



(12) 发明专利申请

(10) 申请公布号 CN 102308284 A

(43) 申请公布日 2012. 01. 04

(21) 申请号 201180001255. 4

(22) 申请日 2011. 07. 01

(85) PCT申请进入国家阶段日
2011. 09. 13

(86) PCT申请的申请数据
PCT/CN2011/076774 2011. 07. 01

(71) 申请人 华为技术有限公司
地址 518129 中国广东省深圳市龙岗区坂田
华为总部办公楼

(72) 发明人 胡海峰

(74) 专利代理机构 北京永新同创知识产权代理
有限公司 11376
代理人 钟胜光

(51) Int. Cl.
G06F 11/00 (2006. 01)

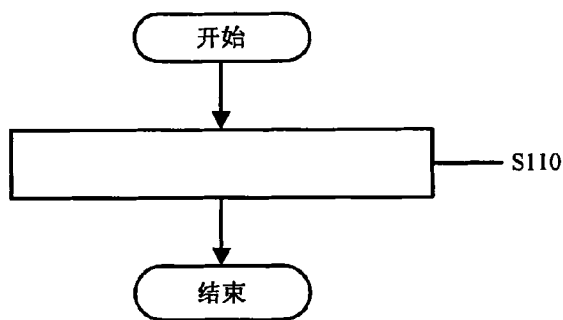
权利要求书 2 页 说明书 5 页 附图 2 页

(54) 发明名称

用于快速复位设备的方法和装置

(57) 摘要

本发明涉及一种用于快速复位设备的方法和装置, 其中, 该方法包括步骤: 当设备的多个处理器的其中一个处理器出现异常时, 仅对所述其中一个处理器进行复位。利用该方法和装置, 能够有效地减少了设备复位所需的时间, 加快了设备复位的过程。



1. 一种用于快速复位设备的方法,包括步骤:

当所述设备的多个处理器的其中一个处理器出现异常时,仅对所述出现异常的处理器进行复位。

2. 如权利要求 1 所述的方法,其中,所述复位步骤进一步包括:

在对所述出现异常的处理器进行复位时,促使所述出现异常的处理器直接从操作系统映像的代码段的第一个指令开始执行。

3. 如权利要求 2 所述的方法,其中,所述复位步骤还包括:

在所述操作系统映像的代码段中用于对所述操作系统映像的 BSS 段进行清零的指令被所述出现异常的处理器执行后,促使所述出现异常的处理器使用以前备份的所述操作系统映像的数据段来覆盖所述设备的随机存取存储器所存储的所述操作系统映像的数据段,其中,所述以前备份的所述操作系统映像的数据段是在所述设备的上一次常规启动或复位时对从所述设备的只读存储器搬移到所述随机存取存储器的所述操作系统映像的数据段进行备份而得到的。

4. 如权利要求 2-3 的任何一个权利要求所述的方法,其中,所述复位步骤还包括:

在促使所述出现异常的处理器直接从所述操作系统映像的代码段的第一个指令开始执行之前,促使所述出现异常的处理器备份所述随机存取存储器所存储的所述出现异常的处理器与所述多个处理器的其它处理器交互的数据;

在所述操作系统映像的代码段中用于进行系统初始化的指令被所述出现异常的处理器执行之后,促使所述出现异常的处理器在所述随机存取存储器中恢复所备份的所述出现异常的处理器与所述其它处理器交互的数据。

5. 如权利要求 2-4 的任意一个权利要求所述的方法,其中,所述复位步骤还包括:

在所述操作系统映像的代码段中用于进行系统初始化的指令被所述出现异常的处理器执行时,促使所述出现异常的处理器跳过用于对所述设备的外围芯片进行初始化的指令。

6. 如权利要求 1-5 的任意一个权利要求所述的方法,其中,

所述设备是网络设备,

所述多个处理器包括用于控制面处理的处理器和用于数据面处理的处理器。

7. 一种用于快速复位设备的装置,包括:

复位模块,用于当所述设备的多个处理器的其中一个处理器出现异常时,仅对所述出现异常的处理器进行复位。

8. 如权利要求 7 所述的装置,其中,所述复位模块进一步包括:

执行模块,用于在对所述出现异常的处理器进行复位时,促使所述出现异常的处理器直接从操作系统映像的代码段的第一个指令开始执行。

9. 如权利要求 8 所述的装置,其中,所述复位模块还包括:

覆盖模块,用于在所述操作系统映像的代码段中用于对所述操作系统映像的 BSS 段进行清零的指令被所述出现异常的处理器执行后,促使所述出现异常的处理器使用以前备份的所述操作系统映像的数据段来覆盖所述设备的随机存取存储器所存储的所述操作系统映像的数据段,其中,所述以前备份的所述操作系统映像的数据段是在所述设备的上一次常规启动或复位时对从所述设备的只读存储器搬移到所述随机存取存储器的所述操作系

统映像的数据段进行备份而得到的。

10. 如权利要求 8-9 的任意一个权利要求所述的装置,其中,所述复位模块还包括:

备份模块,用于在促使所述出现异常的处理器直接从所述操作系统映像的代码段的第一个指令开始执行之前,促使所述出现异常的处理器备份所述随机存取存储器所存储的所述出现异常的处理器与所述多个处理器的其它处理器交互的数据;

恢复模块,用于在所述操作系统映像的代码段中用于进行系统初始化的指令被所述出现异常的处理器执行之后,促使所述出现异常的处理器在所述随机存取存储器中恢复所备份的所述出现异常的处理器与所述其它处理器交互的数据。

11. 如权利要求 8-10 的任意一个权利要求所述的装置,其中,所述复位模块还包括:

跳过模块,用于在所述操作系统映像的代码段中用于进行系统初始化的指令被所述出现异常的处理器执行时,促使所述出现异常的处理器跳过用于对所述设备的外围芯片进行初始化的指令。

12. 如权利要求 8-11 的任意一个权利要求所述的装置,其中,

所述设备是网络设备,

所述多个处理器包括用于控制面处理的处理器和用于数据面处理的处理器。

13. 一种机器可读存储介质,其存储机器可执行指令,当所述机器可执行指令被执行时使得机器执行权利要求 1-6 的任何一个权利要求中的步骤。

用于快速复位设备的方法和装置

技术领域

[0001] 本发明涉及用于快速复位设备的方法和装置。

背景技术

[0002] 现在的大部分网络设备通常采用数据面和控制面分离的方式来进行报文处理。通常,为了实现数据面和控制面的分离,这些网络设备使用两个处理器来分别处理数据面的报文和控制面的报文。

[0003] 在工作过程中,网络设备的各个处理器难免会出现异常。当网络设备的任何一个处理器出现异常时,网络设备都需要中断业务并进行复位以消除所出现的异常。当网络设备复位时,网络设备对各个处理器逐一依次进行复位。按照这种复位方式,与仅具有单个处理器的网络设备相比,具有多个处理器的网络设备的复位需要花费更多的时间。

[0004] 为了给用户提供更好的通信体验,通信运营商对网络设备的设备故障率具有越来越严格的要求。通常,通信运营商对大部分网络设备的设备故障率要求小于 0.99999,有的网络设备的故障率甚至要求小于 0.9999999。按照故障率小于 0.99999 的要求,一个网络设备在每月的业务中断时间应小于 22 秒。

[0005] 然而,具有多个处理器的网络设备从开始复位到复位完成一般都需要几分钟的时间,这远远不能满足通信运营商对网络设备的设备故障率的要求。

发明内容

[0006] 考虑到现有技术的上述问题,本发明的实施例提出一种用于快速复位设备的方法和装置,其可以对设备进行快速复位,从而有效地减少设备复位所需的时间。

[0007] 按照本发明实施例的一种用于快速复位设备的方法,包括步骤:当所述设备的多个处理器的其中一个处理器出现异常时,仅对所述出现异常的处理器进行复位。

[0008] 按照本发明实施例的一种用于快速复位设备的装置,包括:复位模块,用于当所述设备的多个处理器的其中一个处理器出现异常时,仅对所述出现异常的处理器进行复位。

[0009] 由于仅对设备中出现异常的处理器进行复位,而不是对设备中的所有处理器都进行复位,所以与现有技术相比,本发明有效地减少了设备复位所需的时间,加快了设备复位的过程。

附图说明

[0010] 本发明的其它特点、优点和益处通过以下结合附图的详细描述将变得更加显而易见。其中:

[0011] 图 1 示出了按照本发明实施例一的对网络设备进行快速复位的方法流程图;

[0012] 图 2 示出了按照本发明实施例二的对网络设备进行快速复位的方法流程图;以及

[0013] 图 3 示出了按照本发明一个实施例的用于快速复位设备的装置。

具体实施方式

[0014] 本发明的实施例提供一种用于快速复位设备的方案,其可以对设备进行快速复位。按照该方案,当设备的多个处理器的其中一个处理器出现异常(例如,软失效、Dcache、Icache等)时,仅对该出现异常的处理器进行复位,其它没有出现异常的处理器并不进行复位,从而有效地减少了设备复位所需的时间,达到了对设备进行快速复位的目的。

[0015] 下面,在详细描述本发明的各个实施例之前,首先简单介绍具有控制面处理器和数据面处理器的网络设备的常规启动和复位。

[0016] 网络设备通常使用 VxWorks 操作系统作为其操作系统。VxWorks 是一种灵活的、可裁剪的嵌入式实时操作系统。用户可以根据需要创建自己的 VxWorks 映像,其包括代码段、数据段和 BSS(Block Started by Symbol segment) 段。

[0017] 根据应用场合的不同,VxWorks 映像可以分为三种类型:可加载的 VxWorks 映像、基于 ROM(只读存储器)的 VxWorks 映像和驻留 ROM 的 VxWorks 映像。

[0018] 不同类型的 VxWorks 映像使用不同的启动代码(BOOTROM),但所有的启动代码都放置在 ROM 中。其中,用于可加载的 VxWorks 映像的启动代码包括起始引导程序(BootStrap Programs)和 ROM 引导程序(ROM Boot Programs),用于基于 ROM 的 VxWorks 映像的启动代码包括起始引导程序和基于 ROM 的 VxWorks 映像,以及驻留 ROM 的 VxWorks 映像的启动代码包括起始引导程序和驻留 ROM 的 VxWorks 映像。

[0019] 根据所采用的启动代码不同,网络设备的常规启动和复位也会有所不同。

[0020] 如果网络设备采用用于可加载的 VxWorks 映像的启动代码,则当网络设备进行常规的启动或复位时,网络设备的控制面处理器和数据面处理器按照如下步骤依次进行启动或复位(即首先控制面处理器进行启动或复位,然后在控制面处理器启动或复位完成后,数据面处理器进行启动或复位):首先执行放置在网络设备的 ROM 中的启动代码所包括的起始引导程序的指令,以将启动代码所包括的 ROM 引导程序从网络设备的 ROM 搬移到网络设备的 RAM(随机存取存储器);然后,执行已搬移到 RAM 的 ROM 引导程序的指令,以从网络接口或串口将 VxWorks 映像(包括代码段、数据段和 BSS 段)下载到网络设备的 RAM;最后,执行已下载到网络设备的 RAM 中的 VxWorks 映像的代码段的指令,以对下载到网络设备的 RAM 中的 VxWorks 映像的 BSS 段清零和执行系统初始化操作。

[0021] 如果网络设备采用用于基于 ROM 的 VxWorks 映像的启动代码,则当网络设备启动或复位时,网络设备的控制面处理器和数据面处理器按照如下步骤依次进行启动或复位:首先执行放置在网络设备的 ROM 中的启动代码所包括的起始引导程序的指令,以将启动代码中所包括的 VxWorks 映像(包括代码段、数据段和 BSS 段)从网络设备的 ROM 搬移到网络设备的 RAM;然后,执行已搬移到网络设备的 RAM 中的 VxWorks 映像的代码段的指令,以对搬移到网络设备的 RAM 中的 VxWorks 映像的 BSS 段清零和执行系统初始化操作。

[0022] 如果网络设备采用用于驻留 ROM 的 VxWorks 映像的启动代码,则当网络设备启动或复位时,网络设备的控制面处理器和数据面处理器按照如下步骤依次进行启动或复位:首先执行放置在网络设备的 ROM 中的启动代码所包括的起始引导程序的指令,以将启动代码中所包括的 VxWorks 映像的数据段和 BSS 段从网络设备的 ROM 搬移到网络设备的 RAM;然后执行网络设备的 ROM 中的 VxWorks 映像的代码段的指令,以对已搬移到网络设备的 RAM 中的 VxWorks 映像的 BSS 段清零和执行系统初始化操作。

[0023] 下面,以网络设备为例,结合附图详细描述本发明的各个实施例。

[0024] 现在参考图 1,其示出了按照本发明实施例一的对网络设备进行快速复位的方法流程图。这里,假设网络设备具有用于控制面处理的控制面处理器和用于数据面处理的数据面处理器。

[0025] 如图 1 所示,在步骤 S110 中,当网络设备的控制面处理器和数据面处理器的其中一个处理器 C 出现异常时,仅对处理器 C 进行常规复位。由于上面已经描述了常规复位的过程,所以这里省略了对处理器 C 进行常规复位的详细描述。

[0026] 从实施例一可以看出,由于只对网络设备中出现异常的处理器进行复位,而其它没有出现异常的处理器并不进行复位,从而有效地减少了网络设备复位所需的时间,达到了对网络设备进行快速复位的目的。

[0027] 参考图 2,图 2 其示出了按照本发明实施例二的对网络设备进行快速复位的方法流程图。与上述实施例一相比,在本实施例二中,也是仅对网络设备中出现异常的处理器进行复位,但是所进行的复位不是常规复位,而是对常规复位进行裁剪后的复位,以进一步减少网络设备复位所需的时间。这里,假设网络设备采用用于基于 ROM 的 VxWorks 映像的启动代码。

[0028] 在本实施例二中,当网络设备每次进行常规启动和复位时,在网络设备的每一个处理器将 VxWorks 映像的代码段和数据段从网络设备的 ROM 搬移到网络设备的 RAM 后,已搬移到网络设备的 RAM 的 VxWorks 映像的代码段被设置为只读的,以将其保护起来,同时已搬移到网络设备的 RAM 的 VxWorks 映像的数据段被备份。

[0029] 如图 2 所示,在步骤 S210,当网络设备的控制面处理器和数据面处理器中的其中一个处理器 P 出现异常时,促使处理器 P 对网络设备的 RAM 所存储的处理器 P 与网络设备的另一处理器交互的数据进行备份。

[0030] 在步骤 S220,促使处理器 P 跳过位于在网络设备的 ROM 中的启动代码所包括的起始引导程序的指令,直接从位于网络设备的 RAM 中的 VxWorks 映像的代码段的第一个指令开始执行。其中,位于网络设备的 RAM 中的 VxWorks 映像的代码段是上一次网络设备进行常规启动或复位时被加载到网络设备的 RAM 中的。

[0031] 在步骤 S230,在 VxWorks 映像的代码段中用于对网络设备的 RAM 所存储的 VxWorks 映像的 BSS 段进行清零的指令被处理器 P 执行后,促使处理器 P 利用网络设备的上一次常规启动或复位时所备份的 VxWorks 映像的数据段,来覆盖网络设备的 RAM 中的 VxWorks 映像的数据段。

[0032] 在步骤 S240,在 VxWorks 映像的代码段中用于进行系统初始化的指令被处理器 P 执行时,促使处理器 P 跳过用于对网络设备的外围芯片进行初始化的指令。

[0033] 在步骤 S250,在 VxWorks 映像的代码段中用于进行系统初始化的指令被处理器 P 执行后,促使处理器 P 在网络设备的 RAM 中恢复所备份的处理器 P 与网络设备的另一处理器所交互的数据。

[0034] 从上面的描述可以看出,在实施例二中,不但仅对网络设备中出现异常的处理器进行复位,而且该出现异常的处理器的复位并不执行将 VxWorks 映像从网络设备的 ROM 加载到网络设备的 RAM 的过程,而是直接从上一次网络设备进行常规启动或复位时已加载到网络设备的 RAM 中的 VxWorks 映像的代码段的第一个指令开始执行,并且跳过用于对网络

设备的外围芯片进行初始化的指令,因此,与实施例一的方案相比,采用实施例二的技术方案,网络设备的复位更加快速。

[0035] 本领域技术人员应当理解,本实施例二的技术方案也可以应用到采用用于可加载的 VxWorks 映像的启动代码和用于驻留 ROM 的 VxWorks 映像的启动代码的网络设备。在网络设备采用用于驻留 ROM 的 VxWorks 映像的启动代码的情况下, VxWorks 映像的代码段位于网络设备的 ROM 中,而不是位于网络设备的 RAM 中。

[0036] 本领域技术人员应当理解,在实施例二中,当确保在对网络设备进行快速复位时不会对网络设备的 RAM 中的 VxWorks 映像的数据段修改的情况下,也可以不需要对已搬移到网络设备的 RAM 的 VxWorks 映像的数据段进行备份以及利用所备份的 VxWorks 映像的数据段来覆盖网络设备的 RAM 中的 VxWorks 映像的数据段。

[0037] 本领域技术人员应当理解,在实施例二中,当在对网络设备进行快速复位时网络设备的 RAM 所存储的处理器 P 与网络设备的另一处理器交互的数据不会被修改的情况下,也可以不需要对网络设备的 RAM 所存储的处理器 P 与网络设备的另一处理器交互的数据进行备份以及在系统初始化后恢复该备份的数据。

[0038] 本领域技术人员应当理解,虽然在实施例二中,在系统初始化时跳过了用于对网络设备的外围芯片进行初始化的指令,但是本发明并不局限于此。在本发明的其它实施例中,也可以在系统初始化时执行用于对网络设备的外围芯片进行初始化的指令。在这种情况下,该技术方案与实施例二相比复位速度会慢一些,但是与现有技术和实施例一相比,复位速度仍然会快很多。

[0039] 本领域技术人员应当理解,本发明的各个实施例所公开的技术方案不但适用于具有两个处理器的网络设备,也适用于具有两个以上处理器的网络设备。

[0040] 本领域技术人员应当理解,本发明的各个实施例所公开的技术方案不但适用于网络设备,也适用于具有多个处理器的其它设备。

[0041] 本领域技术人员应当理解,本发明的各个实施例所公开的技术方案不但适用于 VxWorks 操作系统,也适用于其它类型的操作系统。

[0042] 参考图 3,其示出了按照本发明一个实施例的用于快速复位设备的装置。如图 3 所示,用于快速复位设备的装置 300 包括复位模块 310,用于当设备的多个处理器的其中一个处理器出现异常时,仅对该出现异常的处理器进行复位。

[0043] 其中,复位模块 310 可以进一步包括执行模块,用于在对该出现异常的处理器进行复位时,促使该出现异常的处理器直接从操作系统映像的代码段的第一个指令开始执行。

[0044] 其中,复位模块 310 还可以包括覆盖模块,用于在该操作系统映像的代码段中用于对该操作系统映像的 BSS 段进行清零的指令被该出现异常的处理器执行后,促使该出现异常的处理器使用以前备份的该操作系统映像的数据段来覆盖该设备的随机存取存储器所存储的该操作系统映像的数据段,其中,该以前备份的该操作系统映像的数据段可以是在该设备的上一次常规启动或复位时对从该设备的只读存储器搬移到该随机存取存储器的该操作系统映像的数据段进行备份而得到的。

[0045] 其中,复位模块 310 还可以包括备份模块和恢复模块,其中,备份模块用于在促使该出现异常的处理器直接从该操作系统映像的代码段的第一个指令开始执行之前,促使该

出现异常的处理器备份该随机存取存储器所存储的该出现异常的处理器与该多个处理器的其它处理器交互的数据,恢复模块用于在该操作系统映像的代码段中用于进行系统初始化的指令被该出现异常的处理器执行之后,促使该出现异常的处理器在该随机存取存储器中恢复所备份的该出现异常的处理器与该其它处理器交互的数据。

[0046] 其中,复位模块 310 还可以包括跳过模块,用于在该操作系统映像的代码段中用于进行系统初始化的指令被该出现异常的处理器执行时,促使该出现异常的处理器跳过用于对该设备的外围芯片进行初始化的指令。

[0047] 其中,该设备可以是网络设备,该多个处理器可以包括用于控制面处理的处理器和用于数据面处理的处理器。

[0048] 本领域技术人员应当理解,装置 300 中的各个模块可以利用软件、硬件或者软硬件结合的方式来实现。

[0049] 本发明实施例还提供机器可读存储介质,其存储机器可执行指令,当该机器可执行指令被执行时使得机器执行一种用于快速复位设备的方法,其包括以下步骤:当该设备的多个处理器的其中一个处理器出现异常时,仅对该出现异常的处理器进行复位。

[0050] 其中,该复位步骤可以进一步包括:在对该出现异常的处理器进行复位时,促使该出现异常的处理器直接从操作系统映像的代码段的第一个指令开始执行。

[0051] 其中,该复位步骤还可以包括:在该操作系统映像的代码段中用于对该操作系统映像的 BSS 段进行清零的指令被该出现异常的处理器执行后,促使该出现异常的处理器使用以前备份的该操作系统映像的数据段来覆盖该设备的随机存取存储器所存储的该操作系统映像的数据段,其中,该以前备份的该操作系统映像的数据段可以是在该设备的上一次常规启动或复位时对从该设备的只读存储器搬移到该随机存取存储器的该操作系统映像的数据段进行备份而得到的。

[0052] 其中,该复位步骤还可以包括:在促使该出现异常的处理器直接从该操作系统映像的代码段的第一个指令开始执行之前,促使该出现异常的处理器备份该随机存取存储器所存储的该出现异常的处理器与该多个处理器的其它处理器交互的数据;在该操作系统映像的代码段中用于进行系统初始化的指令被该出现异常的处理器执行之后,促使该出现异常的处理器在该随机存取存储器中恢复所备份的该出现异常的处理器与该其它处理器交互的数据。

[0053] 其中,该复位步骤还可以包括:在该操作系统映像的代码段中用于进行系统初始化的指令被该出现异常的处理器执行时,促使该出现异常的处理器跳过用于对该设备的外围芯片进行初始化的指令。

[0054] 其中,该设备可以是网络设备,该多个处理器可以包括用于控制面处理的处理器和用于数据面处理的处理器。

[0055] 本领域技术人员应当理解,本发明的各个实施例所公开的技术方案可以在不偏离发明实质的情况下做出各种变形和改变,并且这些变形和改变都应当落入在本发明的保护范围内。因此,本发明的保护范围由所附的权利要求书来定义。

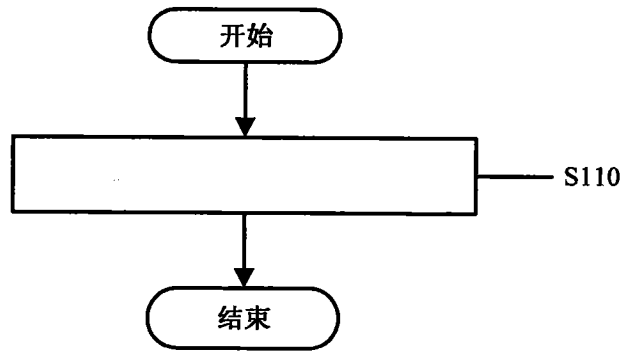


图 1

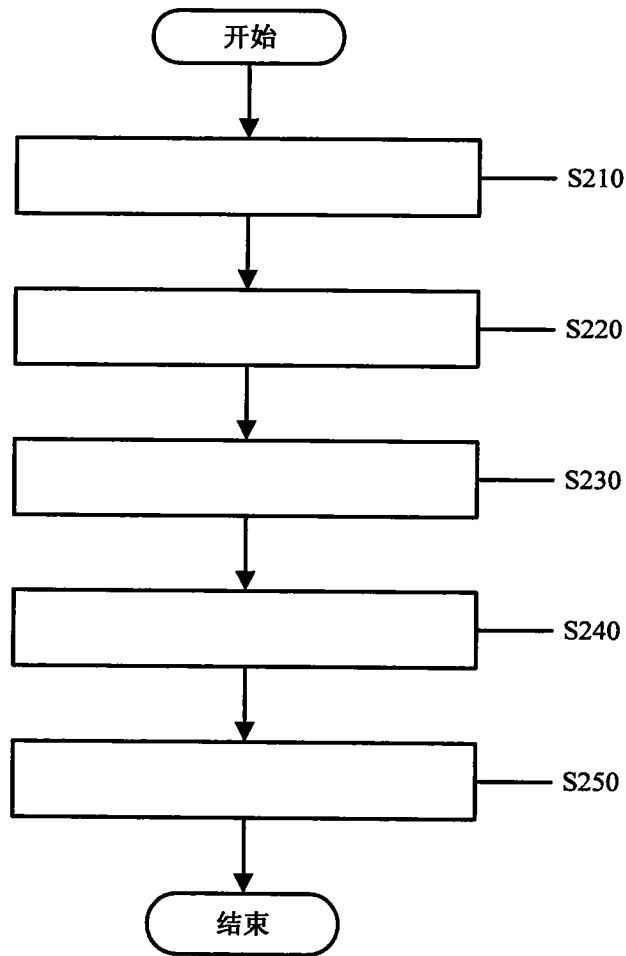


图 2

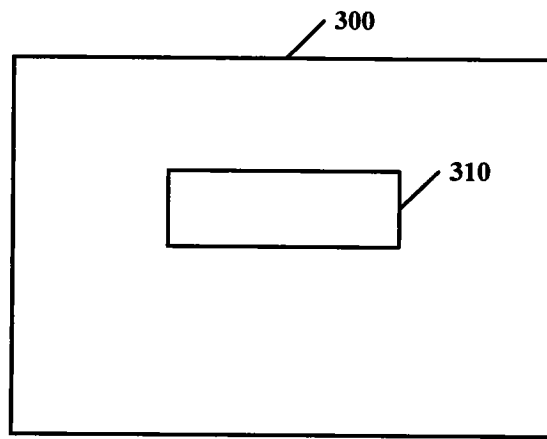


图 3