

(12) 发明专利申请

(10) 申请公布号 CN 102253884 A

(43) 申请公布日 2011. 11. 23

(21) 申请号 201110122635. 5

(22) 申请日 2011. 05. 12

(71) 申请人 中兴通讯股份有限公司

地址 518057 广东省深圳市南山区高新技术产业园科技南路中兴通讯大厦法务部

(72) 发明人 冯海芳

(74) 专利代理机构 深圳市世纪恒程知识产权代理事务所 44287

代理人 胡海国

(51) Int. Cl.

G06F 11/36 (2006. 01)

G06F 9/455 (2006. 01)

H04L 12/56 (2006. 01)

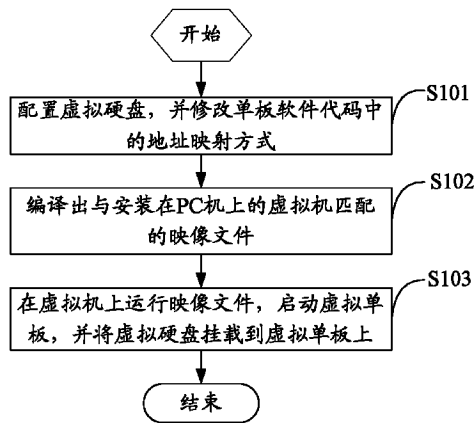
权利要求书 1 页 说明书 6 页 附图 4 页

(54) 发明名称

模拟单板软件运行环境的方法及装置

(57) 摘要

本发明涉及一种模拟单板软件运行环境的方法及装置,其方法包括:配置虚拟硬盘,并修改单板软件代码中的地址映射方式;编译出与安装在PC机上的虚拟机匹配的映像文件;在虚拟机上运行映像文件,启动虚拟单板,并将虚拟硬盘挂载到虚拟单板上。本发明通过配置用于保存固化信息的虚拟硬盘、硬件地址访问的模拟以及编译与虚拟机匹配的映像文件,实现了在PC机上模拟单板软件运行环境,使单板软件在没有硬件环境下能够正常的运行。有助于开发人员在单板硬件开发完成前完成单板软件早期开发及调试,以及在组网测试网元不足时模拟物理网元,从而减少单板的开发周期,降低开发成本。



1. 一种模拟单板软件运行环境的方法,其特征在于,包括:  
配置虚拟硬盘,并修改单板软件代码中的地址映射方式;  
编译出与安装在 PC 机上的虚拟机匹配的映像文件;  
在所述虚拟机上运行所述映像文件,启动所述虚拟单板,并将所述虚拟硬盘挂载到所述虚拟单板上。
2. 根据权利要求 1 所述的方法,其特征在于,所述修改单板软件代码中的地址映射方式的步骤包括:  
申请预定芯片物理地址空间大小的物理地址空间;  
将需要访问的芯片虚拟地址空间映射到所述物理地址空间。
3. 根据权利要求 1 所述的方法,其特征在于,还包括:将现场可编程门阵列 FPGA 加载后的寄存器的值写入文本文件,并将所述文本文件保存在所述虚拟硬盘中;  
在所述虚拟单板启动时,从所述文本文件中读取所述寄存器的值,对所述 FPGA 的寄存器进行初始化。
4. 根据权利要求 1-3 中任一项所述的方法,其特征在于,编译所述映像文件的方式包括:修改所述单板软件使用的板级支持包 BSP 文件。
5. 根据权利要求 4 所述的方法,其特征在于,编译所述映像文件的方式进一步包括:通过宏变量和环境变量控制编译过程进行所述映像文件的编译,若宏变量和环境变量被定义,则按照仿真分支进行编译;若宏变量和环境变量没有被定义,则按照真实的分支进行编译。
6. 一种模拟单板软件运行环境的装置,其特征在于,包括:  
硬盘配置及地址映射模块,用于配置虚拟硬盘,并修改单板软件代码中的地址映射方式;  
映像文件编译模块,用于编译出与安装在 PC 机上的虚拟机匹配的映像文件;  
虚拟单板启动模块,用于在所述虚拟机上运行所述映像文件,启动所述虚拟单板,并将所述虚拟硬盘挂载到所述虚拟单板上。
7. 根据权利要求 6 所述的装置,其特征在于,所述硬盘配置及地址映射模块还用于申请预定芯片物理地址空间大小的物理地址空间;将需要访问的芯片虚拟地址空间映射到所述物理地址空间。
8. 根据权利要求 6 所述的装置,其特征在于,还包括:  
加载文件写入模块,用于将 FPGA 加载后的寄存器的值写入文本文件,并将所述文本文件保存在所述虚拟硬盘中;  
初始化模块,用于在所述虚拟单板启动时,从所述文本文件中读取所述寄存器的值,对所述 FPGA 的寄存器进行初始化。
9. 根据权利要求 6、7 或 8 所述的装置,其特征在于,所述映像文件编译模块编译所述映像文件的方式包括:修改所述单板软件使用的 BSP 文件。
10. 根据权利要求 9 所述的装置,其特征在于,所述映像文件编译模块编译所述映像文件的方式进一步包括:通过宏变量和环境变量控制编译过程进行所述映像文件的编译,若宏变量和环境变量被定义,则按照仿真分支进行编译;若宏变量和环境变量没有被定义,则按照真实的分支进行编译。

## 模拟单板软件运行环境的方法及装置

### 技术领域

[0001] 本发明涉及通讯技术领域,尤其涉及一种模拟单板软件运行环境的方法及装置。

### 背景技术

[0002] 在通讯设备系统中,系统设备由单板或者板卡组成。单板主要由 CPU(Central Processing Unit,中央处理器)、看门狗电路以及外围关键处理芯片组成,根据功能不同,可增加 FPGA(Field-Programmable Gate Array,现场可编程门阵列)或者 DSP(Digital Signal Processing,数字信号处理器)器件。

[0003] 通常情况下,单板的硬件开发周期均比较长,软件开发人员需要等待单板开发完成之后才能进行单板软件的开发和调试。由此增加了单板的开发周期,以及开发成本。

[0004] 而且,在组网过程中,当物理网元不足时,由于无法在仿真模拟环境中运行虚拟单板来模拟网元,组网测试无法及时完成。

### 发明内容

[0005] 本发明的主要目的在于提供一种模拟单板软件运行环境的方法及装置,旨在减少单板的开发周期,并降低单板的开发成本。

[0006] 为了达到上述目的,本发明提出一种模拟单板软件运行环境的方法,包括:

[0007] 配置虚拟硬盘,并修改单板软件代码中的地址映射方式;

[0008] 编译出与安装在 PC 机上的虚拟机匹配的映像文件;

[0009] 在所述虚拟机上运行所述映像文件,启动所述虚拟单板,并将所述虚拟硬盘挂载到所述虚拟单板上。

[0010] 优选地,所述修改单板软件代码中的地址映射方式的步骤包括:

[0011] 申请预定芯片物理地址空间大小的物理地址空间;

[0012] 将需要访问的芯片虚拟地址空间映射到所述物理地址空间。

[0013] 优选地,还包括:将现场可编程门阵列 FPGA 加载后的寄存器的值写入文本文件,并将所述文本文件保存在所述虚拟硬盘中;

[0014] 在所述虚拟单板启动时,从所述文本文件中读取所述寄存器的值,对所述 FPGA 的寄存器进行初始化。

[0015] 优选地,编译所述映像文件的方式包括:修改所述单板软件使用的板级支持包 BSP 文件。

[0016] 优选地,编译所述映像文件的方式进一步包括:通过宏变量和环境变量控制编译过程进行所述映像文件的编译,若宏变量和环境变量被定义,则按照仿真分支进行编译;若宏变量和环境变量没有被定义,则按照真实的分支进行编译。

[0017] 本发明还提出一种模拟单板软件运行环境的装置,包括:

[0018] 硬盘配置及地址映射模块,用于配置虚拟硬盘,并修改单板软件代码中的地址映射方式;

[0019] 映像文件编译模块,用于编译出与安装在 PC 机上的虚拟机匹配的映像文件;

[0020] 虚拟单板启动模块,用于在所述虚拟机上运行所述映像文件,启动所述虚拟单板,并将所述虚拟硬盘挂载到所述虚拟单板上。

[0021] 优选地,所述硬盘配置及地址映射模块还用于申请预定芯片物理地址空间大小的物理地址空间;将需要访问的芯片虚拟地址空间映射到所述物理地址空间。

[0022] 优选地,还包括:

[0023] 加载文件写入模块,用于将 FPGA 加载后的寄存器的值写入文本文件,并将所述文本文件保存在所述虚拟硬盘中;

[0024] 初始化模块,用于在所述虚拟单板启动时,从所述文本文件中读取所述寄存器的值,对所述 FPGA 的寄存器进行初始化。

[0025] 优选地,所述映像文件编译模块编译所述映像文件的方式包括:修改所述单板软件使用的 BSP 文件。

[0026] 优选地,所述映像文件编译模块编译所述映像文件的方式进一步包括:通过宏变量和环境变量控制编译过程进行所述映像文件的编译,若宏变量和环境变量被定义,则按照仿真分支进行编译;若宏变量和环境变量没有被定义,则按照真实的分支进行编译。

[0027] 本发明提出的一种模拟单板软件运行环境的方法及装置,通过配置用于保存固化信息的虚拟硬盘、硬件地址访问的模拟以及编译与虚拟机匹配的映像文件,实现了在 PC 机上模拟单板软件运行环境,为单板软件提供了一个没有硬件支撑的开发和调试环境。该模拟运行环境通过在虚拟机上建立虚拟单板,并对虚拟机中不存在的硬件进行模拟,解决模拟环境中硬件访问的问题,从而使单板软件在没有硬件环境下能够正常的运行。本发明可以帮助开发人员熟悉开发环境,在单板硬件没有开发完成时,帮助单板软件开发人员完成单板软件早期开发和调试,以及在组网测试网元不足时模拟物理网元,从而可以减少单板的开发周期,降低开发成本。

## 附图说明

[0028] 图 1 是本发明在 PC 机上的仿真模拟系统模型图;

[0029] 图 2 是本发明模拟单板软件运行环境的方法一实施例流程示意图;

[0030] 图 3 是本发明仿真模拟环境中仿真模拟硬件的映射关系示意图;

[0031] 图 4 是本发明模拟单板软件运行环境的方法一实施例中修改单板软件代码中的地址映射方式的流程示意图;

[0032] 图 5 是本发明模拟单板软件运行环境的方法另一实施例流程示意图;

[0033] 图 6 是本发明模拟单板软件运行环境的装置一实施例结构示意图;

[0034] 图 7 是本发明模拟单板软件运行环境的装置另一实施例结构示意图。

[0035] 为了使本发明的技术方案更加清楚、明了,下面将结合附图作进一步详述。

## 具体实施方式

[0036] 本发明实施例解决方案主要是:通过在虚拟机上建立虚拟目标单板,并对虚拟机中不存在的硬件进行模拟,实现在 PC(Personal Computer, 个人计算机)机上模拟单板软件运行环境,为单板软件提供了一个没有硬件支撑的开发和调试环境,以解决仿真模拟环

境中硬件访问的问题,减少单板的开发周期,降低开发成本。

[0037] 如图 1 所示,图 1 是本发明在 PC 机上的仿真模拟系统模型图。搭建该仿真模拟环境首先需要在 PC 机上安装虚拟机,然后在虚拟机上运行虚拟单板的操作系统和应用层软件。在虚拟机上运行单板软件需要编译出适合在虚拟机上运行的映像文件。通常情况下,虚拟机只提供了对常用硬件的模拟和访问,如 CPU 的模拟,FLASH(闪存)、网卡、串口的模拟等。在虚拟单板运行过程中,可以通过模拟对硬件地址访问、FPGA 加载、EEPROM(Electrically Erasable Programmable Read-Only Memory,电可擦可编程只读存储器)中信息的读写、固化信息的保存等特殊硬件来实现单板软件的正常运行。

[0038] 如图 2 所示,本发明一实施例提出一种模拟单板软件运行环境的方法,包括:

[0039] 步骤 S101,配置虚拟硬盘,并修改单板软件代码中的地址映射方式;

[0040] 为了模拟单板软件的运行环境,需要在安装于 PC 机上的虚拟机中建立虚拟单板,在建立虚拟单板的过程中,首先需要配置虚拟硬盘,并修改单板软件代码中的地址映射方式。

[0041] 其中,在安装虚拟机时,可以根据系统所具有的资源选择相应的虚拟机安装在 PC 机上。

[0042] 虚拟硬盘在虚拟单板启动后挂载在虚拟单板上,以便对单板软件运行过程中的固化信息进行保存。

[0043] 修改单板软件代码中的地址映射方式是为了实现单板软件运行环境中对硬件地址访问的模拟。通常,一块单板包含业务芯片和业务 FPGA 等器件,这些器件寄存器通常都会映射到 CPU 的内存地址空间,CPU 在访问相应寄存器时,通过内存地址进行访问。

[0044] 单板在运行时会对芯片的某些地址进行直接访问,但是,在仿真模拟环境下,这些芯片并不存在,不能对这些地址进行直接访问。本实施例中,为了实现仿真模拟环境下对硬件地址的访问,在系统内核启动初期,向系统申请芯片地址空间大小的物理地址空间,然后再将芯片的虚拟地址空间映射到物理地址空间,从而完成芯片的虚拟地址的正常访问。本实施例仿真模拟环境中仿真模拟硬件的映射关系如图 3 所示。

[0045] 步骤 S102,编译出与安装在 PC 机上的虚拟机匹配的映像文件;

[0046] 本步骤中,在对单板硬件环境进行仿真模拟时,如何确保单板软件功能不被改动的情况下,编译出适合运行在仿真模拟环境的映像文件,以实现单板软件的正常运行,本实施例通过对仿真模拟时单板软件使用的 BSP(board support package,板级支持包)文件进行修改,编译出与虚拟机匹配的映像文件格式。此外,考虑到仿真模拟的内核与真实单板内核不同,可以通过宏变量和环境变量来控制编译过程,若宏变量和环境变量被定义,则按照仿真分支进行编译;若宏变量和环境变量没有被定义,则按照真实的分支进行编译。即在修改 BSP 文件的过程中可以通过宏来隔离修改,以实现将宏关闭后仍能够编译出适合在真实环境下运行的映像文件,即可以通过将宏关闭,实现对单板软件修改的恢复。

[0047] 步骤 S103,在虚拟机上运行映像文件,启动虚拟单板,并将虚拟硬盘挂载到虚拟单板上。

[0048] 本实施例中,通过虚拟机启动虚拟单板,实现单板在没有硬件支撑的情况下的运行。

[0049] 通过在虚拟机上运行映像文件,启动虚拟单板,在虚拟单板启动后,将虚拟硬盘挂

载在虚拟单板上,以便对单板软件运行过程中的固化信息进行保存。

[0050] 其中,固化信息的保存通过以下两种方式来实现在,一是将一些固化的配置文件写在虚拟单板启动时的根文件系统中。对于需要动态更新的文件则可以保存在虚拟硬盘中。由于虚拟单板运行时有的文件保存的目录是固定的,因此可以通过动态的挂载多块虚拟硬盘来满足文件保存路径多样化的需求。

[0051] 如图 4 所示,步骤 S101 中修改单板软件代码中的地址映射方式的步骤包括:

[0052] 步骤 S1011,申请预定芯片物理地址空间大小的物理地址空间;

[0053] 步骤 S1012,将需要访问的芯片虚拟地址空间映射到物理地址空间。

[0054] 将需要访问的芯片虚拟地址空间映射到物理地址空间,以实现虚拟单板对虚拟地址的访问。

[0055] 本实施例通过配置用于保存固化信息的虚拟硬盘、硬件地址访问的模拟以及编译适合于虚拟机即与虚拟机匹配的映像文件,实现了在 PC 机上模拟单板软件运行环境,使单板软件在没有硬件环境下能够正常的运行。有助于开发人员熟悉开发环境,在单板硬件没有开发完成时,帮助单板软件开发人员完成单板软件早期开发和调试,以及在组网测试网元不足时模拟物理网元,从而可以减少单板的开发周期,降低开发成本。

[0056] 如图 5 所示,本发明另一实施例提出一种模拟单板软件运行环境的方法,在上述实施例的基础上,在步骤 S103 之前还包括:

[0057] 步骤 S1031,将 FPGA 加载后的寄存器的值写入文本文件,并将文本文件保存在虚拟硬盘中;

[0058] 步骤 S1032 在虚拟单板启动时从文本文件中读取寄存器的值,对 FPGA 的寄存器进行初始化。

[0059] 本实施例与上述实施例的不同之处在于,本实施例还可实现 FPGA 的加载。

[0060] 在单板启动过程中,FPGA 的寄存器会记录单板启动时的一些硬件的启动信息。在仿真模拟环境下,由于只能对 FPGA 的寄存器进行访问,不能够通过 FPGA 的逻辑功能改变寄存器的值。因此,在模拟 FPGA 加载时,可以将 FPGA 加载后的寄存器的值写入文本文件中并保存于虚拟硬盘中,在虚拟单板启动时,通过读取虚拟硬盘中该文本文件内的寄存器值来初始化 FPGA 的寄存器,从而实现真实环境下情况下 FPGA 加载后的初始化效果。

[0061] 通过以上方法即可以实现单板软件在 PC 机上运行,实现对真实单板的模拟。

[0062] 本发明实施例通过配置用于保存固化信息的虚拟硬盘、硬件地址访问的模拟、编译与虚拟机匹配的映像文件及模拟 FPGA 加载,实现了在 PC 机上模拟单板软件运行环境。该模拟运行环境通过在虚拟机上建立虚拟单板,并对虚拟机中不存在的硬件进行模拟,以解决模拟环境中硬件访问的问题,从而使单板软件在没有硬件环境下能够正常的运行。从而有助于开发人员熟悉开发环境,在单板硬件没有开发完成时,帮助单板软件开发人员完成单板软件早期开发和调试,以及在组网测试网元不足时模拟物理网元,从而可以减少单板的开发周期,降低开发成本。

[0063] 如图 6 所示,本发明一实施例提出一种模拟单板软件运行环境的装置,包括:硬盘配置及地址映射模块 401、映像文件编译模块 402、虚拟单板启动模块 403,其中:

[0064] 硬盘配置及地址映射模块 401,用于配置虚拟硬盘,并修改单板软件代码中的地址映射方式;

- [0065] 映像文件编译模块 402,用于编译出与安装在 PC 机上的虚拟机匹配的映像文件;
- [0066] 虚拟单板启动模块 403,用于在虚拟机上运行映像文件,启动虚拟单板,并将虚拟硬盘挂载到虚拟单板上。
- [0067] 硬盘配置及地址映射模块 401 还用于申请预定芯片物理地址空间大小的物理地址空间;将需要访问的芯片虚拟地址空间映射到物理地址空间,以实现虚拟单板对虚拟地址的访问。
- [0068] 为了模拟单板软件的运行环境,需要在安装于 PC 机上的虚拟机中建立虚拟单板,在建立虚拟单板的过程中,首先通过硬盘配置及地址映射模块 401 配置虚拟硬盘,并修改单板软件代码中的地址映射方式。
- [0069] 其中,在安装虚拟机时,可以根据系统所具有的资源选择相应的虚拟机安装在 PC 机上。
- [0070] 虚拟硬盘在虚拟单板启动时挂载在虚拟单板上,以便对单板软件运行过程中的固化信息进行保存。
- [0071] 修改单板软件代码中的地址映射方式是为了实现单板软件运行环境中对硬件地址访问的模拟。通常,一块单板包含业务芯片和业务 FPGA 等器件,这些器件寄存器通常都会映射到 CPU 的内存地址空间,CPU 在访问相应寄存器时,通过内存地址进行访问。
- [0072] 单板在运行时会对芯片的某些地址进行直接访问,但是,在仿真模拟环境下,这些芯片并不存在,不能对这些地址进行直接访问。本实施例中,为了实现仿真模拟环境下对硬件地址的访问,在系统内核启动初期,向系统申请预定芯片物理地址空间大小的物理地址空间,然后再将芯片的虚拟地址空间映射到物理地址空间,从而完成芯片的虚拟地址的正常访问。本实施例仿真模拟环境中仿真模拟硬件的映射关系如图 3 所示。
- [0073] 在对单板硬件环境进行仿真模拟时,如何确保单板软件功能不被改动的情况下,编译出适合运行在仿真模拟环境的映像文件,以实现单板软件的正常运行,本实施例通过映像文件编译模块 402 对仿真模拟时单板软件使用的 BSP 文件进行修改,编译出与虚拟机匹配的映像文件格式。此外,考虑到仿真模拟的内核与真实单板内核不同,可以通过宏变量和环境变量来控制编译过程,若宏变量和环境变量被定义,则按照仿真分支进行编译;若宏变量和环境变量没有被定义,则按照真实的分支进行编译。即在修改 BSP 文件的过程中可以通过宏来隔离修改,以实现将宏关闭后仍能够编译出适合在真实环境下运行的映像文件,即可以通过将宏关闭,实现对单板软件修改的恢复。
- [0074] 在编译映像文件之后,通过虚拟单板启动模块 403 在虚拟机上运行映像文件,启动虚拟单板,并在虚拟单板启动后,将虚拟硬盘挂载在虚拟单板上,以便对单板软件运行过程中的固化信息进行保存。
- [0075] 其中,固化信息的保存通过以下两种方式来实现,一是将一些固化的配置文件写死在虚拟单板启动时的根文件系统中。对于需要动态更新的文件则可以保存在虚拟硬盘中。由于虚拟单板运行时有的文件保存的目录是固定的,因此可以通过动态的挂载多块虚拟硬盘来满足文件保存路径多样化的需求。
- [0076] 如图 7 所示,本发明另一实施例提出一种模拟单板软件运行环境的装置,在上述实施例的基础上,还包括:
- [0077] 加载文件写入模块 4031,与虚拟单板启动模块 403 连接,用于将 FPGA 加载后的寄

寄存器的值写入文本文件,并将文本文件保存在虚拟硬盘中;

[0078] 初始化模块 4032,用于在虚拟单板启动时,从文本文件中读取寄存器的值,对 FPGA 的寄存器进行初始化。

[0079] 本实施例与上述实施例的不同之处在于,本实施例还可实现 FPGA(Field-Programmable Gate Array,现场可编程门阵列)的加载。

[0080] 在单板启动过程中,FPGA 的寄存器会记录单板启动时的一些硬件的启动信息。在仿真模拟环境下,由于只能对 FPGA 的寄存器进行访问,不能够通过 FPGA 的逻辑功能改变寄存器的值。因此,在模拟 FPGA 加载时,可以通过加载文件写入模块 4031 将 FPGA 加载后的寄存器的值写入文本文件中并保存于虚拟硬盘中,初始化模块 4032 在虚拟单板启动时,通过读取虚拟硬盘中该文本文件内的寄存器值来初始化 FPGA 的寄存器,从而实现真实环境下情况下 FPGA 加载后的初始化效果。

[0081] 本发明实施例模拟单板软件运行环境的方法及装置,通过配置用于保存固化信息的虚拟硬盘、硬件地址访问的模拟以及编译出与虚拟机匹配的映像文件,实现了在 PC 机上模拟单板软件运行环境,为单板软件提供了一个没有硬件支撑的开发和调试环境。该模拟运行环境通过在虚拟机上建立虚拟目标单板,并对虚拟机中不存在的硬件进行模拟,以解决模拟环境中硬件访问的问题,从而使单板软件在没有硬件环境下能够正常的运行。本发明可以帮助开发人员熟悉开发环境,在单板硬件没有开发完成时,帮助单板软件开发人员完成单板软件早期开发和调试,以及在组网测试网元不足时模拟物理网元,从而可以减少单板的开发周期,降低开发成本。

[0082] 以上所述仅为本发明的优选实施例,并非因此限制本发明的专利范围,凡是利用本发明说明书及附图内容所作的等效结构或流程变换,或直接或间接运用在其它相关的技术领域,均同理包括在本发明的专利保护范围内。



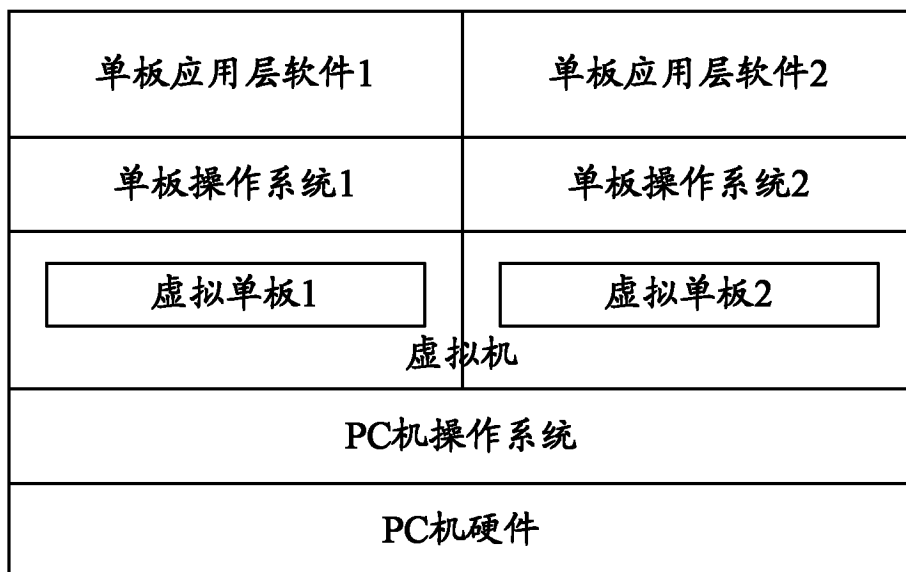


图 1

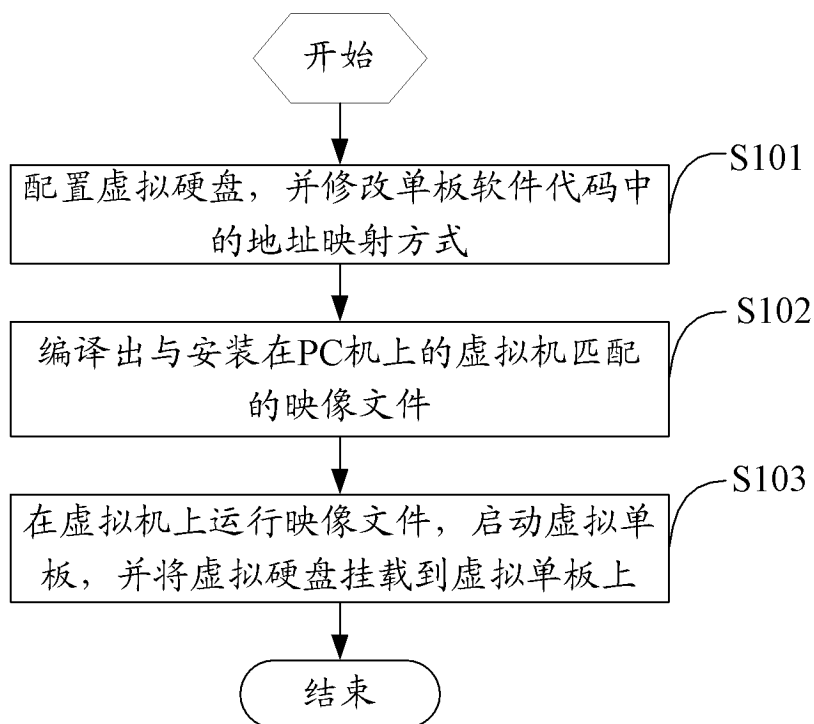


图 2

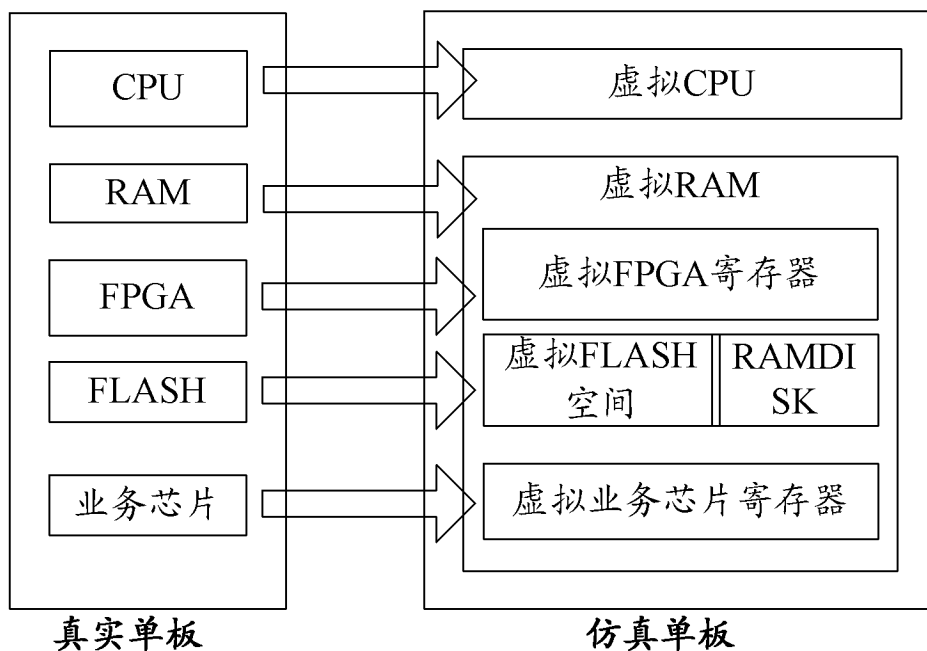


图 3

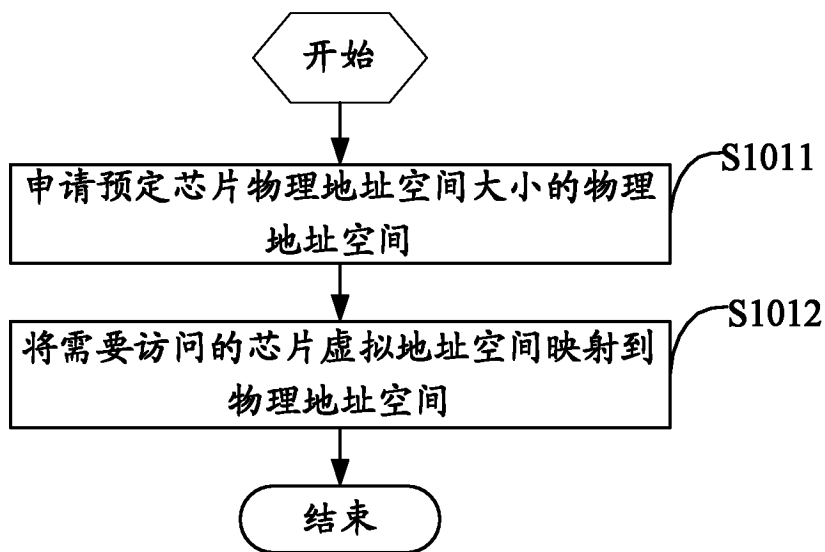


图 4

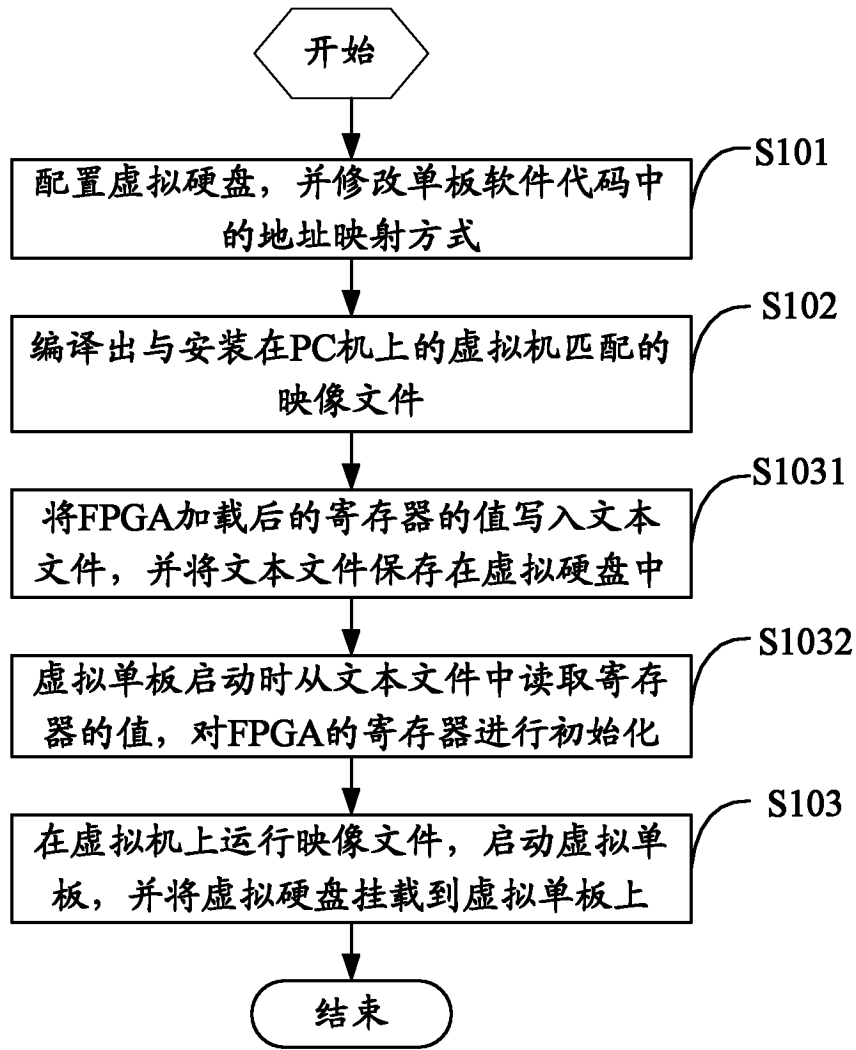


图 5

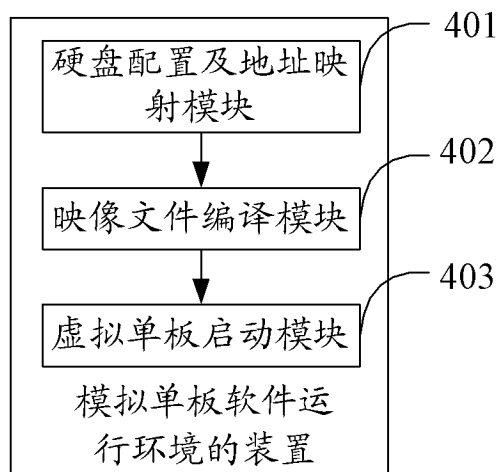


图 6

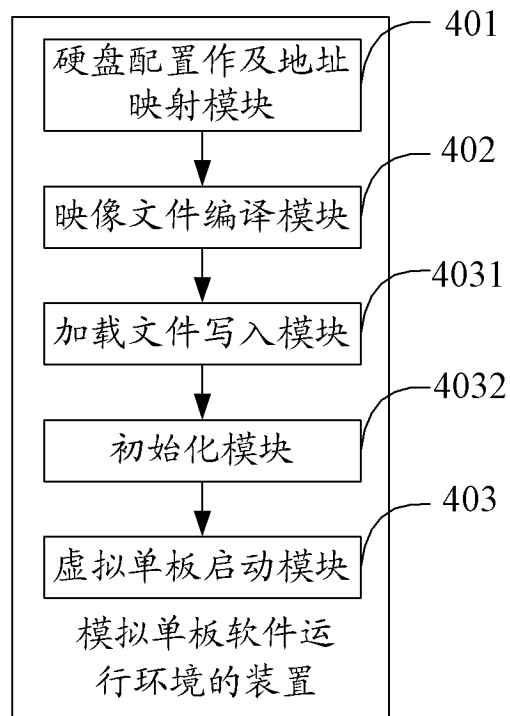


图 7