预设运行单元放置于所述内核进程对应的运行队列中，向所述虚拟处理器发送挂载成功消息；

响应于接收到所述虚拟处理器发送的内核进程唤醒请求，启动所述内核进程。

为了保证内核进程的正常工作的，在该实施方式中，在内核进程开始工作之前，还需对其执行初始化操作。具体地，首先创建一个或多个内核进程，对于每个内核进程来说，在接收到所述虚拟处理器发送的挂载请求后，将所述虚拟处理器的一个或多个预设运行单元放置于所述内核进程对应的运行队列中，并向所述虚拟处理器发送挂载成功消息，其中，所述挂载请求可包括内核进程标识信息和请求本体信息，以使所述虚拟机监视器根据所述内核进程标识信息将所述虚拟处理器的一个或多个预设运行单元放置于与所述内核进程标识信息对应的内核进程的运行队列中，当然，所述挂载请求也可仅包括请求本体信息，此时，所述虚拟机监视器随机选择一个内核进程来放置所述虚拟处理器的一个或多个预设运行单元；之后在接收到所述虚拟处理器发送的内核进程唤醒请求后，启动所述内核进程运行。

在本发明一实施方式中，所述第三初始化模块 2701 中对于虚拟处理器进行初始化的部分，可被配置为：

生成一个或多个预设运行单元；

向所述虚拟机监视器发送挂载请求，以将所述一个或多个预设运行单元放置于所述虚拟机监视器内核进程对应的运行队列中；

响应于接收到所述虚拟机监视器发送的挂载成功消息，向所述虚拟机监视器发送内核进程唤醒请求并进入睡眠状态，以使所述虚拟机监视器启动所述内核进程。

为了保证虚拟处理器的正常工作，在该实施方式中，在虚拟处理器开始工作之前，

还需对其执行初始化操作。具体地，在对于虚拟处理器进行初始化时，首先根据所述虚拟处理器运行的最小循环封装生成一个或多个预设运行单元；然后向所述虚拟机监视器发送挂载请求，以将所述一个或多个预设运行单元放置于所述虚拟机监视器内核进程对应的运行队列中由内核进程执行运行操作；在接收到所述虚拟机监视器发送的挂载成功消息后，向所述虚拟机监视器发送内核进程唤醒请求以唤醒所述内核进程，所述虚拟处理器进而进入睡眠状态。

在本发明一实施方式中，如图 28 所示，所述第二启动模块 2702 包括：

检测子模块 2801，被配置为响应于中断队列中当前中断事件处理结束，检测内核进程的阻塞时间；

10 触发子模块 2802，被配置为当所述内核进程的阻塞时间超过预设时间阈值时，触发虚拟处理器软中断事件；

第三发送子模块 2803，被配置为响应于所述虚拟处理器软中断事件被触发，向所述虚拟处理器发送中间启动指令；

15 第二启动子模块 2804，被配置为响应于接收到所述虚拟机监视器发送的中间启动指令，启动中间运行，其中，所述中间运行的运行时长为预设时间段；

停止子模块 2805，被配置为响应于所述预设时间段结束，停止所述中间运行。

为了避免虚拟处理器出现长时间阻塞和单次长时间抖动现象，在本发明一实施方式中，每当中断队列中的某一中断事件处理结束，虚拟机监视器的检测子模块 2801 就检测一下内核进程的阻塞时间，若所述内核进程的阻塞时间超过预设时间阈值，则认为虚拟处理器有可能出现长时间阻塞现象，需要对于虚拟处理器执行补偿运行操作，则触发子模块 2802 触发虚拟处理器软中断事件，之后第三发送子模块 2803 向所述虚拟处理器发送中间启动指令，虚拟处理器的第二启动子模块 2804 接收到虚拟机监视器发送的中间启动指令后，启动中间运行，其中，所述虚拟处理器中间运行的运行时长为预设时间段；并在所述预设时间段结束后，所述虚拟处理器的停止子模块 2805 停止其中间运行。

25 在本发明一实施方式中，所述检测子模块 2801 之前，还包括响应于检测到中断事件被触发，所述虚拟机监视器处理中断队列中的当前中断事件的部分，即如图 29 所示，所述第二启动模块 2702 包括：

处理子模块 2901，被配置为响应于检测到中断事件被触发，处理中断队列中的当前中断事件；

30 检测子模块 2902，被配置为响应于中断队列中当前中断事件处理结束，检测内核进

程的阻塞时间；

触发子模块 2903，被配置为当所述内核进程的阻塞时间超过预设时间阈值时，触发虚拟处理器软中断事件；

5 第三发送子模块 2904，被配置为响应于所述虚拟处理器软中断事件被触发，向所述虚拟处理器发送中间启动指令；

第二启动子模块 2905，被配置为响应于接收到所述虚拟机监视器发送的中间启动指令，启动中间运行，其中，所述中间运行的运行时长为预设时间段；

停止子模块 2906，被配置为响应于所述预设时间段结束，停止所述中间运行。

10 上文提及，所述中断队列中有可能存在一个或多个中断事件，因此，在该实施方式中，在检测到中断事件被触发后，所述虚拟机监视器的处理子模块 2901 按照当前时间以及中断队列中中断事件的排列顺序来处理所述中断队列中的当前中断事件，并在所述当前中断事件处理结束之后，再由检测子模块 2902 检测内核进程由于中断事件的处理而产生的阻塞时间。

15 在本发明一实施方式中，所述处理子模块 2901 之前，还包括所述虚拟机监视器获取与所述内核进程对应的运行队列，并按照队列内预设运行单元排列顺序以所述虚拟处理器的预设运行单元为单位运行，直至接收到进程调度命令的部分，即如图 30 所示，所述第二启动模块 2702 包括：

20 第一运行子模块 3001，被配置为获取与所述内核进程对应的运行队列，并按照队列内预设运行单元排列顺序以所述虚拟处理器的预设运行单元为单位运行，直至接收到进程调度命令，其中，所述运行队列中放置有一个或多个虚拟处理器预设运行单元；

处理子模块 3002，被配置为响应于检测到中断事件被触发，处理中断队列中的当前中断事件；

检测子模块 3003，被配置为响应于中断队列中当前中断事件处理结束，检测内核进程的阻塞时间；

25 触发子模块 3004，被配置为当所述内核进程的阻塞时间超过预设时间阈值时，触发虚拟处理器软中断事件；

第三发送子模块 3005，被配置为响应于所述虚拟处理器软中断事件被触发，向所述虚拟处理器发送中间启动指令；

30 第二启动子模块 3006，被配置为响应于接收到所述虚拟机监视器发送的中间启动指令，启动中间运行，其中，所述中间运行的运行时长为预设时间段；

停止子模块 3007，被配置为响应于所述预设时间段结束，停止所述中间运行。

为了虚拟处理器数据处理的完整性，在该实施方式中，在所述虚拟机监视器运行虚拟处理器数据时，所述虚拟机监视器的第一运行子模块 3001 以所述虚拟处理器的预设运行单元为单位运行，即所述虚拟机监视器的第一运行子模块 3001 首先获取、确定与所述
5 内核进程对应的运行队列，并按照队列内预设运行单元排列顺序以所述虚拟处理器的预设运行单元为单位运行，直至接收到进程调度命令停止所述虚拟处理器的运行。其中，所述运行队列中放置有一个或多个虚拟处理器预设运行单元。在本发明一实施方式中，所述虚拟处理器预设运行单元可以由所述虚拟处理器运行的最小循环封装而成。

在本发明一实施方式中，所述装置还包括响应于接收到预设运行请求，所述虚拟机
10 监视器对于所述虚拟处理器进行中间唤醒的部分，即如图 31 所示，所述进程处理装置包括：

第三初始化模块 3101，被配置为对于虚拟机监视器的内核进程和虚拟处理器进行初始化；

第二启动模块 3102，被配置为当所述内核进程中断事件被触发且满足预设条件时，
15 对于所述虚拟处理器进行中间启动，以使所述虚拟处理器在预设时间段内运行；

第三处理模块 3103，被配置为响应于所述预设时间段结束，处理中断队列中的下一中断事件；

唤醒模块 3104，被配置为响应于接收到预设运行请求，对于所述虚拟处理器进行中间唤醒。

20 在该实施方式中，若所述虚拟机监视器接收到其无法完成、只能由虚拟处理器来完成的运行请求，比如基于用户态的运行请求等，则需设置唤醒模块 3104 唤醒所述虚拟处理器进入运行状态以处理所述预设运行请求。

在本发明一实施方式中，如图 32 所示，所述唤醒模块 3104 包括：

第四发送子模块 3201，被配置为响应于接收到预设运行请求，向所述虚拟处理器发送中间唤醒请求，其中，所述中间唤醒请求携带有所述预设运行请求信息；
25

第五发送子模块 3202，被配置为响应于接收到所述虚拟机监视器发送的中间唤醒请求，由睡眠状态进入中间运行状态，并向所述虚拟机监视器发送唤醒成功消息；

第一状态改变子模块 3203，被配置为在接收到所述虚拟处理器发送的唤醒成功消息后进入睡眠状态；

30 第二运行子模块 3204，被配置为运行所述预设运行请求，并在所述预设运行请求处

理结束后，向所述虚拟机监视器发送预设运行请求处理结束消息，进入睡眠状态；

第二状态改变子模块 3205，被配置为响应于接收到所述虚拟处理器发送的预设运行请求处理结束消息，进入运行状态。

当所述虚拟机监视器接收到其无法完成、只能由虚拟处理器来完成的运行请求时，
5 第四发送子模块 3201 需向所述虚拟处理器发送携带有所述预设运行请求信息的中间唤醒请求；所述虚拟处理器的第五发送子模块 3202 在接收到所述虚拟机监视器发送的中间唤醒请求后，其由睡眠状态进入中间运行状态，并向所述虚拟机监视器发送唤醒成功消息；所述虚拟机监视器的第一状态改变子模块 3203 在接收到所述虚拟处理器发送的唤醒成功消息后进入睡眠状态，转由所述虚拟处理器的第二运行子模块 3204 来完成所述预设
10 运行请求；所述虚拟处理器的第二运行子模块 3204 运行所述预设运行请求，并在所述预设运行请求处理结束后，向所述虚拟机监视器发送预设运行请求处理结束消息，进入睡眠状态；所述虚拟机监视器的第二状态改变子模块 3205 在接收到所述虚拟处理器发送的预设运行请求处理结束消息后，恢复进入运行状态。

接下来以一应用场景为例对于本发明技术方案进行进一步的说明，如图 33 所示，在
15 该应用场景中，若确定中断队列中当前中断事件处理结束，检测内核进程的阻塞时间是否超过预设时间阈值，当超过时，触发虚拟处理器软中断事件，使所述虚拟处理器运行 0.1ms，之后继续原进程，若未超过时，不触发虚拟处理器软中断事件继续原进程。

本发明实施例还公开了一种电子设备，图 34 示出根据本发明一实施方式的电子设备的结构框图，如图 34 所示，所述电子设备 3400 包括存储器 3401 和处理器 3402；其中，
20 所述存储器 3401 用于存储一条或多条计算机指令，其中，所述一条或多条计算机指令被所述处理器 3402 执行以实现上述任一方法步骤。

图 35 适于用来实现根据本发明实施方式的进程处理方法的计算机系统的结构示意图。

如图 35 所示，计算机系统 3500 包括处理单元 3501，其可以根据存储在只读存储器
25 (ROM) 3502 中的程序或者从存储部分 3508 加载到随机访问存储器 (RAM) 3503 中的程序而执行上述实施方式中的各种处理。在 RAM3503 中，还存储有系统 3500 操作所需的各种程序和数据。处理单元 3501、ROM3502 以及 RAM3503 通过总线 3504 彼此相连。输入/输出 (I/O) 接口 3505 也连接至总线 3504。

以下部件连接至 I/O 接口 3505：包括键盘、鼠标等的输入部分 3506；包括诸如阴极
30 射线管 (CRT)、液晶显示器 (LCD) 等以及扬声器等的输出部分 3507；包括硬盘等的存

储部分 3508；以及包括诸如 LAN 卡、调制解调器等网络接口卡的通信部分 3509。通信部分 3509 经由诸如因特网的网络执行通信处理。驱动器 3510 也需要连接至 I/O 接口 3505。可拆卸介质 3511，诸如磁盘、光盘、磁光盘、半导体存储器等等，根据需要安装在驱动器 3510 上，以便于从其上读出的计算机程序根据需要被安装入存储部分 3508。

5 其中，所述处理单元 3501 可实现为 CPU、GPU、FPGA、NPU、AI 芯片等处理单元。

特别地，根据本发明的实施方式，上文描述的方法可以被实现为计算机软件程序。例如，本发明的实施方式包括一种计算机程序产品，其包括有形地包含在及其可读介质上的计算机程序，所述计算机程序包含用于执行所述进程处理方法的程序代码。在这样的实施方式中，该计算机程序可以通过通信部分 3509 从网络上被下载和安装，和/或从可拆卸介质 3511 被安装。

附图中的流程图和框图，图示了按照本发明各种实施方式的系统、方法和计算机程序产品的可能实现的体系架构、功能和操作。在这点上，流程图或框图中的每个方框可以代表一个模块、程序段或代码的一部分，所述模块、程序段或代码的一部分包含一个或多个用于实现规定的逻辑功能的可执行指令。也应当注意，在有些作为替换的实现中，15 方框中所标注的功能也可以以不同于附图中所标注的顺序发生。例如，两个接连地表示的方框实际上可以基本并行地执行，它们有时也可以按相反的顺序执行，这依所涉及的功能而定。也要注意，框图和/或流程图中的每个方框、以及框图和/或流程图中的方框的组合，可以用执行规定的功能或操作的专用的基于硬件的系统来实现，或者可以用专用硬件与计算机指令的组合来实现。

20 描述于本发明实施方式中所涉及到的单元或模块可以通过软件的方式实现，也可以通过硬件的方式来实现。所描述的单元或模块也可以设置在处理器中，这些单元或模块的名称在某种情况下并不构成对该单元或模块本身的限定。

作为另一方面，本发明实施例还提供了一种计算机可读存储介质，该计算机可读存储介质可以是上述实施方式中所述装置中所包含的计算机可读存储介质；也可以是单独存在，未装配入设备中的计算机可读存储介质。计算机可读存储介质存储有一个或者一个以上程序，所述程序被一个或者一个以上的处理器用来执行描述于本发明实施例的方法。

以上描述仅为本发明的较佳实施例以及对所运用技术原理的说明。本领域技术人员应当理解，本发明实施例中所涉及的发明范围，并不限于上述技术特征的特定组合而成的技术方案，同时也应涵盖在不脱离所述发明构思的情况下，由上述技术特征或其等同

特征进行任意组合而形成的其它技术方案。例如上述特征与本发明实施例中公开的（但不限于）具有类似功能的技术特征进行互相替换而形成的技术方案。