



(12) 发明专利申请

(10) 申请公布号 CN 115858024 A

(43) 申请公布日 2023. 03. 28

(21) 申请号 202211426262.5

(22) 申请日 2022.11.14

(71) 申请人 北京物芯科技有限责任公司
地址 100089 北京市海淀区知春路1号1幢
15层1514室

(72) 发明人 唐锋

(74) 专利代理机构 北京华圣典睿知识产权代理
有限公司 11510
专利代理师 刘文静

(51) Int. Cl.
G06F 9/4401 (2018.01)

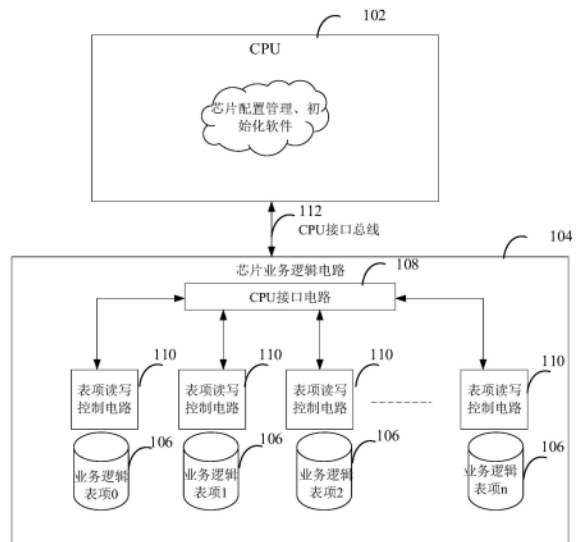
权利要求书2页 说明书9页 附图3页

(54) 发明名称

芯片启动的初始化方法、装置、计算机设备和存储介质

(57) 摘要

本申请涉及一种芯片启动的初始化方法、装置、计算机设备和存储介质。所述方法包括：所述芯片设置有初始化状态寄存器、以及与每个芯片表项对应的初始化控制组件；当检测到芯片启动时，通过所述初始化控制组件对所述芯片中对应的芯片表项进行初始化配置；查询所述初始化状态寄存器的参数值；根据所述初始化状态寄存器的参数值确认所述芯片是否完成初始化。采用本方法能够提高了芯片的启动速度。



1. 一种芯片启动的初始化方法,其特征在于,所述芯片设置有初始化状态寄存器、以及与每个芯片表项对应的初始化控制组件;所述方法包括:

当检测到芯片启动时,通过所述初始化控制组件对所述芯片中对应的芯片表项进行初始化配置;

查询所述初始化状态寄存器的参数值;

根据所述初始化状态寄存器的参数值确认所述芯片是否完成初始化。

2. 根据权利要求1所述的方法,其特征在于,所述初始化控制组件包括表项读写控制电路、表项初始化控制电路和表项初始化状态寄存器;

所述通过所述初始化控制组件对所述芯片中对应的芯片表项进行初始化配置包括:

通过各个所述芯片表项对应的所述初始化控制组件对各自对应的所述芯片表项进行初始化写操作;

当所述芯片表项的初始化写操作完成后,通过对应的所述表项初始化控制电路将所述表项初始化状态寄存器的参数值更新为初始化完成标识。

3. 根据权利要求2所述的方法,其特征在于,所述通过各个所述芯片表项对应的所述初始化控制组件对各自对应的所述芯片表项进行初始化写操作包括:

通过各个所述芯片表项对应的所述初始化控制组件的所述表项初始化状态寄存器,向对应组件内的所述表项初始化控制电路发送初始化指令;所述初始化指令用于指示所述表项初始化状态寄存器对应的芯片表项的初始化写操作;

通过所述初始化控制电路接收所述初始化指令,控制对应组件内的所述表项读写控制电路向随机存取存储器发起初始化写操作。

4. 根据权利要求2所述的方法,其特征在于,所述方法还包括:

通过每个所述芯片表项对应的表项初始化状态寄存器传输更新后的参数值至所述表项初始化状态寄存器;

将每个所述表项初始化状态寄存器的参数值进行逻辑或操作,得到初始化状态标识;

将所述初始化状态寄存器的参数值更新为所述初始化状态标识。

5. 根据权利要求1所述的方法,其特征在于,所述根据所述初始化状态寄存器的参数值确认所述芯片是否完成初始化包括:

当所述参数值为初始化完成标识时,确定所述芯片完成初始化;

当所述参数值为非初始化完成标识时,确定所述芯片没有完成初始化。

6. 根据权利要求1所述的方法,其特征在于,所述通过所述初始化控制组件对所述芯片中对应的芯片表项进行初始化配置包括:

从多个所述芯片表项中确定待初始化的芯片表项作为目标芯片表项;

向所述目标芯片表项对应的表项初始化状态寄存器中写入非初始化完成标识;

通过所述目标芯片表项对应的初始化控制组件,对所述目标芯片表项进行初始化配置。

7. 根据权利要求6所述的方法,其特征在于,所述方法还包括:

查询所述目标芯片表项的初始化控制组件中的初始化寄存器的参数值;

根据当前所述初始化状态寄存器的参数值判断所述目标芯片表项是否完成初始化。

8. 一种芯片启动的初始化装置,所述芯片设置有表项初始化状态寄存器、以及与每个

芯片表项对应的初始化控制组件;其特征在于,所述装置包括:

初始化配置模块,用于当检测到芯片启动时,通过所述初始化控制组件对所述芯片中对应的芯片表项进行初始化配置;

寄存器参数值查询模块,用于查询所述初始化状态寄存器的参数值;

芯片初始化确认模块,用于根据所述初始化状态寄存器的参数值确认所述芯片是否完成初始化。

9.一种计算机设备,包括存储器和处理器,所述存储器存储有计算机程序,其特征在于,所述处理器执行所述计算机程序时实现权利要求1至7中任一项所述方法的步骤。

10.一种芯片,应用于权利要求1至7中任一项所述的方法中,其特征在于,所述芯片包括一个或多个芯片表项、初始化状态寄存器、以及与每个芯片表项对应的初始化控制组件;所述初始化控制组件包括表项读写控制电路、表项初始化控制电路和表项初始化状态寄存器。

芯片启动的初始化方法、装置、计算机设备和存储介质

技术领域

[0001] 本申请涉及芯片技术领域,特别是涉及一种芯片启动的初始化方法、装置、计算机设备和存储介质。

背景技术

[0002] 在以太网交换芯片中,一般设计有大量的表项,比如配置表项或转发业务表项,表项的数目可能达到数百个,且每个表项的容量可以达到数千条或上万条。

[0003] 由于芯片中的大量表项一般使用随机存取存储器(RAM)实现,然而RAM在芯片上电解除复位后,RAM里面存储的内容是不确定值(即X态)。因此,芯片在上电启动后,一般需要中央处理器CPU对芯片中各个表项进行初始化配置。

[0004] 目前,对芯片中的表项进行初始化的方式主要是由CPU对表项逐个进行初始化,而且各表项的初始过程是串行进行的,同时由于芯片需要初始化的表项及表项的容量比较大,所以初始过程需要较长的时间。在一些对芯片启动时间要求比较苛刻的场景下,现有的对芯片表项初始化的方法往往不能满足要求。

发明内容

[0005] 基于此,有必要针对上述技术问题,提供一种能够减少芯片启动时间、提高芯片表项初始化速度的芯片启动的初始化方法、装置、计算机设备和存储介质。

[0006] 一种芯片启动的初始化方法,所述芯片设置有初始化状态寄存器、以及与每个芯片表项对应的初始化控制组件;所述方法包括:

[0007] 当检测到芯片启动时,通过所述初始化控制组件对所述芯片中对应的芯片表项进行初始化配置;

[0008] 查询所述初始化状态寄存器的参数值;

[0009] 根据所述初始化状态寄存器的参数值确认所述芯片是否完成初始化。

[0010] 在一个实施例中,所述初始化控制组件包括表项读写控制电路、表项初始化控制电路和表项初始化状态寄存器;

[0011] 所述通过所述初始化控制组件对所述芯片中对应的芯片表项进行初始化配置包括:

[0012] 通过各个所述芯片表项对应的所述初始化控制组件对各自对应的所述芯片表项进行初始化写操作;

[0013] 当所述芯片表项的初始化写操作完成后,通过对应的所述表项初始化控制电路将所述表项初始化状态寄存器的参数值更新为初始化完成标识。

[0014] 在一个实施例中,所述通过各个所述芯片表项对应的所述初始化控制组件对各自对应的所述芯片表项进行初始化写操作包括:

[0015] 通过各个所述芯片表项对应的所述初始化控制组件的所述表项初始化状态寄存器,向对应组件内的所述表项初始化控制电路发送初始化指令;所述初始化指令用于指示

所述表项初始化状态寄存器对应的芯片表项的初始化写操作；

[0016] 通过所述初始化控制电路接收所述初始化指令，控制对应组件内的所述表项读写控制电路向随机存取存储器发起初始化写操作。

[0017] 在一个实施例中，所述方法还包括：

[0018] 通过每个所述芯片表项对应的表项初始化状态寄存器传输更新后的参数值至所述表项初始化状态寄存器；

[0019] 将每个所述表项初始化状态寄存器的参数值进行逻辑或操作，得到初始化状态标识；

[0020] 将所述初始化状态寄存器的参数值更新为所述初始化状态标识。

[0021] 在一个实施例中，所述根据所述初始化状态寄存器的参数值确认所述芯片是否完成初始化包括：

[0022] 当所述参数值为初始化完成标识时，确定所述芯片完成初始化；

[0023] 当所述参数值为非初始化完成标识时，确定所述芯片没有完成初始化。

[0024] 在一个实施例中，所述通过所述初始化控制组件对所述芯片中对应的芯片表项进行初始化配置包括：

[0025] 从多个所述芯片表项中确定待初始化的芯片表项作为目标芯片表项；

[0026] 向所述目标芯片表项对应的表项初始化状态寄存器中写入非初始化完成标识；

[0027] 通过所述目标芯片表项对应的初始化控制组件，对所述目标芯片表项进行初始化配置。

[0028] 在一个实施例中，所述方法还包括：

[0029] 查询所述目标芯片表项的初始化控制组件中的初始化寄存器的参数值；

[0030] 根据当前所述初始化状态寄存器的参数值判断所述目标芯片表项是否完成初始化。

[0031] 一种芯片启动的初始化装置，所述芯片设置有表项初始化状态寄存器、以及与每个芯片表项对应的初始化控制组件；所述装置包括：

[0032] 初始化配置模块，用于当检测到芯片启动时，通过所述初始化控制组件对所述芯片中对应的芯片表项进行初始化配置；

[0033] 寄存器参数值查询模块，用于查询所述初始化状态寄存器的参数值；

[0034] 芯片初始化确认模块，用于根据所述初始化状态寄存器的参数值确认所述芯片是否完成初始化。

[0035] 一种芯片，所述芯片包括一个或多个芯片表项、初始化状态寄存器、以及与每个芯片表项对应的初始化控制组件；所述初始化控制组件包括表项读写控制电路、表项初始化控制电路和表项初始化状态寄存器。

[0036] 一种计算机设备，包括存储器和处理器，所述存储器存储有计算机程序，所述处理器执行所述计算机程序时实现上述实施例中芯片启动的初始化方法的步骤。

[0037] 一种计算机可读存储介质，其上存储有计算机程序，所述计算机程序被处理器执行时实现上述实施例中芯片启动的初始化方法的步骤。

[0038] 上述芯片启动的初始化方法、装置、计算机设备和存储介质，通过在芯片内设置表项初始化状态寄存器、并针对芯片内的每个芯片表项设置对应的初始化控制组件，在芯片

上电启动时,CPU控制可以同时控制每个芯片表项的初始化控制组件对芯片表项进行初始化配置,并且各个芯片表项的初始化配置可以并行进行,以此提高了芯片表项的初始化速度;进一步,CPU通过查询表项初始化状态寄存器的参数值来确定各个芯片表项的初始化配置是否都完成,从而判定芯片的初始化是否完成,提高了芯片的启动速度。并且,CPU还可以针对指定的一个或多个芯片表项进行初始化配置,进一步提高了芯片的启动速度,也可以定制化对芯片表项进行初始化配置。

附图说明

- [0039] 图1为一个实施例中芯片启动的初始化方法的应用场景图;
- [0040] 图2为另一个实施例中芯片启动的初始化方法的应用场景图;
- [0041] 图3为一个实施例中芯片启动的初始化方法的流程示意图;
- [0042] 图4为一个实施例中芯片启动的初始化装置的结构框图;
- [0043] 图5为一个实施例中计算机设备的内部结构图。

具体实施方式

[0044] 为了使本申请的目的、技术方案及优点更加清楚明白,以下结合附图及实施例,对本申请进行进一步详细说明。应当理解,此处描述的具体实施例仅仅用以解释本申请,并不用于限定本申请。

[0045] 在以太网交换芯片中,一般设置有大量的配置表项或转发业务表项,表项的数量巨大。在一个实施例中,如图1所示,示出了一个实施例中芯片启动的初始化方法应用场景图。从图中可以看出,该芯片启动的初始化系统包括中央处理器CPU102和芯片104。其中,CPU 102和芯片104通过CPU接口总线112连接。其中芯片104内设置有一个或多个芯片表项106、CPU接口电路108、以及与芯片表项对应的表项读写控制电路110。其中,芯片表项106如图中所示的业务逻辑表项0~业务逻辑表项n。

[0046] 每个芯片表项对应设置有一个表项读写控制电路,各个表项读写控制电路与CPU接口电路连接。CPU通过表项读写控制电路对对应的芯片表项进行初始化配置,CPU对表项逐个进行初始化,而且各表项的初始过程是串行进行的,同时由于芯片需要初始化的表项及表项的容量比较大,所以初始过程需要较长的时间。在一些对芯片启动时间要求比较苛刻的场景下,现有的对芯片表项初始化的方法往往不能满足要求。

[0047] 在一个实施例中,本申请提供的芯片启动的初始化方法,可以应用于如图2所示的应用环境中。该芯片启动的初始化方法应用于芯片启动的初始化系统中,该系统包括中央处理器CPU 202和芯片204;其中芯片204内设置一个或多个芯片表项206、初始化状态寄存器208、初始化控制组件210、CPU接口电路212。其中,CPU 202和芯片204通过CPU接口总线214连接。其中,CPU202内还设置有芯片配置管理程序、初始化软件等。每个芯片表项206对应设置有初始化控制组件210,通过初始化控制组件210给对应的芯片表项206进行初始化配置。

[0048] 初始化控制组件210包括表项读写控制电路2101、表项初始化控制电路2202和表项初始化状态寄存器2103。其中,对于每个芯片表项对应的初始化控制组件210,表项初始化控制电路2202分别与表项读写控制电路2101和表项初始化状态寄存器2103连接。

[0049] 每个芯片表项206对应的表项读写控制电路2101、以及表项初始化状态寄存器2103分别与CPU接口电路212连接。CPU接口电路212还与表项初始化状态寄存器208连接。

[0050] 每个芯片表项206对应的表项初始化状态寄存器2103与表项初始化状态寄存器208连接；且表项初始化状态寄存器208具有逻辑或的关系，即可以将每个表项初始化状态寄存器2103的参数值进行逻辑或操作得到计算结果，并将计算结果作为芯片的初始化状态标识存储在表项初始化状态寄存器208中。

[0051] 其中，表项初始化状态寄存器2103还包括芯片复位信号。当芯片上电启动时，此时CPU的初始化软件对初始化控制组件210发送初始化指令，指示初始化控制组件210对对应的芯片表项206进行初始化配置。在初始化控制组件210内部，当芯片204上电启动时，此时各个芯片表项206对应的表项初始化状态寄存器2103的芯片复位信号解复位后，表项初始化状态寄存器2103向对应的表项初始化控制电路2202发送初始化指令，以指示表项初始化状态寄存器2103所对应的芯片表项206的初始写操作。

[0052] CPU202和芯片204可以设置在终端或服务器内，终端可以但不限于各种个人计算机、笔记本电脑、智能手机、平板电脑和便携式可穿戴设备，服务器可以用独立的服务器或者是多个服务器组成的服务器集群来实现。

[0053] 在一个实施例中，在芯片中，采用了本方案的芯片启动初始化方法，可以支持硬件辅助自动初始化，减少了芯片启动过程中软件初始时间，确保芯片快速启动，满足苛刻的芯片启动时间要求。

[0054] 在一个实施例中，如图3所示，提供了一种芯片启动的初始化方法，以该方法应用于图1中的CPU 202为例进行说明，包括以下步骤：

[0055] 步骤S302，当检测到芯片启动时，通过所述初始化控制组件对所述芯片中对应的芯片表项进行初始化配置。

[0056] 其中，初始化配置是指对芯片表项进行初始内容配置，比如初始化写操作，对芯片表项写入初始内容，初始内容比如0、1或特定序列中的一种。

[0057] 芯片内包括一个或多个芯片表项，对应每个芯片表项设置有对应的初始化控制组件；其中初始化控制组件包括表项读写控制电路、表项初始化控制电路和表项初始化状态寄存器。表项初始化控制电路分别与表项读写控制电路和表项初始化状态寄存器连接；表项读写控制电路和表项初始化状态寄存器分别与CPU接口电路连接。

[0058] 具体地，当芯片上电启动时，芯片内的芯片复位信号解复位后，CPU通过初始化控制组件对芯片中对应的芯片表项进行初始化配置。即CPU通过CPU接口总线向CPU接口电路发送初始化配置指令，CPU接口电路将初始化配置指令分发给各个芯片表项对应的初始化控制组件，已用于指示芯片内对应芯片表项的初始化控制组件对芯片表项进行初始化配置。

[0059] 在一个实施例中，所述初始化控制组件包括表项读写控制电路、表项初始化控制电路和表项初始化状态寄存器；所述通过所述初始化控制组件对所述芯片中对应的芯片表项进行初始化配置包括：通过各个所述芯片表项对应的所述初始化控制组件对各自对应的所述芯片表项进行初始化写操作；当所述芯片表项的初始化写操作完成后，通过对应的所述表项初始化控制电路将所述表项初始化状态寄存器的参数值更新为初始化完成标识。

[0060] 在一个实施例中，所述通过各个所述芯片表项对应的所述初始化控制组件对各自

对应的所述芯片表项进行初始化写操作包括：通过各个所述芯片表项对应的所述初始化控制组件的所述表项初始化状态寄存器，向对应组件内的所述表项初始化控制电路发送初始化指令；所述初始化指令用于指示所述表项初始化状态寄存器对应的芯片表项的初始化写操作；通过所述初始化控制电路接收所述初始化指令，控制对应组件内的所述表项读写控制电路向随机存取存储器发起初始化写操作。

[0061] 其中，随机存取存储器RAM在芯片上电解除复位后，RAM里面存储的内容是不确定值，也就是常说的X态，因此，在芯片启动后，才需要对芯片表项进行初始化配置。

[0062] 当完成对应芯片表项的初始化写操作后，通过表项初始化控制电路将对应芯片表项、即属于同一初始化控制组件内的表项初始化状态寄存器的参数值更新为初始化完成标识。并通过表项初始化状态寄存器将自身存储的寄存器的参数值传输给初始化状态寄存器。其中，初始化完成标识用于指示芯片表项的初始化操作已经完成，比如可以用参数值0来标识，也可以用参数值1来标识。

[0063] 当芯片内所有芯片表项的初始化写操作完成后，初始化状态寄存器将接收到各个表项初始化状态寄存器的参数值，将各个参数值进行逻辑或操作，得到计算值，并将得到的计算值存储在初始化状态寄存器中作为初始化状态标识，CPU通过查询初始化状态寄存器的参数值即可以确认所有芯片表项是否都完成初始化配置。

[0064] 步骤S304，查询所述初始化状态寄存器的参数值。

[0065] CPU查询初始化状态寄存器的参数值，根据参数值来确定各个芯片表项是否完成初始化写操作，从而确定芯片是否完成初始化。

[0066] 步骤S306，根据所述表项初始化状态寄存器的参数值确认所述芯片是否完成初始化。

[0067] 在一个实施例中，所述方法还包括：通过每个所述芯片表项对应的表项初始化状态寄存器，传输更新后的参数值至所述初始化状态寄存器；将每个所述表项初始化状态寄存器的参数值进行逻辑或操作，得到初始化状态标识；将所述初始化状态寄存器的参数值更新为所述初始化状态标识。

[0068] 其中，初始化状态标识包括初始化完成标识和非初始化完成标识。

[0069] 在一个实施例中，所述根据所述初始化状态寄存器的参数值确认所述芯片是否完成初始化包括：当所述参数值为初始化完成标识时，确定所述芯片完成初始化；当所述参数值为非初始化完成标识时，确定所述芯片没有完成初始化。

[0070] 在一个实施例中，所述通过所述初始化控制组件对所述芯片中对应的芯片表项进行初始化配置包括：从多个所述芯片表项中确定待初始化的芯片表项作为目标芯片表项；向所述目标芯片表项对应的表项初始化状态寄存器中写入非初始化完成标识；通过所述目标芯片表项对应的初始化控制组件，对所述目标芯片表项进行初始化配置。CPU不仅可以对所有芯片表项进行初始化配置，还可以对指定的芯片表项进行初始化配置。具体地，CPU从多个芯片表项中确定待初始化的目标芯片表项，然后向目标芯片表项的表项初始化状态寄存器中写入非初始化完成标识，以此来指示对目标芯片表项进行初始化配置。其中，目标芯片表项可以是一个或多个，目标芯片表项的初始化配置可以是并行处理，也可以按序进行。进而，通过目标芯片表项对应的初始化控制组件对目标芯片表项进行初始化配置。其中，目标芯片表项的初始配置步骤与上述实施例中描述的一般芯片表项的初始化配置的步骤一

样,在此不再重复描述。

[0071] 在一个实施例中,当目标芯片的初始化配置完成后,所述方法还包括:CPU查询所述目标芯片表项的初始化控制组件中的初始化寄存器的参数值;根据当前所述初始化寄存器的参数值判断该芯片表项是否完成初始化。如果目标芯片的当前的初始化状态寄存器的参数值为初始化完成标识,则表明目标芯片表项完成初始化配置;如果目标芯片的当前的初始化状态寄存器的参数值为非初始化完成标识,则表明目标芯片表项没有完成初始化配置。

[0072] 上述芯片启动的初始化方法中,通过在芯片内设置表项初始化状态寄存器、并针对芯片内的每个芯片表项设置对应的初始化控制组件,在芯片上电启动时,CPU控制可以同时控制每个芯片表项的初始化控制组件对芯片表项进行初始化配置,并且各个芯片表项的初始化配置可以并行进行,以此提高了芯片表项的初始化速度;进一步,CPU通过查询表项初始化状态寄存器的参数值来确定各个芯片表项的初始化配置是否都完成,从而判定芯片的初始化是否完成,降低了芯片的启动时间。并且,CPU还可以针对指定的一个或多个芯片表项进行初始化配置,进一步提高了芯片的启动速度。

[0073] 应该理解的是,虽然图3的流程图中的各个步骤按照箭头的指示依次显示,但是这些步骤并不是必然按照箭头指示的顺序依次执行。除非本文中有明确的说明,这些步骤的执行并没有严格的顺序限制,这些步骤可以以其它的顺序执行。而且,图3中的至少一部分步骤可以包括多个子步骤或者多个阶段,这些子步骤或者阶段并不必然是在同一时刻执行完成,而是可以在不同的时刻执行,这些子步骤或者阶段的执行顺序也不必然是依次进行,而是可以与其它步骤或者其它步骤的子步骤或者阶段的至少一部分轮流或者交替地执行。

[0074] 在一个实施例中,如图4所示,提供了一种芯片启动的初始化装置400,包括:初始化配置模块402、寄存器参数值查询模块404和芯片初始化确认模块406,其中:

[0075] 初始化配置模块402,用于当检测到芯片启动时,通过所述初始化控制组件对所述芯片中对应的芯片表项进行初始化配置。

[0076] 寄存器参数值查询模块404,用于查询所述初始化状态寄存器的参数值。

[0077] 芯片初始化确认模块406,用于根据所述初始化状态寄存器的参数值确认所述芯片是否完成初始化。

[0078] 在一个实施例中,初始化配置模块还用于通过各个所述芯片表项对应的所述初始化控制组件对各自对应的所述芯片表项进行初始化写操作;当所述芯片表项的初始化写操作完成后,通过对应的所述表项初始化控制电路将所述表项初始化状态寄存器的参数值更新为初始化完成标识。

[0079] 在一个实施例中,初始化配置模块还用于通过各个所述芯片表项对应的所述初始化控制组件的所述表项初始化状态寄存器,向对应组件内的所述表项初始化控制电路发送初始化指令;所述初始化指令用于指示所述表项初始化状态寄存器对应的芯片表项的初始化写操作;通过所述初始化控制电路接收所述初始化指令,控制对应组件内的所述表项读写控制电路向随机存取存储器发起初始化写操作。

[0080] 在一个实施例中,所述装置还包括初始化状态寄存器模块,用于通过每个所述芯片表项对应的表项初始化状态寄存器传输更新后的参数值至所述表项初始化状态寄存器;将每个所述表项初始化状态寄存器的参数值进行逻辑或操作,得到初始化状态标识;将所述

初始化状态寄存器的参数值更新为所述初始化状态标识。

[0081] 在一个实施例中,芯片初始化确认模块还用于当所述参数值为初始化完成标识时,确定所述芯片完成初始化;当所述参数值为非初始化完成标识时,确定所述芯片没有完成初始化。

[0082] 在一个实施例中,初始化配置模块还用于从多个所述芯片表项中确定待初始化的芯片表项作为目标芯片表项;向所述目标芯片表项对应的表项初始化状态寄存器中写入非初始化完成标识;通过所述目标芯片表项对应的初始化控制组件,对所述目标芯片表项进行初始化配置。

[0083] 在一个实施例中,初始化配置模块还用于查询所述目标芯片表项的初始化控制组件中的初始化寄存器的参数值;根据当前所述初始化状态寄存器的参数值判断所述目标芯片表项是否完成初始化。

[0084] 上述芯片启动的初始化装置中,通过在芯片内设置表项初始化状态寄存器、并针对芯片内的每个芯片表项设置对应的初始化控制组件,在芯片上电启动时,CPU控制可以同时控制每个芯片表项的初始化控制组件对芯片表项进行初始化配置,并且各个芯片表项的初始化配置可以并行进行,以此提高了芯片表项的初始化速度;进一步,CPU通过查询表项初始化状态寄存器的参数值来确定各个芯片表项的初始化配置是否都完成,从而判定芯片的初始化是否完成,降低了芯片的启动时间。并且,CPU还可以针对指定的一个或多个芯片表项进行初始化配置,进一步提高了芯片的启动速度。

[0085] 关于芯片启动的初始化装置的具体限定可以参见上文中对于芯片启动的初始化方法的限定,在此不再赘述。上述芯片启动的初始化装置中的各个模块可全部或部分通过软件、硬件及其组合来实现。上述各模块可以硬件形式内嵌于或独立于计算机设备中的处理器中,也可以以软件形式存储于计算机设备中的存储器中,以便于处理器调用执行以上各个模块对应的操作。

[0086] 在一个实施例中,提供了一种计算机设备,该计算机设备可以是终端,终端内可以设置有芯片,该芯片的结构可以设置为本方案中提供的。终端内部结构图可以如图5所示。该计算机设备包括通过系统总线连接的处理器、存储器、网络接口、显示屏和输入装置。其中,该计算机设备的处理器用于提供计算和控制能力。该计算机设备的存储器包括非易失性存储介质、内存储器。该非易失性存储介质存储有操作系统和计算机程序。该内存储器为非易失性存储介质中的操作系统和计算机程序的运行提供环境。该计算机设备的网络接口用于与外部的终端通过网络连接通信。该计算机程序被处理器执行时以实现一种芯片启动的初始化方法。该计算机设备的显示屏可以是液晶显示屏或者电子墨水显示屏,该计算机设备的输入装置可以是显示屏上覆盖的触摸层,也可以是计算机设备外壳上设置的按键、轨迹球或触控板,还可以是外接的键盘、触控板或鼠标等。

[0087] 本领域技术人员可以理解,图5中示出的结构,仅仅是与本申请方案相关的部分结构的框图,并不构成对本申请方案所应用于其上的计算机设备的限定,具体的计算机设备可以包括比图中所示更多或更少的部件,或者组合某些部件,或者具有不同的部件布置。

[0088] 在一个实施例中,提供了一种计算机设备,包括存储器和处理器,该存储器存储有计算机程序,该处理器执行计算机程序时实现以下步骤:当检测到芯片启动时,通过所述初始化控制组件对所述芯片中对应的芯片表项进行初始化配置;查询所述初始化状态寄存器

的参数值;根据所述初始化状态寄存器的参数值确认所述芯片是否完成初始化。

[0089] 在一个实施例中,处理器执行计算机程序时还实现以下步骤:通过各个所述芯片表项对应的所述初始化控制组件对各自对应的所述芯片表项进行初始化写操作;当所述芯片表项的初始化写操作完成后,通过对应的所述表项初始化控制电路将所述表项初始化状态寄存器的参数值更新为初始化完成标识。

[0090] 在一个实施例中,处理器执行计算机程序时还实现以下步骤:通过各个所述芯片表项对应的所述初始化控制组件的所述表项初始化状态寄存器,向对应组件内的所述表项初始化控制电路发送初始化指令;所述初始化指令用于指示所述表项初始化状态寄存器对应的芯片表项的初始化写操作;通过所述初始化控制电路接收所述初始化指令,控制对应组件内的所述表项读写控制电路向随机存取存储器发起初始化写操作。

[0091] 在一个实施例中,处理器执行计算机程序时还实现以下步骤:通过每个所述芯片表项对应的表项初始化状态寄存器传输更新后的参数值至所述表初始化状态寄存器;将每个所述表项初始化状态寄存器的参数值进行逻辑或操作,得到初始化状态标识;将所述初始化状态寄存器的参数值更新为所述初始化状态标识。

[0092] 在一个实施例中,处理器执行计算机程序时还实现以下步骤:当所述参数值为初始化完成标识时,确定所述芯片完成初始化;当所述参数值为非初始化完成标识时,确定所述芯片没有完成初始化。

[0093] 在一个实施例中,处理器执行计算机程序时还实现以下步骤:从多个所述芯片表项中确定待初始化的芯片表项作为目标芯片表项;向所述目标芯片表项对应的表项初始化状态寄存器中写入非初始化完成标识;通过所述目标芯片表项对应的初始化控制组件,对所述目标芯片表项进行初始化配置。

[0094] 在一个实施例中,处理器执行计算机程序时还实现以下步骤:查询所述目标芯片表项的初始化控制组件中的初始化寄存器的参数值;根据当前所述初始化状态寄存器的参数值判断所述目标芯片表项是否完成初始化。

[0095] 在一个实施例中,在一个实施例中,提供了一种计算机可读存储介质,其上存储有计算机程序,计算机程序被处理器执行时实现以下步骤:当检测到芯片启动时,通过所述初始化控制组件对所述芯片中对应的芯片表项进行初始化配置;查询所述初始化状态寄存器的参数值;根据所述初始化状态寄存器的参数值确认所述芯片是否完成初始化。

[0096] 在一个实施例中,计算机程序被处理器执行时还实现以下步骤:通过各个所述芯片表项对应的所述初始化控制组件对各自对应的所述芯片表项进行初始化写操作;当所述芯片表项的初始化写操作完成后,通过对应的所述表项初始化控制电路将所述表项初始化状态寄存器的参数值更新为初始化完成标识。

[0097] 在一个实施例中,计算机程序被处理器执行时还实现以下步骤:通过各个所述芯片表项对应的所述初始化控制组件的所述表项初始化状态寄存器,向对应组件内的所述表项初始化控制电路发送初始化指令;所述初始化指令用于指示所述表项初始化状态寄存器对应的芯片表项的初始化写操作;通过所述初始化控制电路接收所述初始化指令,控制对应组件内的所述表项读写控制电路向随机存取存储器发起初始化写操作。

[0098] 在一个实施例中,计算机程序被处理器执行时还实现以下步骤:通过每个所述芯片表项对应的表项初始化状态寄存器传输更新后的参数值至所述表初始化状态寄存器;将

每个所述表项初始化状态寄存器的参数值进行逻辑或操作,得到初始化状态标识;将所述初始化状态寄存器的参数值更新为所述初始化状态标识。

[0099] 在一个实施例中,计算机程序被处理器执行时还实现以下步骤:当所述参数值为初始化完成标识时,确定所述芯片完成初始化;当所述参数值为非初始化完成标识时,确定所述芯片没有完成初始化。

[0100] 在一个实施例中,计算机程序被处理器执行时还实现以下步骤:从多个所述芯片表项中确定待初始化的芯片表项作为目标芯片表项;向所述目标芯片表项对应的表项初始化状态寄存器中写入非初始化完成标识;通过所述目标芯片表项对应的初始化控制组件,对所述目标芯片表项进行初始化配置。

[0101] 在一个实施例中,计算机程序被处理器执行时还实现以下步骤:查询所述目标芯片表项的初始化控制组件中的初始化寄存器的参数值;根据当前所述初始化状态寄存器的参数值判断所述目标芯片表项是否完成初始化。

[0102] 上述芯片启动的初始化计算机设备和存储介质中,通过在芯片内设置表项初始化状态寄存器、并针对芯片内的每个芯片表项设置对应的初始化控制组件,在芯片上电启动时,CPU控制可以同时控制每个芯片表项的初始化控制组件对芯片表项进行初始化配置,并且各个芯片表项的初始化配置可以并行进行,以此提高了芯片表项的初始化速度;进一步,CPU通过查询表项初始化状态寄存器的参数值来确定各个芯片表项的初始化配置是否都完成,从而判定芯片的初始化是否完成,降低了芯片的启动时间。并且,CPU还可以针对指定的一个或多个芯片表项进行初始化配置,进一步提高了芯片的启动速度。

[0103] 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来完成,所述的计算机程序可存储于一非易失性计算机可读存储介质中,该计算机程序在执行时,可包括如上述各方法的实施例的流程。其中,本申请所提供的各实施例中所使用的对存储器、存储、数据库或其它介质的任何引用,均可包括非易失性和/或易失性存储器。非易失性存储器可包括只读存储器(ROM)、可编程ROM(PROM)、电可编程ROM(EPROM)、电可擦除可编程ROM(EEPROM)或闪存。易失性存储器可包括随机存取存储器(RAM)或者外部高速缓冲存储器。作为说明而非局限,RAM以多种形式可得,诸如静态RAM(SRAM)、动态RAM(DRAM)、同步DRAM(SDRAM)、双数据率SDRAM(DDRSDRAM)、增强型SDRAM(ESDRAM)、同步链路(Synchlink)DRAM(SLDRAM)、存储器总线(Rambus)直接RAM(RDRAM)、直接存储器总线动态RAM(DRDRAM)、以及存储器总线动态RAM(RDRAM)等。

[0104] 以上实施例的各技术特征可以进行任意的组合,为使描述简洁,未对上述实施例中的各个技术特征所有可能的组合都进行描述,然而,只要这些技术特征的组合不存在矛盾,都应当认为是本说明书记载的范围。

[0105] 以上所述实施例仅表达了本申请的几种实施方式,其描述较为具体和详细,但并不能因此而理解为对发明专利范围的限制。应当指出的是,对于本领域的普通技术人员来说,在不脱离本申请构思的前提下,还可以做出若干变形和改进,这些都属于本申请的保护范围。因此,本申请专利的保护范围应以所附权利要求为准。

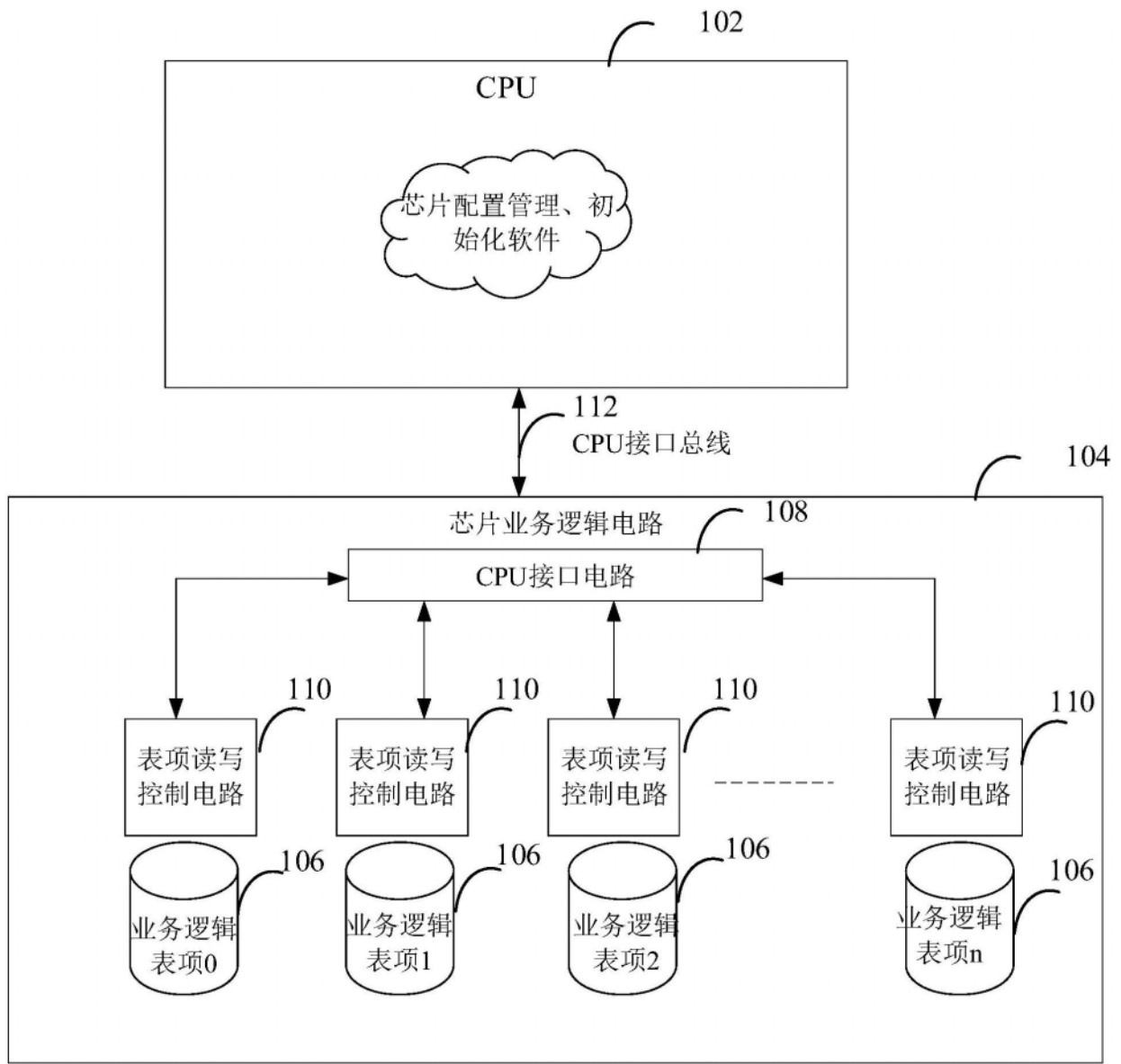


图1

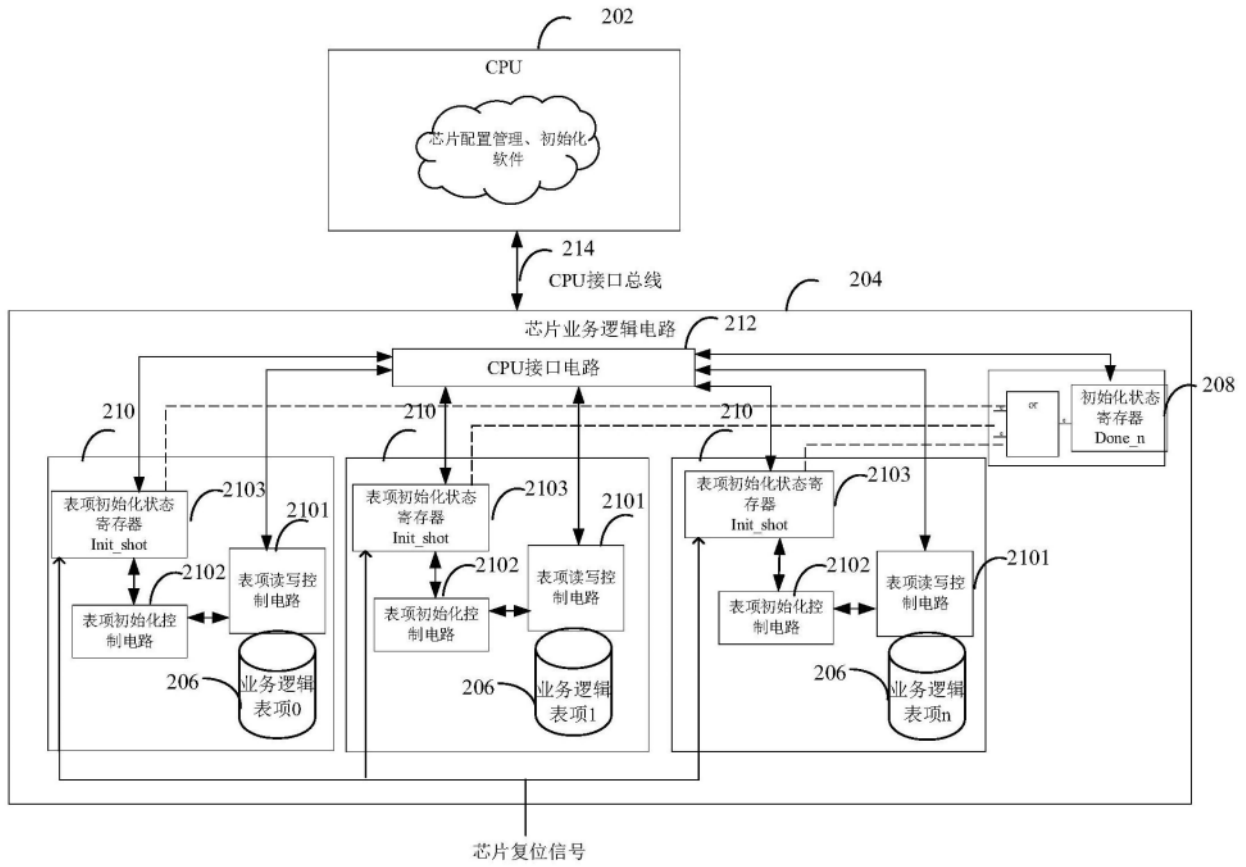


图2

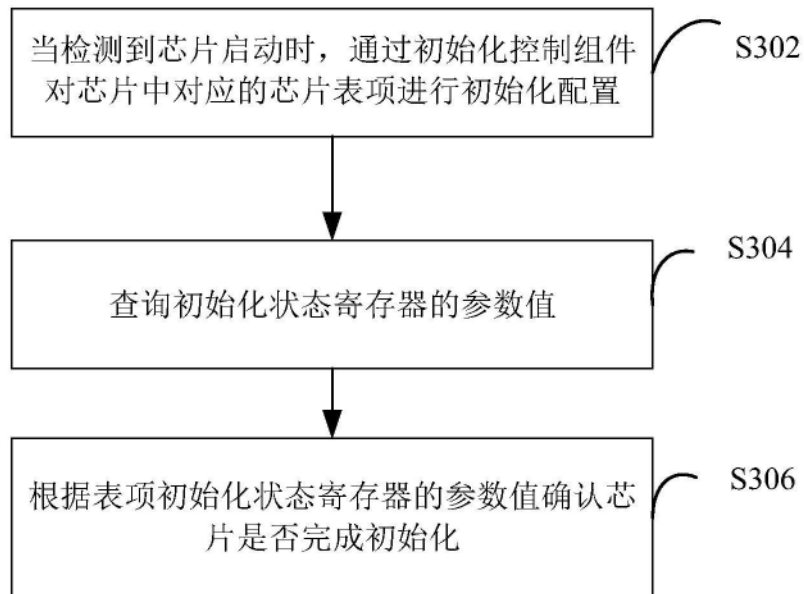


图3

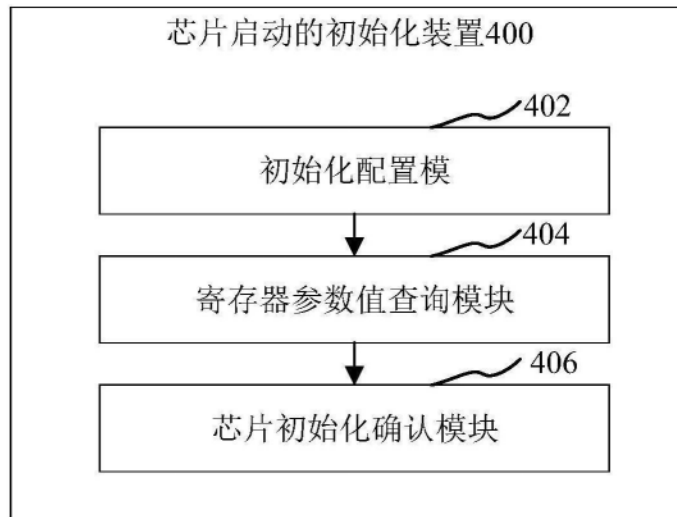


图4

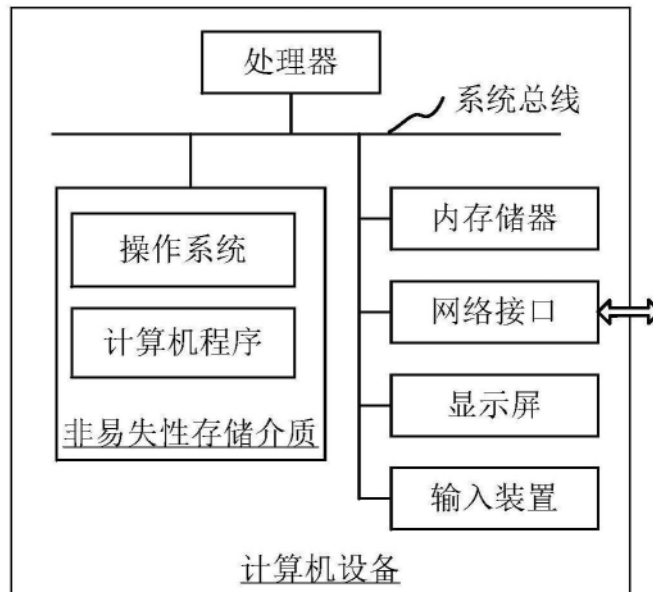


图5