



(12)发明专利申请

(10)申请公布号 CN 108600331 A

(43)申请公布日 2018.09.28

(21)申请号 201810282828.9

(22)申请日 2018.04.02

(71)申请人 北京大米科技有限公司

地址 100142 北京市海淀区西四环北路158
号1幢9层9071号

(72)发明人 孙东阳 刘军立

(74)专利代理机构 北京正理专利代理有限公司

11257

代理人 付生辉

(51)Int.Cl.

H04L 29/08(2006.01)

权利要求书2页 说明书6页 附图2页

(54)发明名称

交互界面呈现方法、客户端、终端及系统

(57)摘要

本发明公开一种交互界面呈现方法、客户端、终端及系统。该方法的具体实施方式包括：客户端内嵌的HTML5页面为由待下载数据包拆分得到的每个子数据包分别分配一个下载线程，其中每个子数据包封装一个交互界面的数据；HTML5页面依据呈现时序下载子数据包；HTML5页面依次加载子数据包并呈现交互界面。该实现方式可加快首个交互界面的呈现速度，后续子数据包的下载过程不影响当前交互界面的呈现，交互界面呈现的复杂度低、稳定性高。

客户端内嵌的HTML5页面为由待下载数据包拆分得到的每个子数据包分别分配一个下载线程，其中每个子数据包封装一个交互界面的数据

HTML5页面依据呈现时序下载子数据包

HTML5页面依次加载子数据包并呈现交互界面

1. 一种交互界面呈现方法,其特征在于,包括:

客户端内嵌的HTML5页面为由待下载数据包拆分得到的每个子数据包分别分配一个下载线程,其中每个子数据包封装一个交互界面的数据;

所述HTML5页面依据呈现时序下载子数据包;

所述HTML5页面依次加载子数据包并呈现交互界面。

2. 根据权利要求1所述的交互界面呈现方法,其特征在于,所述HTML5页面依次加载子数据包并呈现交互界面进一步包括:所述HTML5页面分配加载线程和界面线程,所述加载线程解析子数据包中的数据;所述界面线程对子数据包中的数据进行渲染,从而呈现交互界面。

3. 根据权利要求1所述的交互界面呈现方法,其特征在于,所述依据呈现时序下载所述子数据包进一步包括:

依据呈现时序依次下载子数据包;或,

依据呈现时序,先下载第一个呈现的交互界面对应的子数据包,再同时下载其他的子数据包。

4. 根据权利要求3所述的交互界面呈现方法,其特征在于,所述依据呈现时序下载所述子数据包进一步包括:

判断下载速度是否小于阈值:

若是,则依据呈现时序依次下载子数据包;

若否,则依据呈现时序,先下载第一个呈现的交互界面对应的子数据包,再同时下载其他的子数据包;

其中,所述阈值依据当前网络带宽和子数据包字节长度最大值设定。

5. 根据权利要求1所述的交互界面呈现方法,其特征在于,所述子数据包为将转换为base64字符串的待下载数据包压缩得到的ZIP压缩包。

6. 一种客户端,该客户端内嵌有HTML5页面,其特征在于,所述HTML5页面包括:

下载模块,基于下载线程,依据呈现时序下载由待下载数据包拆分得到的子数据包,所述下载线程与子数据包一一对应;

加载模块,基于加载线程解析子数据包中的数据;

界面模块,基于界面线程对子数据包中的数据进行渲染,从而呈现交互界面。

7. 根据权利要求6所述的客户端,其特征在于,

所述下载模块,依据呈现时序依次下载子数据包;或,

所述下载模块,依据呈现时序,先下载第一个呈现的交互界面对应的子数据包,再同时下载其他的子数据包。

8. 根据权利要求7所述的客户端,其特征在于,所述HTML5页面还包括:下载速度判断模块,判断下载速度是否小于阈值,若是则向下载模块发送第一信号,若否则向下载模块发送第二信号,其中,所述阈值依据子数据包字节长度的最大值设定;

所述下载模块,收到第一信号后,依据呈现时序依次下载子数据包;收到第二信号后,依据呈现时序,先下载第一个呈现的交互界面对应的子数据包,再同时下载其他的子数据包。

9. 一种交互界面呈现系统,其特征在于,包括:服务器和至少一个客户端,所述客户端

内嵌有HTML5页面,所述HTML5页面包括下载模块、加载模块和界面模块;

所述服务器,将待下载数据包包拆分为多个子数据包,每个子数据包封装一个交互界面的数据;

所述下载模块,基于下载线程,依据呈现时序下载子数据包,所述下载线程与子数据包一一对应;

所述加载模块,基于加载线程解析子数据包中的数据;

所述界面模块,基于界面线程对子数据包中的数据进行渲染,从而呈现交互界面。

10.根据权利要求9所述的交互界面呈现系统,其特征在于,

所述下载模块,依据呈现时序依次下载子数据包;或,

所述下载模块,依据呈现时序,先下载第一个呈现的交互界面对应的子数据包,再同时下载其他的子数据包。

11.根据权利要求10所述的交互界面呈现系统,其特征在于,所述HTML5页面还包括:下载速度判断模块,判断下载速度是否小于阈值,若是则向下载模块发送第一信号,若否则向下载模块发送第二信号,其中,所述阈值依据子数据包字节长度的最大值设定;

所述下载模块,收到第一信号后,依据呈现时序依次下载子数据包;收到第二信号后,依据呈现时序,先下载第一个呈现的交互界面对应的子数据包,再同时下载其他的子数据包。

12.根据权利要求9所述的交互界面呈现系统,其特征在于,所述服务器将转换为base64字符串的待下载数据包压缩得到多个ZIP压缩包形式的子数据包。

13.一种终端,包括存储器、处理器及存储在存储器上并可在处理器上运行的计算机程序,其特征在于,所述处理器执行所述程序时实现如权利要求1-5中任一项所述的方法。

交互界面呈现方法、客户端、终端及系统

技术领域

[0001] 本发明涉及网络技术领域。更具体地，涉及一种交互界面呈现方法、客户端、终端及系统。

背景技术

[0002] 目前，客户端分为基于本地系统运行的原生客户端和基于浏览器运行的网页客户端，多数的客户端开发都是使用的原生程序编写的应用程序，也就是说大部分的客户端属于原生客户端。在原生客户端中，例如微信、QQ等大量客户端都内嵌了HTML5，本地系统与H5互调可以实现混合开发，将一部分原生的功能改为HTML 5来开发。

[0003] 在交互界面下载和呈现时，以在线课堂的作业交互界面为例，为保证用户在做作业时的稳定性和连贯性，平板电脑（例如IPAD）和计算机的客户端均采用作业数据包下载的方式，即客户端预先下载作业数据包，然后将作业交互界面逐一呈现，以使得用户可逐题完成作业。但是随着作业内容越来越丰富，作业数据包也越来越大，有的已经达到几十MB甚至上百MB。这就使得用户在做作业前要等待较长的下载时间，用户体验非常不佳。而且现有客户端采用的基于HTML5结合原生客户端的下载方案需要客户端进程和HTML5进程之间频繁通信，而且由于是客户端下载，数据交互都要在客户端启动作业后传入传出，任何一个环节异常都会让作业的呈现出现卡顿，无法继续，数据交互异常复杂，稳定性较差。

[0004] 因此，需要提供一种基于HTML5的方式及分包下载的交互界面呈现方法、客户端、终端及系统。

发明内容

[0005] 为解决交互界面下载和呈现时客户端加载等待时间过长，复杂度增加，稳定性较差的问题。本发明提供一种交互界面呈现方法、客户端、终端及系统。

[0006] 为达到上述目的，本发明采用下述技术方案：

[0007] 本发明第一方面提供了一种交互界面呈现方法，包括：

[0008] 客户端内嵌的HTML5页面为由待下载数据包拆分得到的每个子数据包分别分配一个下载线程，其中每个子数据包封装一个交互界面的数据；

[0009] 所述HTML5页面依据呈现时序下载子数据包；

[0010] 所述HTML5页面依次加载子数据包并呈现交互界面。

[0011] 优选地，所述HTML5页面依次加载子数据包并呈现交互界面进一步包括：所述HTML5页面分配加载线程和界面线程，所述加载线程解析子数据包中的数据；所述界面线程对子数据包中的数据进行渲染，从而呈现交互界面。

[0012] 优选地，所述依据呈现时序下载所述子数据包进一步包括：

[0013] 依据呈现时序依次下载子数据包；或，

[0014] 依据呈现时序，先下载第一个呈现的交互界面对应的子数据包，再同时下载其他的子数据包。

- [0015] 优选地，所述依据呈现时序下载所述子数据包进一步包括：
- [0016] 判断下载速度是否小于阈值；
- [0017] 若是，则依据呈现时序依次下载子数据包；
- [0018] 若否，则依据呈现时序，先下载第一个呈现的交互界面对应的子数据包，再同时下载其他的子数据包；
- [0019] 其中，所述阈值依据当前网络带宽和子数据包字节长度最大值设定。
- [0020] 优选地，所述子数据包为将转换为base64字符串的待下载数据包压缩得到的ZIP压缩包。
- [0021] 本发明第二方面提供了一种客户端，该客户端内嵌有HTML5页面，所述HTML5页面包括：
- [0022] 下载模块，基于下载线程，依据呈现时序下载由待下载数据包拆分得到的子数据包，所述下载线程与子数据包一一对应；
- [0023] 加载模块，基于加载线程解析子数据包中的数据；
- [0024] 界面模块，基于界面线程对子数据包中的数据进行渲染，从而呈现交互界面。
- [0025] 优选地，所述下载模块，依据呈现时序依次下载子数据包；或，所述下载模块，依据呈现时序，先下载第一个呈现的交互界面对应的子数据包，再同时下载其他的子数据包。
- [0026] 优选地，所述HTML5页面还包括：下载速度判断模块，判断下载速度是否小于阈值，若是则向下载模块发送第一信号，若否则向下载模块发送第二信号，其中，所述阈值依据子数据包字节长度的最大值设定；所述下载模块，收到第一信号后，依据呈现时序依次下载子数据包；收到第二信号后，依据呈现时序，先下载第一个呈现的交互界面对应的子数据包，再同时下载其他的子数据包。
- [0027] 本发明第三方面提供了一种交互界面呈现系统，包括：服务器和至少一个客户端，所述客户端内嵌有HTML5页面，所述HTML5页面包括下载模块、加载模块和界面模块；
- [0028] 所述服务器，将待下载数据包包拆分为多个子数据包，每个子数据包封装一个交互界面的数据；
- [0029] 所述下载模块，基于下载线程，依据呈现时序下载子数据包，所述下载线程与子数据包一一对应；
- [0030] 所述加载模块，基于加载线程解析子数据包中的数据；
- [0031] 所述界面模块，基于界面线程对子数据包中的数据进行渲染，从而呈现交互界面。
- [0032] 优选地，所述下载模块，依据呈现时序依次下载子数据包；或，所述下载模块，依据呈现时序，先下载第一个呈现的交互界面对应的子数据包，再同时下载其他的子数据包。
- [0033] 优选地，所述HTML5页面还包括：下载速度判断模块，判断下载速度是否小于阈值，若是则向下载模块发送第一信号，若否则向下载模块发送第二信号，其中，所述阈值依据子数据包字节长度的最大值设定；所述下载模块，收到第一信号后，依据呈现时序依次下载子数据包；收到第二信号后，依据呈现时序，先下载第一个呈现的交互界面对应的子数据包，再同时下载其他的子数据包。
- [0034] 优选地，所述服务器将转换为base64字符串的待下载数据包压缩得到多个ZIP压缩包形式的子数据包。
- [0035] 本发明第四方面提供了一种终端，包括存储器、处理器及存储在存储器上并可在

处理器上运行的计算机程序,所述处理器执行所述程序时实现本发明第一方面提供的交互界面呈现方法。

[0036] 本发明的有益效果如下:

[0037] 本发明所述技术方案在交互界面下载和呈现时不依赖于原生客户端,使交互界面的呈现和后续升级更加灵活。且本发明所述技术方案能够快速启动交互界面的呈现,不需要等待过长的下载时间,还可按需加载后续的交互界面。

附图说明

[0038] 下面结合附图对本发明的具体实施方式作进一步详细的说明;

[0039] 图1示出本发明实施例提供的交互界面呈现方法的流程图。

[0040] 图2示出本发明实施例提供的交互界面呈现系统的示意图。

[0041] 图3示出本发明实施例提供的交互界面呈现系统的工作流程图。

[0042] 图4示出本发明实施例提供的终端的结构示意图。

具体实施方式

[0043] 为了更清楚地说明本发明,下面结合优选实施例和附图对本发明做进一步的说明。附图中相似的部件以相同的附图标记进行表示。本领域技术人员应当理解,下面所具体描述的内容是说明性的而非限制性的,不应以此限制本发明的保护范围。

[0044] 如图1所示,本发明的一个实施例提供了一种交互界面呈现方法,包括:

[0045] 客户端内嵌的HTML5页面为由待下载数据包拆分得到的每个子数据包分别分配一个下载线程,其中每个子数据包封装一个交互界面的数据;

[0046] HTML5页面依据呈现时序下载子数据包;

[0047] HTML5页面依次加载子数据包并呈现交互界面。

[0048] 本实施例提供的交互界面呈现方法,将待下载的交互界面数据包拆分为多个封装一个交互界面的数据的子数据包,在客户端下载子数据包时调用其内嵌的HTML5页面依据呈现时序下载,且下载子数据包时HTML5页面为每一个子数据分配一个下载线程。本实施例提供的交互界面呈现方法将交互界面下载和呈现与原生客户端解耦(把交互界面封装成一个配置消息,原生客户端中各个功能模块通过信息交互模块获取交互界面)。由此,本实施例提供的交互界面呈现方法在交互界面下载和呈现时不依赖于原生客户端,使交互界面的呈现和后续升级更加灵活。且本实施例提供的交互界面呈现方法能够快速启动交互界面的呈现,不需要等待过长的下载时间,还可按需加载后续的交互界面。

[0049] 本实施例中,客户端为平板电脑(例如IPAD)、计算机或智能手机中的客户端,客户端可通过控件WebView调用其内嵌的HTML5页面,由HTML5页面实现下载、加载及呈现。

[0050] 在本实施例的一些可选的实现方式中,HTML5页面依次加载子数据包并呈现交互界面的方式为:HTML5页面分配加载线程和界面线程,加载线程解析子数据包中的数据;界面线程对子数据包中的数据进行渲染,从而呈现交互界面。这样实现了后续的下载线程下载和加载线程解析的过程不影响当前界面线程进行渲染。本实施例中,HTML5页面采用多线程技术WebWork,各线程内采用Ajax技术实现。

[0051] 在本实施例的一些可选的实现方式中,依据呈现时序下载子数据包进一步包括:

依据呈现时序依次下载子数据包;或,依据呈现时序,先下载第一个呈现的交互界面对应的子数据包,再同时下载其他的子数据包。这两种方式均可实现快速启动交互界面的呈现。

[0052] 在本实施例的一些可选的实现方式中,依据呈现时序下载子数据包进一步包括:

[0053] 判断下载速度是否小于阈值:

[0054] 若是,则依据呈现时序依次下载子数据包;

[0055] 若否,则依据呈现时序,先下载第一个呈现的交互界面对应的子数据包,再同时下载其他的子数据包;

[0056] 其中,阈值依据当前网络带宽和子数据包字节长度最大值设定。

[0057] 这样,通过对当前网络带宽和子数据包大小的预估下载速度,或者说预估单个子数据包的下载时间,若下载速度较快,则可采用先下载第一个呈现的交互界面对应的子数据包,之后再同时下载后续的子数据包的策略;若下载速度较慢,则可采用依次下载子数据包的策略。

[0058] 在本实施例的一些可选的实现方式中,子数据包为将转换为base64字符串的交互界面的数据压缩得到的ZIP压缩包。以交互界面为在线课堂的作业交互界面为例:对于答题形式的作业交互界面,可进一步将答题形式的作业交互界面中的题型逻辑和样式主题再次分包下载,这样一方面可以控制按需下载的最少字节,二来可以控制主题样式切换。而对于和游戏形式的作业交互界面,在拆分子数据包时可将图片、音频资源等数据全部转成base64字符串再压缩成多个ZIP压缩包,并生成多个ZIP压缩包地址和管理关系的配置文件。

[0059] 本发明的另一个实施例提供了一种客户端,该客户端内嵌有HTML5页面,HTML5页面包括:

[0060] 下载模块,基于下载线程,依据呈现时序下载由待下载数据包拆分得到的子数据包,下载线程与子数据包一一对应;

[0061] 加载模块,基于加载线程解析子数据包中的数据;

[0062] 界面模块,基于界面线程对子数据包中的数据进行渲染,从而呈现交互界面。

[0063] 需要说明的是,本实施例提供的客户端的原理及工作流程与上述交互界面呈现方法相似,相关之处可以参照上述说明,在此不再赘述。

[0064] 如图2所示,本发明的另一个实施例提供了一种交互界面呈现系统,包括:服务器和至少一个客户端,客户端内嵌有HTML5页面,HTML5页面包括下载模块、加载模块和界面模块;

[0065] 服务器,将待下载数据包包拆分为多个子数据包,每个子数据包封装一个交互界面的数据;

[0066] 下载模块,基于下载线程,依据呈现时序下载子数据包,其中,下载线程与子数据包一一对应;

[0067] 加载模块,基于加载线程解析子数据包中的数据;

[0068] 界面模块,基于界面线程对子数据包中的数据进行渲染,从而呈现交互界面。

[0069] 如图3所示,本实施例提供的交互界面呈现系统在工作过程中,服务器在拆分子数据包时可将图片、音频资源等数据全部转成base64字符串再压缩成多个ZIP压缩包,客户端内嵌的HTML5页面采用多线程技术WebWork为每一个ZIP压缩包分配一个下载线程进行下载。

[0070] 需要说明的是，本实施例提供的交互界面呈现系统的原理及工作流程与上述交互界面呈现方法相似，相关之处可以参照上述说明，在此不再赘述。

[0071] 本发明的另一个实施例提供了一种终端。如图4所示，适于用来实现本实施例提供的终端的计算机系统，包括中央处理单元(CPU)，其可以根据存储在只读存储器(ROM)中的程序或者从存储部分加载到随机访问存储器(RAM)中的程序而执行各种适当的动作和处理。在RAM中，还存储有计算机系统操作所需的各種程序和数据。CPU、ROM以及RAM通过总线彼此相连。输入/输出(I/O)接口也连接至总线。

[0072] 以下部件连接至I/O接口：包括键盘、鼠标等的输入部分；包括诸如液晶显示器(LCD)等以及扬声器等的输出部分；包括硬盘等的存储部分；以及包括诸如LAN卡、调制解调器等的网络接口卡的通信部分。通信部分经由诸如因特网的网络执行通信处理。驱动器也需要连接至I/O接口。可拆卸介质，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。

[0073] 特别地，根据本实施例，上文流程图描述的过程可以被实现为计算机软件程序。例如，本实施例包括一种计算机程序产品，其包括有形地包含在计算机可读介质上的计算机程序，上述计算机程序包含用于执行流程图所示的方法的程序代码。在这样的实施例中，该计算机程序可以通过通信部分从网络上被下载和安装，和/或从可拆卸介质被安装。

[0074] 附图中的流程图和示意图，图示了本实施例的系统、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，流程图或示意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，上述模块、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，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。也要注意的是，示意图和/或流程图中的每个方框、以及示意和/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。

[0075] 作为另一方面，本实施例还提供了一种计算机可读存储介质(本实施例中为非易失性计算机存储介质)，该非易失性计算机存储介质可以是上述实施例中上述终端中所包含的非易失性计算机存储介质，也可以是单独存在，未装配入终端中的非易失性计算机存储介质。上述非易失性计算机存储介质存储有一个或者多个程序，当上述一个或者多个程序被一个终端执行时，使得上述终端中客户端内嵌的HTML5页面为由待下载数据包拆分得到的每个子数据包分别分配一个下载线程，其中每个子数据包封装一个交互界面的数据；HTML5页面依据呈现时序下载子数据包；HTML5页面依次加载子数据包并呈现交互界面。

[0076] 在本发明的描述中，需要说明的是，术语“上”、“下”等指示的方位或位置关系为基于附图所示的方位或位置关系，仅是为了便于描述本发明和简化描述，而不是指示或暗示所指的装置或元件必须具有特定的方位、以特定的方位构造和操作，因此不能理解为对本发明的限制。除非另有明确的规定和限定，术语“安装”、“相连”、“连接”应做广义理解，例如，可以是固定连接，也可以是可拆卸连接，或一体地连接；可以是机械连接，也可以是电连接；可以是直接相连，也可以通过中间媒介间接相连，可以是两个元件内部的连通。对于本领域的普通技术人员而言，可以根据具体情况理解上述术语在本发明中的具体含义。

[0077] 还需要说明的是,在本发明的描述中,诸如第一和第二等之类的关系术语仅仅用来将一个实体或者操作与另一个实体或操作区分开来,而不一定要求或者暗示这些实体或操作之间存在任何这种实际的关系或者顺序。而且,术语“包括”、“包含”或者其任何其他变体意在涵盖非排他性的包含,从而使得包括一系列要素的过程、方法、物品或者设备不仅包括那些要素,而且还包括没有明确列出的其他要素,或者是还包括为这种过程、方法、物品或者设备所固有的要素。在没有更多限制的情况下,由语句“包括一个……”限定的要素,并不排除在包括所述要素的过程、方法、物品或者设备中还存在另外的相同要素。

[0078] 显然,本发明的上述实施例仅仅是为清楚地说明本发明所作的举例,而并非是对本发明的实施方式的限定,对于本领域的普通技术人员来说,在上述说明的基础上还可以做出其它不同形式的变化或变动,这里无法对所有的实施方式予以穷举,凡是属于本发明的技术方案所引伸出的显而易见的变化或变动仍处于本发明的保护范围之列。

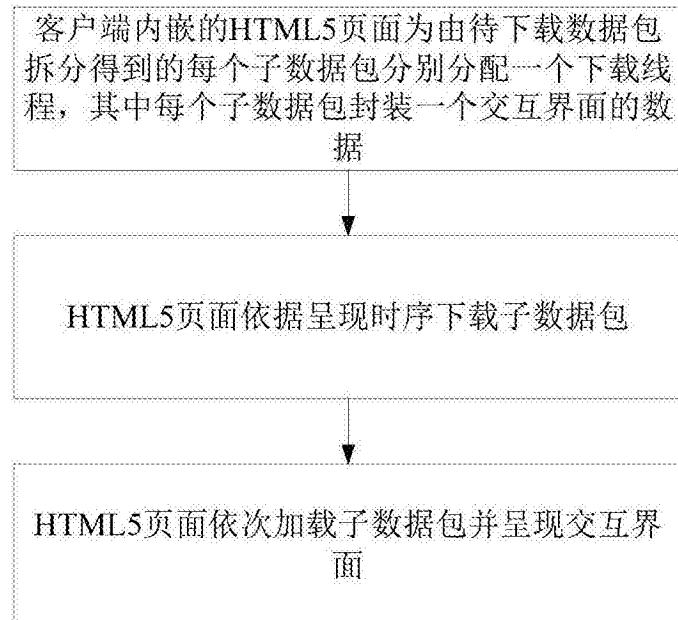


图1

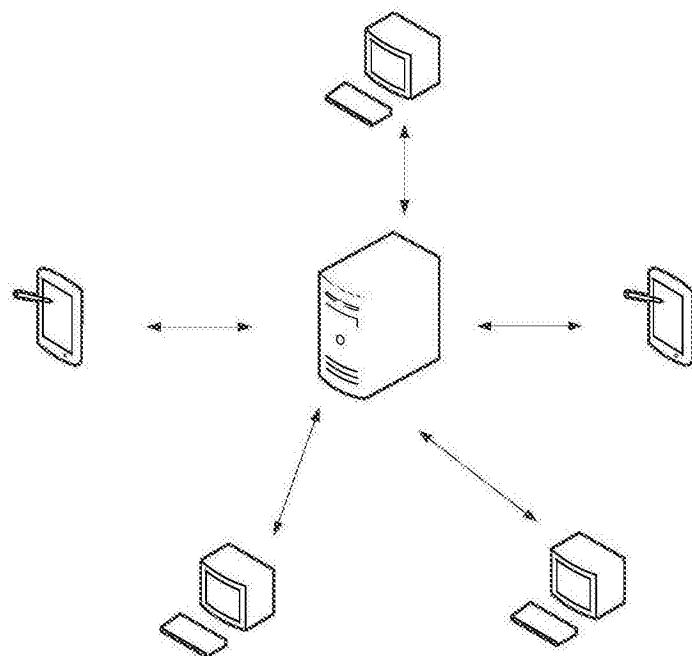


图2

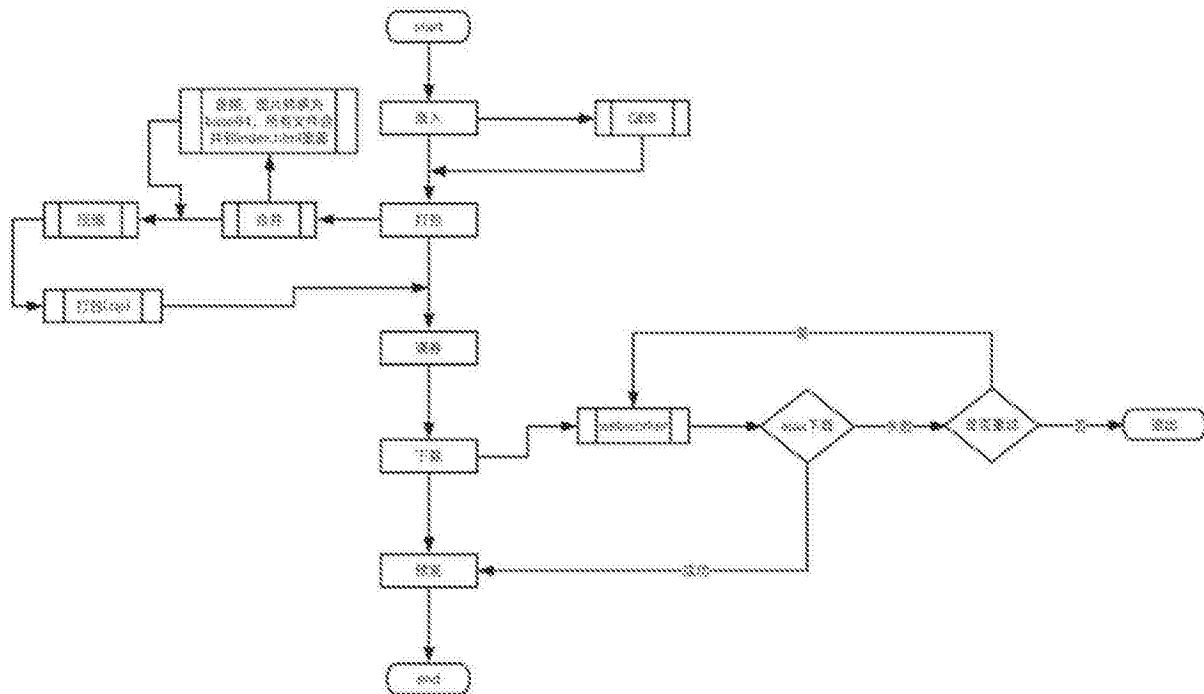


图3

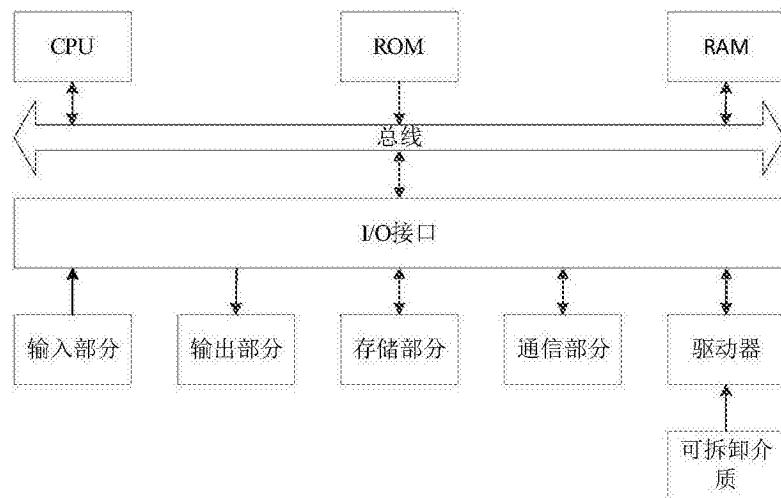


图4