



## (12)发明专利

(10)授权公告号 CN 106575228 B

(45)授权公告日 2019.11.08

(21)申请号 201580043094.3

(22)申请日 2015.08.03

(65)同一申请的已公布的文献号

申请公布号 CN 106575228 A

(43)申请公布日 2017.04.19

(30)优先权数据

14/465,371 2014.08.21 US

(85)PCT国际申请进入国家阶段日

2017.02.10

(86)PCT国际申请的申请数据

PCT/US2015/043466 2015.08.03

(87)PCT国际申请的公布数据

W02016/028482 EN 2016.02.25

(73)专利权人 高通股份有限公司

地址 美国加利福尼亚州

(72)发明人 克里斯托弗·保罗·弗拉斯卡蒂

穆拉特·巴尔契

阿温阿什·赛塔拉迈亚

莫里斯·富兰克林·里布尔

希滕德拉·莫汉·甘尼

(74)专利代理机构 北京律盟知识产权代理有限公司

责任公司 11287

代理人 宋献涛

(51)Int.Cl.

G06F 9/451(2018.01)

G06T 1/20(2006.01)

G06T 1/60(2006.01)

G06T 15/00(2011.01)

(56)对比文件

US 2014/0184623 A1, 2014.07.03,

US 2013/0127891 A1, 2013.05.23,

CN 103534680 A, 2014.01.22,

CN 103959338 A, 2014.07.30,

CN 103959337 A, 2014.07.30,

审查员 杨潇

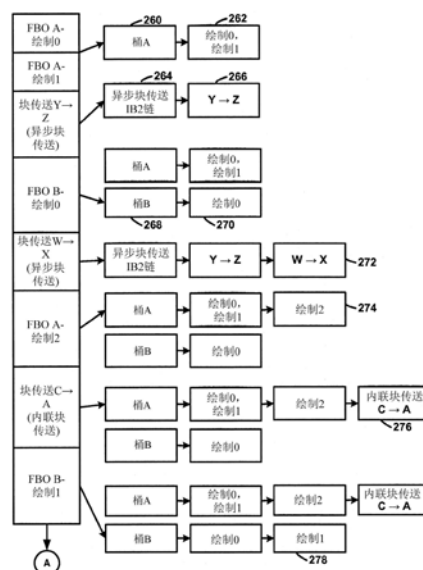
权利要求书4页 说明书17页 附图11页

(54)发明名称

图形处理中的渲染目标命令重新排序

(57)摘要

在实例中,一种用于渲染图形数据的方法包含接收与多个渲染目标相关联的多个命令,其中所述多个命令是以初始次序接收。所述方法还包含确定所述多个命令的执行次序,包含基于命令之间的数据相依性,以与所述初始次序不同的次序对所述多个命令中的一或多个者进行重新排序。所述方法还包含以所述所确定的执行次序来执行所述多个命令。



1. 一种用于渲染图形数据的方法,所述方法包括:

接收与用于渲染图像的多个渲染目标相关联的多个命令,其中所述多个命令中的命令是以初始次序接收,其中所述多个渲染目标中的每个渲染目标是相应的缓冲器,并且其中所述图像由来自所述多个渲染目标的内容组成;

确定所述命令的执行次序,所述执行次序是不同于所述初始次序的次序,其中确定所述执行次序包括基于所述初始次序的所述命令的渲染目标之间的数据相依性,产生所述多个渲染目标中的每一渲染目标的相应每渲染目标命令列表;

以所确定的执行次序将所述每渲染目标命令列表添加到命令缓冲器中;以及

以所述所确定的执行次序来执行所述命令缓冲器中的所述多个命令。

2. 根据权利要求1所述的方法,其中产生每一渲染目标的所述每渲染目标命令列表包括,针对所述多个命令中的每一相应命令:

当所述相应命令并不具有数据相依性时,基于与所述相应命令相关联的渲染目标将所述相应命令添加到每渲染目标命令列表;以及

其中执行所述多个命令包括执行相应每渲染目标命令列表的多个命令。

3. 根据权利要求1所述的方法,其中产生每一渲染目标的所述每渲染目标命令列表包括,针对所述多个命令中的每一相应命令:

当所述相应命令取决于先前命令时,确定所述执行次序,而不对在所述初始次序中在所述相应命令之后的命令进行重新排序。

4. 根据权利要求1所述的方法,其中确定所述执行次序包括,针对所述多个命令中的每一相应命令:

确定所述相应命令是否具有与所述相应命令的相同渲染目标的所述初始次序中的先前命令不同的相关联渲染模式;以及

当所述相应命令具有不同的相关联渲染模式时,确定所述执行次序而不对在所述初始次序中在所述相应命令之后的所述多个命令进行重新排序。

5. 根据权利要求1所述的方法,其进一步包括将渲染模式指派给所述每渲染目标命令列表中的每一者。

6. 根据权利要求1所述的方法,其进一步包括:基于与所述多个命令相关联的一或多个时戳来确定所述渲染目标之间的所述数据相依性。

7. 根据权利要求1所述的方法,

其中接收所述多个命令包括接收第一渲染目标的第一命令,接收第二渲染目标的第二命令,以及接收所述第一渲染目标的第三命令;

其中基于所述数据相依性来产生相应的每渲染目标命令列表包括确定所述第一渲染目标的所述第三命令是否取决于所述第二渲染目标的所述第二命令的结果;以及

当所述第三命令并不取决于所述第二命令时,确定所述执行次序包括对所述第三命令和所述第二命令进行重新排序,使得执行所述命令包括在所述第二命令之前执行所述第三命令。

8. 根据权利要求1所述的方法,其中产生相应命令列表包括基于逐命令地产生所述相应每渲染目标命令列表。

9. 根据权利要求8所述的方法,其中产生相应命令列表包括以初始次序将相应渲染目

标的相应命令添加到所述相应每渲染目标命令列表。

10. 根据权利要求1所述的方法,其中接收所述多个命令包括接收多个应用程序编程接口API命令。

11. 根据权利要求1所述的方法,其中所述多个命令包括在命令流中,并且其中产生相应的命令列表包括产生相应命令列表而不分析整个命令流。

12. 一种用于渲染图形数据的装置,所述装置包括:

存储器,其经配置以存储与用于渲染图像的多个渲染目标相关联的多个命令,其中所述图像由来自所述多个渲染目标的内容组成;

多个缓冲器,其中所述多个渲染目标中的每个渲染目标是所述多个缓冲器的相应的缓冲器;

命令缓冲器;以及

一或多个处理器,其经配置以:

接收与所述多个渲染目标相关联的所述多个命令,其中所述多个命令中的命令是以初始次序接收;

确定所述命令的执行次序,所述执行次序是不同于所述初始次序的次序,其中确定所述执行次序包括基于所述初始次序的所述命令的渲染目标之间的数据相依性,产生所述多个渲染目标中的每一渲染目标的相应每渲染目标命令列表;

将所述每渲染目标命令列表添加到所述命令缓冲器中;以及

以所确定的执行次序来执行所述命令缓冲器中的所述多个命令。

13. 根据权利要求12所述的装置,其中为了产生每一渲染目标的所述每渲染目标命令列表,所述一或多个处理器经配置以针对所述多个命令中的每一相应命令:

当所述相应命令并不具有数据相依性时,基于与所述相应命令相关联的渲染目标将所述相应命令添加到每渲染目标命令列表;及

其中为了执行所述多个命令,所述一或多个处理器经配置以执行相应每渲染目标命令列表的多个命令。

14. 根据权利要求12所述的装置,其中为了产生每一渲染目标的所述每渲染目标命令列表,所述一或多个处理器经配置以针对所述多个命令中的每一相应命令:

当所述相应命令取决于先前命令时,确定所述执行次序,而不对在所述初始次序中在所述相应命令之后的命令进行重新排序。

15. 根据权利要求12所述的装置,其中为了确定所述执行次序,所述一或多个处理器经配置以针对所述多个命令中的每一相应命令:

确定所述相应命令是否具有与所述相应命令的相同渲染目标的所述初始次序中的先前命令不同的相关联渲染模式;以及

当所述相应命令具有不同的相关联渲染模式时,确定所述执行次序而不对在所述初始次序中在所述相应命令之后的所述多个命令进行重新排序。

16. 根据权利要求12所述的装置,其中所述一或多个处理器进一步经配置以将渲染模式指派给所述每渲染目标命令列表中的每一者。

17. 根据权利要求12所述的装置,其中所述一或多个处理器进一步经配置以基于与所述多个命令相关联的一或多个时戳来确定所述渲染目标之间的所述数据相依性。

18. 根据权利要求12所述的装置，

其中为了接收所述多个命令，所述一或多个处理器经配置以接收第一渲染目标的第一命令，接收第二渲染目标的第二命令，且接收所述第一渲染目标的第三命令；

其中为了基于所述数据相依性来产生相应的每渲染目标命令列表，所述一或多个处理器经配置以确定所述第一渲染目标的所述第三命令是否取决于所述第二渲染目标的所述第二命令的结果；以及

当所述第三命令并不取决于所述第二命令时，为了确定所述执行次序，所述一或多个处理器经配置以对所述第三命令和所述第二命令进行重新排序，使得为了执行所述命令，所述一或多个处理器经配置以在所述第二命令之前执行所述第三命令。

19. 根据权利要求12所述的装置，其中所述一或多个处理器包含于图形处理单元GPU中。

20. 一种用于渲染图形数据的装置，所述装置包括：

用于接收与用于渲染图像的多个渲染目标相关联的多个命令的装置，其中所述多个命令中的命令是以初始次序接收，其中所述多个渲染目标中的每个渲染目标是相应的缓冲器，并且其中所述图像由来自所述多个渲染目标的内容组成；

用于确定所述命令的执行次序的装置，所述执行次序是与所述初始次序不同的次序，其中所述用于确定所述执行次序的装置包括用于基于所述初始次序的所述命令的渲染目标之间的数据相依性产生所述多个渲染目标中的每一渲染目标的相应每渲染目标命令列表的装置；

用于将所述每渲染目标命令列表添加到命令缓冲器中的装置；以及

用于以所确定的执行次序来执行所述命令缓冲器中的所述多个命令的装置。

21. 根据权利要求20所述的装置，其中所述用于产生每一渲染目标的所述每渲染目标命令列表的装置包括，对于所述多个命令中的每一相应命令：

用于当所述相应命令并不具有数据相依性时基于与所述相应命令相关联的渲染目标将所述相应命令添加到每渲染目标命令列表的装置；及

其中用于执行所述多个命令的装置包括用于执行相应每渲染目标命令列表的多个命令的装置。

22. 根据权利要求20所述的装置，其中所述用于产生每一渲染目标的所述每渲染目标命令列表的装置包括，对于所述多个命令中的每一相应命令：

用于当所述相应命令取决于先前命令时确定所述执行次序而不对在所述初始次序中在所述相应命令之后的命令进行重新排序的装置。

23. 根据权利要求20所述的装置，其中所述用于确定所述执行次序的装置包括对于所述多个命令中的每一相应命令：

用于确定所述相应命令是否具有与所述相应命令的相同渲染目标的所述初始次序中的先前命令不同的相关联渲染模式的装置；以及

当所述相应命令具有不同的相关联渲染模式时，用于确定所述执行次序而不对在所述初始次序中在所述相应命令之后的所述多个命令进行重新排序的装置。

24. 根据权利要求20所述的装置，其进一步包括用于将渲染模式指派给所述每渲染目标命令列表中的每一者的装置。

25. 根据权利要求20所述的装置,其进一步包括用于基于与所述多个命令相关联的一或多个时戳来确定所述渲染目标之间的所述数据相依性的装置。

26. 一种上面存储有指令的非暂时性计算机可读媒体,所述指令在被执行时致使一或多个处理器:

接收与用于渲染图像的多个渲染目标相关联的多个命令,其中所述多个命令中的命令是以初始次序接收,其中所述多个渲染目标中的每个渲染目标是相应的缓冲器,并且其中所述图像由来自所述多个渲染目标的内容组成;

确定所述命令的执行次序,所述执行次序是与所述初始次序不同的次序,其中确定所述执行次序包括基于所述初始次序的所述命令的渲染目标之间的数据相依性产生所述多个渲染目标中的每一渲染目标的相应每渲染目标命令列表;

将所述每渲染目标命令列表添加到命令缓冲器中;以及

以所确定的执行次序来执行所述命令缓冲器中的所述多个命令。

27. 根据权利要求26所述的非暂时性计算机可读媒体,其中为了产生每一渲染目标的所述每渲染目标命令列表,所述指令致使所述一或多个处理器针对所述多个命令中的每一相应命令:

当所述相应命令并不具有数据相依性时,基于与所述相应命令相关联的渲染目标将所述相应命令添加到每渲染目标命令列表;以及

其中为了执行所述多个命令,所述指令使得所述一或多个处理器执行相应每渲染目标命令列表的多个命令。

28. 根据权利要求26所述的非暂时性计算机可读媒体,其中为了产生每一渲染目标的所述每渲染目标命令列表,所述指令致使所述一或多个处理器针对所述多个命令中的每一相应命令:

当所述相应命令取决于先前命令时,确定所述执行次序,而不对在所述初始次序中在所述相应命令之后的命令进行重新排序。

29. 根据权利要求26所述的非暂时性计算机可读媒体,其中为了确定所述执行次序,所述指令致使所述一或多个处理器针对所述多个命令中的每一相应命令:

确定所述相应命令是否具有与所述相应命令的相同渲染目标的所述初始次序中的先前命令不同的相关联渲染模式;以及

当所述相应命令具有不同的相关联渲染模式时,确定所述执行次序而不对在所述初始次序中在所述相应命令之后的所述多个命令进行重新排序。

30. 根据权利要求26所述的非暂时性计算机可读媒体,其中所述指令进一步致使所述一或多个处理器将渲染模式指派给所述每渲染目标命令列表中的每一者。

## 图形处理中的渲染目标命令重新排序

### 技术领域

[0001] 本发明涉及渲染图形处理。

### 背景技术

[0002] 提供用于在电子显示器上视觉呈现的内容的装置通常包含图形处理单元 (GPU)。GPU 将表示内容的像素渲染在显示器上。GPU 产生显示器上的每一像素的一或多个像素值，且对显示器上的每一像素的像素值执行图形处理以渲染每一像素以供呈现。

### 发明内容

[0003] 本发明的技术通常涉及渲染图形数据。图形处理单元 (GPU) 可在渲染期间改变渲染目标。改变渲染目标在计算上相对密集和/或时间密集，归因于与渲染目标改变相关联的加载操作 (例如将数据加载到 GPU 存储器) 和存储操作 (例如将来自 GPU 存储器的数据存储在外部存储器)。本发明的技术包含以可减少加载和存储操作的方式，基于命令之间的相依性来对与渲染目标相关联的命令重新排序。

[0004] 在实例中，一种用于渲染图形数据的方法包含：接收与多个渲染目标相关联的多个命令，其中所述多个命令是以初始次序接收；确定所述多个命令的执行次序，包含基于命令之间的数据相依性，以与所述初始次序不同的次序，对所述多个命令中的一或多个者重新排序；以及以所述所确定的执行次序来执行所述多个命令。

[0005] 在另一实例中，一种用于渲染图形数据的装置包含存储器，其经配置以存储与多个渲染目标相关联的多个命令。所述装置还包含一或多个处理器，其经配置以：接收与多个渲染目标相关联的多个命令，其中所述多个命令是以初始次序接收；确定所述多个命令的执行次序，包含基于命令之间的数据相依性，以与所述初始次序不同的次序，对所述多个命令中的一或多个者重新排序；以及以所述所确定的执行次序来执行所述多个命令。

[0006] 在另一实例中，一种用于渲染图形数据的装置包含：用于接收与多个渲染目标相关联的多个命令的装置，其中所述多个命令是以初始次序接收；用于确定所述多个命令的执行次序的装置，包含基于命令之间的数据相依性，以与所述初始次序不同的次序，对所述多个命令中的一或多个者重新排序；以及用于以所述所确定的执行次序来执行所述多个命令的装置。

[0007] 在另一实例中，一种非暂时性计算机可读媒体上面存储有指令，所述指令在被执行时，致使一或多个处理器：接收与多个渲染目标相关联的多个命令，其中所述多个命令是以初始次序接收；基于命令之间的数据相依性来确定所述多个命令的执行次序，包含以与所述初始次序不同的次序对所述多个命令中的一或多个者进行重新排序；且以所述所确定的执行次序来执行所述多个命令。

[0008] 在附图和下文描述中陈述本发明的一或多个实例的细节。其它特征、目标和优点将从所述描述和图式以及所附权利要求书而显而易见。

## 附图说明

- [0009] 图1是说明可经配置以实施本发明的方面的计算装置的框图。
- [0010] 图2是更详细地说明来自图1的计算装置的框图。
- [0011] 图3A到3D是说明符合本发明的技术的实例命令重新排序过程的流程图。
- [0012] 图4是说明符合本发明的技术的另一实例命令重新排序过程的流程图。
- [0013] 图5A到5B是说明符合本发明的技术的另一实例命令重新排序过程的流程图。
- [0014] 图6是说明符合本发明的技术的另一实例命令重新排序过程的流程图。
- [0015] 图7是说明根据本发明的方面的用于对命令进行重新排序的实例过程的流程图。

## 具体实施方式

[0016] 传统图形处理单元 (GPU) 架构在渲染图形数据帧 (其可被称为图像) 时可能需要将相对较大的数据量从系统存储器读取以及写入到系统存储器。移动架构 (即, 移动装置上的 GPU) 可能缺乏处理整个数据帧所需的存储器带宽容量。因此, 已经开发了基于平铺块的架构, 其将图像划分成多个平铺块。设定平铺块的大小, 使得可使用相对较少量 (例如 256kB) 的高带宽芯片上图形存储器 (有时被称作图形存储器或 GMEM) 来处理它们。就是说, 每一平铺块的大小可取决于可用的芯片上图形存储器的量。接着在个别地处理每一平铺块之后重构图像。

[0017] 基于平铺块的渲染可相对于数个处理遍次进行描述。举例来说, 当执行基于平铺块的渲染时, GPU 可执行分格遍次和渲染遍次。相对于所述分格遍次, GPU 可处理整个图像, 并将光栅化的图元 (例如三角形) 分类成平铺块大小的区域 (被称作格) 中。举例来说, GPU 处理用于整个图像的命令流, 并且将图像的光栅化图元指派给所述格。

[0018] 在一些实例中, 在分格遍次期间, GPU 产生一或多个可见性流。可见性流指示在最终图像中可见的图元以及在最终图像中不可见的图元。举例来说, 如果图元被一或多个其它图元遮蔽, 使得图元在加阴影的成品图像中不可看见, 那么所述图元可为不可见。

[0019] 可见性流可为全部图像产生, 或可在每个区间基础上产生 (例如, 对于每个区间一个可见性流)。一般来说, 可见性流可包含一系列 1 和 0, 其中每个 “1” 或 “0” 与特定图元相关联。每个 “1” 指示图元在最终图像中可见。每个 “0” 指示图元在最终图像中不可见。可见性流可控制渲染遍次 (下文描述)。举例来说, 所述可见性流可用于跳过不可见图元的渲染。因此, 仅渲染和加阴影实际上促成格 (即, 在最终图像中可见) 的图元, 从而减少渲染和加阴影操作。

[0020] 在其它实例中, GPU 可使用不同过程 (例如, 不为上述可见性流或除了上述可见性流之外) 来将图元分类为位于特定格中。在另一实例中, GPU 可输出仅表示存在于给定格中的图元的 “指数” 的每格单独列表。举例来说, GPU 可最初将所有图元 (即, 顶点) 包含在一个数据结构中。GPU 可产生进入对于仅指向在每一格中可见的图元的每一格的结构的一组指针。因此, 仅针对可见索引的指针包含于中每格索引列表中。此类指针可起到与上述可见性流类似的目的, 其中指针指示包括哪些图元 (以及与图元相关联的像素) 以及哪些图元在特定格中可见。

[0021] 在任何情况下, 每一渲染遍次可包含清晰/未解析阶段、渲染阶段和解析阶段。在清晰/未解析阶段期间, GPU 可初始化芯片上存储器以用于待渲染的新平铺块。举例来说,

GPU可将芯片上存储器初始化到某一值(清晰),或将来自外部存储器的读取值初始化到芯片上存储器(未解析)。在显现阶段期间,GPU可处理平铺块,并且将经处理的平铺块存储到芯片上存储器。就是说,GPU可实施图形处理管线以确定像素值,并且将像素值写入到芯片上存储器。在解析阶段期间,GPU可将平铺块的最终像素值从芯片上存储器传送到外部存储器(可被称为帧缓冲器)。在修整图像的所有平铺块之后,例如将图像的所有平铺块存储到帧缓冲器,图像准备好输出(例如显示)。

[0022] GPU可使用一或多个渲染目标来渲染图形数据。一般来说,渲染目标是GPU在其中绘制正渲染的图像的像素的缓冲器。创建渲染目标可涉及保留存储器中的特定区用于绘制。在一些情况下,图像可由来自多个渲染目标的内容组成。举例来说,GPU可将内容渲染到若干渲染目标(例如离屏渲染),并组合所述内容以产生最终图像(还被称作场景)。

[0023] 渲染目标可与若干命令相关联。举例来说,渲染目标通常具有宽度和高度。渲染目标也可具有表面格式,其描述将多少位分配给每一像素,以及它们在红色、绿色、蓝色和 $\alpha$ (或另一色彩格式)之间如何划分。渲染目标的内容可由一或多个渲染命令(例如与片段着色器相关联的命令)修改。渲染目标还可结合深度模板缓冲器而作用。

[0024] 程序开发者可使用应用程序编程接口(API)(例如微软公司开发的DirectX)来定义渲染目标和相关联的命令。在一些实例中,渲染目标可类似于帧缓冲器对象(FBO),如开放式图形库(“OpenGL”)API中所定义。虽然通常相对于渲染目标来描述本发明的技术,但本文所述的技术可应用于FBO或任何其它类似建构。

[0025] 在一些情况下,GPU可在渲染期间改变渲染目标。举例来说,命令的初始执行次序(例如如由开发者确定)可致使GPU在图像的渲染期间,在渲染目标之间切换。改变渲染目标可在计算上和/或时间上相对密集。举例来说,如上所述,每一渲染遍次具有三个相关联阶段,包含清晰/未解析阶段、渲染阶段和解析阶段。当改变渲染目标时,GPU对图像的所有平铺块执行所有三个阶段。举例来说,GPU处理所有平铺块,并在移动到下一渲染目标之前,将所有相关联数据冲洗到用于当前渲染目标的存储器。

[0026] 在出于说明的目的的实例中,GPU可接收具有若干已定义渲染目标和相关联命令的命令流。GPU可执行与第一渲染目标相关联的渲染命令。GPU接着可切换到第二渲染目标,并执行与第二渲染目标相关联的渲染命令。GPU接着可切换回到第一渲染目标,并执行与所述第一渲染目标相关联的另一渲染命令。在此实例中,在切换到第二渲染目标之前,GPU可将与第一渲染目标相关联的数据冲洗到外部存储器。另外,当切换回到第一渲染目标(例如在第二渲染目标之后),GPU可将与第一渲染目标相关联的数据加载回到GPU存储器,以确保维持第一渲染目标的适当状态。数据的此冲洗和所述数据的重新加载可在渲染过程中呈现瓶颈。

[0027] 本发明的技术包含以可减少上文所述的低效率的方式,基于命令之间的相依性来对与渲染目标相关联的命令重新排序。举例来说,根据本发明的方面,GPU可在渲染数据之前确定与渲染目标相关联的命令的执行次序,以减少与渲染所述数据相关联的加载操作(例如将数据从外部存储器加载到GPU存储器)和存储操作(例如将数据从GPU存储器存储到外部存储器)的数目。GPU可以与其中接收和/或定义(例如经由API)命令的初始次序不同的次序来对所述执行次序的命令重新排序。

[0028] 在出于说明的目的的实例中,GPU(和/或GPU驱动程序)可在渲染之前产生每渲染



目标命令列表。举例来说, GPU可将命令识别为与特定渲染目标相关联, 并将所述命令添加到用于所述特定渲染目标的命令列表。GPU可继续将命令添加到命令列表中的每一者, 直到识别到命令之间的数据相依性(例如识别到先前经处理命令所取决于的命令)为止。在识别到数据相依性后, GPU可即刻执行包含于命令列表中的命令。GPU接着可重新开始组合每渲染目标命令列表的过程。

[0029] 本发明的技术可允许GPU(和/或GPU驱动程序)对命令重新排序, 以消除加载和存储操作。举例来说, 相对于上文所提供的实例, 在渲染之前, GPU可产生第一渲染目标的包含第一命令和第三命令的第一命令列表, 以及第二渲染目标的包含第二命令的第二命令列表。在渲染期间, 在切换到第二渲染目标之前, GPU可执行与第一渲染目标相关联的第一命令和第三命令两者。以此方式, 在执行第二命令之后, GPU并不需要切换回到第一渲染目标。

[0030] 如下文更详细地描述, 执行次序可由GPU、GPU驱动程序(例如如由中央处理单元(CPU)执行)或其一组合确定。作为一个实例, GPU驱动程序可产生命令流以及基于所述命令流的每渲染目标命令列表。在另一实例中, GPU可从GPU驱动程序接收命令流, 并基于所述命令流产生每渲染目标命令列表。在再一实例中, GPU驱动程序和GPU可共享确定命令的执行次序的职责。

[0031] 图1是说明可实施本发明的用于渲染图形数据的技术的计算装置30的框图。计算装置30的实例包含(但不限于)无线装置、移动或蜂窝电话(包含所谓的智能电话)、个人数字助理(PDA)、视频游戏控制台(包含视频显示器、移动视频游戏装置、移动视频会议单元)、膝上型计算机、桌上型计算机、电视机顶盒、平板计算装置、电子书阅读器、固定或移动媒体播放器, 等。

[0032] 在图1的实例中, 计算装置30包含具有CPU存储器34的中央处理单元(CPU)32、具有GPU存储器38和一或多个着色单元40的图形处理单元(GPU)36、显示单元42、存储经渲染数据45(“ren. 数据”)的显示缓冲器单元44、用户接口单元46以及数据存储单元48。另外, 存储单元48可存储具有编译器54的GPU驱动程序50、GPU程序52和本地编译的GPU程序56。

[0033] CPU 32的实例包含(但不限于)数字信号处理器(DSP)、通用微处理器、专用集成电路(ASIC)、现场可编程逻辑阵列(FPGA)或其它等效集成或离散逻辑电路。尽管在图1的实例中将CPU 32和GPU 36说明为单独的单元, 但在一些实例中, CPU 32和GPU 36可集成到单个单元中。CPU 32可执行一或多个应用程序。应用程序的实例可包含网络浏览器、电子邮件应用程序、电子表格、视频游戏、音频和/或视频捕获、重放或编辑应用程序, 或起始待经由显示单元42呈现的图形数据的产生的其它应用程序。

[0034] 在图1中展示的实例中, CPU 32包含CPU存储器34。CPU存储器34可表示在执行机器或目标代码中使用的芯片上存储装置或存储器。CPU存储器34可各自包括能够存储固定数目个数字位的硬件存储器寄存器。与从存储单元48读取值或将值写入到存储单元48相比, CPU 32可能更迅速地本地CPU存储器34读取值或将值写入到本地CPU存储器34, 存储单元48可(例如)经由系统总线存取。

[0035] GPU 36表示用于执行图形操作的一或多个专用处理器。也就是说, 例如, GPU 36可为专用硬件单元, 其具有用于渲染图形且执行GPU应用程序的固定功能和可编程组件。GPU 36还可包含DSP、通用微处理器、ASIC、FPGA, 或其它等效集成或离散逻辑电路。

[0036] GPU 36还包含GPU存储器38, 其可表示在执行机器或目标代码中使用的芯片上存

储装置或存储器。GPU存储器38可各自包括能够存储固定数目个数字位的硬件存储器寄存器。与从存储单元48读取值或将值写入到存储单元48相比, GPU 36可能更迅速地从本地GPU存储器38读取值或将值写入到本地GPU存储器38, 存储单元48可(例如)经由系统总线存取。

[0037] 显示单元42表示能够显示视频数据、图像、文本或任何其它类型的数据以供检视者消耗的单元。显示单元42可包含液晶显示器(LCD)、发光二极管(LED)显示器、有机LED(OLED)、有源矩阵OLED(AMOLED)显示器,或其类似者。

[0038] 显示缓冲器单元44表示专用于存储数据以供呈现显示单元42的例如计算机产生的图形、静态图像、视频帧等图像(经渲染数据45)的存储器或存储装置。显示缓冲器单元44可表示包含多个存储位置的二维缓冲器。显示缓冲器单元44内的存储位置的数目可实质上类似于待显示在显示单元42上的像素的数目。举例来说,如果显示单元42经配置以包含640x480个像素,那么显示缓冲器单元44可包含存储像素颜色和强度信息(例如,红色、绿色和蓝色像素值或其它色彩值)的640x480个存储位置。

[0039] 显示缓冲器单元44可存储由GPU 36处理的像素中的每一者的最终像素值。显示单元42可从显示缓冲器单元44检索最终像素值,且基于存储在显示缓冲器单元44中的像素值显示最终图像。

[0040] 用户接口单元46表示用户可与之交互或以其它方式介接以与计算装置30的其它单元(例如,CPU 32)通信的单元。用户接口单元46的实例包含(但不限于)轨迹球、鼠标、键盘以及其它类型的输入装置。用户接口单元46也可为或包含触摸屏,且触摸屏可并入为显示单元42的一部分。

[0041] 存储单元48可包括一或多个计算机可读存储媒体。存储单元48的实例包含(但不限于):随机存取存储器(RAM)、只读存储器(ROM)、电可擦除可编程只读存储器(EEPROM)、CD-ROM或其它光盘存储装置、磁盘存储装置或其它磁性存储装置、快闪存储器,或可用以存储呈指令或数据结构形式的所要程序代码且可通过计算机或处理器存取的任何其它媒体。

[0042] 在一些实例实施方案中,存储单元48可包含致使CPU 32和/或GPU 36执行本发明中归于CPU 32和GPU 36的功能的指令。在一些实例中,存储单元48可被视为非暂时性存储媒体。术语“非暂时性”可指示存储媒体未包含于载波或所传播信号中。然而,术语“非暂时性”应被解释为意味着存储单元48是不可移动的。作为一个实例,可将存储单元48从计算装置30中去除,且将其移动到另一装置。作为另一实例,可将大体类似于存储单元48的存储单元插入到计算装置30中。在某些实例中,非暂时性存储媒体可存储可随时间而改变的数据(例如,在RAM中)。

[0043] 存储单元48存储GPU驱动程序50和编译器54、GPU程序52以及本地编译的GPU程序56。GPU驱动程序50表示提供接口以存取GPU 36的计算机程序或可执行码。CPU32执行GPU驱动程序50或其部分以与GPU 36介接,且出于此原因,在图1的实例中将GPU驱动程序50示出为标记CPU 32内的“GPU驱动程序50”的短划线框。由CPU32执行的程序或其它可执行码(包含GPU程序52)可存取GPU驱动程序50。

[0044] GPU程序52可包含以高级(HL)编程语言撰写(例如,使用应用程序编程接口(API))的代码。API的实例包含开放计算语言(“OpenCL”)、开放图形库(“OpenGL”)和DirectX,如由微软(Microsoft)公司开发。一般来说,API包含由相关联硬件执行的一组预定的标准化命

令。API命令允许用户指令GPU的硬件组件执行命令，而无需用户了解所述硬件组件的详情。

[0045] GPU程序52可调用或以其它方式包含GPU驱动程序50所提供的一或多个功能。CPU 32通常执行其中嵌入GPU程序52的程序，且在遇到GPU程序52后，即刻将GPU程序52传递到GPU驱动程序50。在此上下文中，CPU 32执行GPU驱动程序50以处理GPU程序52。也就是说，例如，GPU驱动程序50可通过将GPU程序52编译成可通过GPU 36执行的目标或机器代码来处理GPU程序52。在图1的实例中将此目标代码示出为本地编译的GPU程序56。

[0046] 在一些实例中，编译器54可在执行其中嵌入有GPU程序52的程序期间，实时地或接近实时地操作以编译GPU程序52。举例来说，编译器54通常表示将根据HL编程语言定义的HL指令还原为LL编程语言的低级(LL)指令的单元。在编译之后，这些LL指令能够通过特定类型的处理器或其它类型的硬件来执行，其它类型的硬件例如FPGA、ASIC及其类似者(包含(例如)CPU 32和GPU 36)。

[0047] 在图1的实例中，当执行包含GPU程序52的HL代码时，编译器54可从CPU 32接收GPU程序52。编译器54可编译GPU程序52以产生符合LL编程语言的本地编译的GPU程序56。编译器54接着输出包含LL指令的本地编译的GPU程序56。

[0048] GPU 36通常接收经本地编译的GPU程序56(如由GPU 36内的标记为“经本地编译的GPU程序56”的短划线框所示)，于是在一些情况下，GPU 36渲染一或多个图像，且将经渲染的图像输出到显示缓冲器单元44。举例来说，GPU 36可产生待显示在显示单元42处的若干图元。图元可包含以下各项中的一或多者：线(包含曲线、样条线等)、点、圆、椭圆、多边形(其中通常多边形被定义为一或多个图元的集合)或任何其它二维(2D)图元。术语“图元”还可指三维(3D)图元，例如立方体、圆柱体、球体、锥体、角锥、环面或其类似者。一般来说，术语“图元”是指能够通过GPU 36渲染以用于经由显示单元42显示为图像(或在视频数据的上下文中为帧)的任何基本几何形状或元素。

[0049] GPU 36可通过应用一或多个模型变换(其也可在状态数据中指定)将图元和图元的其它属性(例如，其定义色彩、纹理、照明、相机配置或其它方面)变换到所谓的“世界空间”中。一旦进行了变换，GPU 36就可对有源相机应用视图变换(并且，其还可在定义相机的状态数据中加以指定)以将图元和光的坐标变换到相机或眼睛空间中。GPU 36还可执行顶点着色以鉴于任何作用中光来渲染图元的外观。GPU 36可在上述模型、世界空间或检视空间中一或多者中执行顶点着色(但其通常在世界空间中执行)。

[0050] 一旦图元经着色，GPU 36就可执行投影以将图像投影到具有极点的单元立方体中，作为一个实例，在(-1,-1,-1)和(1,1,1)处。此单位立方体通常被称作正规化检视体积。在将模型从眼睛空间变换到正规化检视体积之后，GPU 36可执行剪裁以去除至少部分地不会驻留在检视体积内的任何图元。换句话说，GPU 36可去除不在相机的帧内的任何图元。GPU 36可接着将图元的坐标从检视体积映射到屏幕空间，从而有效地将图元的3D坐标缩减到屏幕的2D坐标。

[0051] 在给出定义图元的经变换且经投影的顶点及其相关联的着色数据的情况下，GPU 36可接着将图元光栅化。在光栅化期间，GPU 36可应用与图元相关联的任何纹理(其中纹理可包括状态数据)。GPU 36还可在光栅化期间执行Z-缓冲器算法(还被称作深度测试)，以确定图元和/或物体中的任一者是否被任何其它物体遮挡。Z-缓冲算法根据图元的深度对图元进行排序，使得GPU 36知晓将每一图元绘制到屏幕的次序。在分格(例如，用于基于平铺

块的渲染)时,可不在光栅化期间执行着色。然而,在渲染图元时,GPU 36可计算且设定由图元覆盖的屏幕的像素的颜色。GPU 36接着将经渲染的像素输出到显示缓冲器单元44。

[0052] 显示缓冲器单元44可临时存储经渲染的图像的经渲染像素,直到整个图像被渲染为止。在此上下文中,显示缓冲器单元44可被视为图像帧缓冲器。显示缓冲器单元44可发射经渲染图像以显示在显示单元42上。虽然分开示出和描述,但在一些情况下,显示缓冲器单元44可形成存储单元48的一部分。

[0053] 在一些实例中,GPU 36可实施基于平铺块的渲染来渲染图像。举例来说,GPU 36可实施基于平铺块的架构,所述基于平铺块的架构通过将图像分裂成被称作平铺块的多个部分来渲染图像。平铺块可基于GPU存储器38的大小来设定大小。

[0054] 当实施基于平铺块的渲染时,GPU 36可执行一分格遍次以及一或多个渲染遍次。举例来说,相对于分格遍次,GPU 36可处理整个图像,并且将光栅化图元分类成初始分格配置的格(通过GPU驱动程序50设置)。GPU 36还可在分格遍次期间产生可见性流,其可根据格来分离。举例来说,可将每一格指派给所述图像的可见性流的对应部分。GPU驱动程序50可存取可见性流,并且产生用于渲染每一格的命令流。

[0055] 相对于每一渲染遍次,GPU 36可执行清晰/未解析阶段、渲染阶段和解析阶段。在清晰/未解析阶段期间,GPU 36初始化GPU存储器38以用于待渲染的新平铺块。在渲染阶段期间,GPU 36可渲染平铺块,并且将渲染的平铺块存储到GPU存储器38。就是说,GPU 36可执行像素着色和其它操作,以确定对于平铺块的每个像素的像素值,并且将像素值写入到GPU存储器38。在解析阶段期间,GPU 36可将平铺块的经修整的像素值从GPU存储器38传送到显示缓冲器单元44(或存储单元48)。在GPU 36已经以此方式渲染与帧相关联的所有平铺块之后,显示缓冲器单元44可将经修整的图像输出到显示单元42。

[0056] 在一些情况下,GPU 36可通过在渲染之后将像素值存储到显示缓冲器单元44(或存储单元48),而不是将像素值存储到GPU存储器38,来直接渲染数据。通过直接渲染,GPU驱动程序50并不使用可见性流来识别和跳过最终图像中不可见的图元。实际上,命令流包含渲染所有图元的指令,不管所述图元是否可见。因此,存储单元48和/或显示缓冲器单元44中的不可见图元可最终由与一或多个其它图元相关联的像素值盖写。

[0057] 在上文所述的渲染之前,GPU驱动程序50使用GPU程序52产生命令流。例如,命令流可含有用于渲染来自GPU程序52的图像的指令。GPU驱动程序50可将指令添加到命令流,这通过GPU 36以其中它们在流中出现的次序执行。命令流可定义组成来自GPU程序52的图像的图元。

[0058] 如上所述,命令流可包含定义图像的一或多个渲染目标的指令。创建渲染目标可保留存储器(例如显示缓冲器单元44或存储单元48)中的特定区以用于绘制。在一些情况下,图像可由来自多个渲染目标的内容组成。每一渲染目标可具有包含渲染命令的多个相关联命令。渲染目标还可结合深度模板缓冲器而作用。

[0059] 在一些情况下,GPU 36可在渲染期间改变渲染目标。举例来说,GPU程序52可包含在产生命令流时由GPU驱动程序50维持的初始执行次序的命令。用于不同渲染目标的命令可在命令流中交错,使得GPU 36在执行本地编译的GPU程序56时在渲染目标之间切换。

[0060] 如上所述,改变渲染目标可为计算上和/或时间上相对密集的,归因于与渲染目标相关联的加载和存储操作。举例来说,在执行与渲染目标相关联的命令之前,GPU 36可将来

自存储单元48或显示缓冲器单元44的必需数据加载到用于正渲染的每一平铺块的GPU存储器38。另外,在执行与特定渲染目标相关联的命令之后,且在切换到新渲染目标之前,GPU 36可将与特定渲染目标相关联的数据从GPU存储器38冲洗到用于正渲染的每一平铺块的存储单元48或显示缓冲器单元44。

[0061] 根据本发明的方面,GPU 36(和/或GPU驱动程序50)可接收与多个渲染目标相关联的命令,所述渲染目标包含具有多个第一命令的第一渲染目标,以及具有多个第二命令的第二渲染目标。GPU 36(和/或GPU驱动程序50)可基于所述命令之间的数据相依性来确定所述多个渲染目标的命令的执行次序,所述命令包含所述多个第一命令和所述多个第二命令。GPU 36(和/或GPU驱动程序50)接着可以所述所确定的执行次序来执行所述多个渲染目标的命令。在一些实例中,所述所确定的执行次序可减少与渲染所述数据相关联的加载操作和存储操作的数目。

[0062] 在一些情况下,当确定执行次序时,GPU 36(和/或GPU驱动程序50)可对命令进行重新排序,使得所述命令呈现初始命令流的初始次序不同的次序。在出于说明的目的的实例中,假定命令流包含用于第一渲染目标A和第二渲染目标B的指令。另外,假定命令流包含初始执行次序的命令,其包含与渲染目标A相关联的第一命令、与渲染目标B相关联的第一命令、与渲染目标A相关联的第二命令,以及与渲染目标B相关联的第二命令。

[0063] 在以上实例中,GPU 36(和/或GPU驱动程序50)可确定用于渲染目标A和渲染目标B的命令的执行次序,其可不同于命令流所指定的初始次序。举例来说,GPU 36(和/或GPU驱动程序50)可在渲染之前产生每渲染目标命令列表。在此实例中,GPU 36(和/或GPU驱动程序50)可开始用用于渲染目标A的第一命令来建立渲染目标命令列表A。GPU 36(和/或GPU驱动程序50)接着可识别渲染目标B,并开始用用于渲染目标B的第一命令来建立渲染目标命令列表B。GPU 36(和/或GPU驱动程序50)接着可确定用于渲染目标A的第二命令是否取决于用于渲染目标B的命令。如果不存在数据相依性,那么GPU 36(和/或GPU驱动程序50)可将用于渲染目标A的第二命令添加到渲染目标命令列表A。同样地,如果不存在数据相依性,那么GPU 36(和/或GPU驱动程序50)可将用于渲染目标B的第二命令添加到渲染目标命令列表B。

[0064] 因此,在此实例中,并非执行用于渲染目标A的命令,切换到渲染目标B,执行用于渲染目标B的命令,切换到渲染目标A,且执行用于渲染目标A的命令,切换到渲染目标B,以及执行用于渲染目标B的命令,GPU 36可根据所确定的执行次序来执行所述命令。举例来说,GPU 36可基于命令列表来执行所述命令,其允许GPU 36来执行与渲染目标命令列表A相关联的所有命令,接着是与渲染目标命令列表B相关联的所有命令。在这样做时,GPU 36仅进行渲染目标A与渲染目标B之间的单次转变,这可减少与切换渲染目标相关联的加载和存储操作的数目。

[0065] 应理解,仅作为实例而提供计算装置30,且可以不同方式来布置执行本发明的技术的其它计算装置30。举例来说,虽然将显示缓冲器单元44与存储单元48分离来展示和描述,但在其它实例中,显示单元缓冲器44和存储单元48可并入到同一组件中。

[0066] 此外,应理解,计算装置30可包含图1中为了清楚起见而未示出的额外模块或单元。举例来说,计算装置30可包含用于发射和接收数据的收发器单元,且可包含允许计算装置30与另一装置或网络之间的无线或有线通信的电路。计算装置30还可包含扬声器和麦克风(所述两者在图1中均未示出),以在其中计算装置30为移动无线电话(例如,智能电话)的

实例中实现电话通信,或可包含其中计算装置30为媒体播放器或平板计算机的扬声器和/或麦克风。在一些情况下,在其中计算装置30为桌上型计算机或经装备以与外部用户接口或显示器介接的其它装置的实例中,用户接口单元46和显示单元42可在计算装置30的外部。

[0067] 图2是更详细地说明计算设备30的部分的框图。在图2的实例中,GPU 36包含GPU存储器38、命令处理器60、一或多个处理单元64、光栅化器68和可见性单元72。另外,CPU 32包含CPU存储器34、GPU驱动程序50、编译器54和命令组合单元76。应理解,图2的某些单元可高度集成,但是出于概念目的分开示出。此外,一些单元可相对于出于概念目的单个单元描述,但是可包含一或多个功能单元。

[0068] 提供图2作为可利用本发明的用于渲染图形数据的技术的GPU的仅一个实例。在其它实例中,用于渲染图形数据的技术可由具有其它组件的多种其它GPU进行。举例来说,GPU 36还可包含与分析及渲染图像相关的多种其它组件和单元,例如输入组合单元、纹理单元、调度单元、算术逻辑单元(ALU)或其它固定功能或可编程GPU组件。

[0069] GPU 36的组件可与存取例如存储单元48(图1)等外部存储器相比相对较低的时延存取GPU存储器38。举例来说,GPU存储器38可为在GPU 36的芯片上且与GPU组件相对紧密近接的芯片上存储器,且可与GPU 36内的专用存储器总线相关联。为了存取存储在存储单元48中的数据,相对来说,GPU 36可必须与计算装置30的其它组件(例如CPU 32)共享存储器总线,这可导致更受限制的可用带宽。

[0070] 为了利用高带宽、低等待时间GPU存储器38,如上文所描述,GPU 36可使用基于平铺块的渲染架构来渲染图形。GPU 36可将图像(其还可被称作场景)分成若干较小部分(例如平铺块)。GPU存储器38可存储与平铺块相关联的数据,同时GPU 36渲染所述平铺块。在显现平铺块之后,GPU 36可经由存储器总线来解析来自GPU存储器38渲染的像素数据或将其复制到显示缓冲器单元44。

[0071] 命令处理器60可负责从GPU驱动程序50读取命令流。举例来说,如上文相对于图1所描述,GPU驱动程序50可发出用于通过GPU 36执行的指令,其可被称为命令流。命令处理器60可读取命令流的指令和/或对其进行解码。在一些实例中,命令处理器60可从含有命令流的指令的缓冲器读取。命令处理器60还可起始所述指令在GPU 36处的实行。举例来说,命令处理器60可将指令馈送到安排通过处理单元64执行指令的线程调度程序。

[0072] 处理单元64可包含一或多个处理单元,其中的每一者可为可编程处理单元或固定功能处理单元。在一些实例中,可编程着色器单元可包含经配置以并行操作的多个处理单元(例如,SIMD管线)。可编程着色器单元可具有存储着色器程序指令的程序存储器和执行状态寄存器,例如指示程序存储器中正执行的当前指令或待提取的下一指令的程序计数器寄存器。处理单元64中的可编程着色器单元可包含(例如)顶点着色器单元、像素着色器单元、几何形状着色器单元、壳着色器单元、域着色器单元、曲面细分控制着色器单元、曲面细分评估着色器单元、计算着色器单元和/或统一着色器单元。

[0073] 处理单元64可负责执行指令。举例来说,处理单元64可负责执行一或多个着色器程序。在一些实例中,着色器程序可为以高级着色语言(例如,OpenGL着色语言(GLSL)、高级着色语言(HLSL)、用于图形的C(Cg)着色语言等)编写的程序的经编译版本。在一些实例中,着色器程序可被称为着色器内核。一般来说,内核可包含定义将由GPU 36执行的任务或函

数的程序代码。

[0074] 因此,处理单元64可为负责顶点、几何结构和像素着色操作的可编程着色单元。举例来说,处理单元64中的一或多者可负责确定组成场景的图元(例如三角形)的顶点位置,以产生所述场景的三角形网。另外,处理单元64中的一或多者可负责从三角形网产生图元,以及像素填充和着色操作。

[0075] 处理单元64可相同地配置,或可个别地被配置成执行特定任务。举例来说,处理单元64中的一个可表示为负责分格操作的“分格着色器”,同时其余处理单元64可负责执行上述顶点、几何或像素着色操作。

[0076] 光栅化器68可包含经硬连线以执行某些功能的一或多个固定处理单元。尽管固定功能硬件可经由例如一或多个控制信号配置以执行不同功能,但所述固定功能硬件通常并不包含能够接收用户编译的程序的程序存储器。在一些实例中,光栅化器68可被配置成执行光栅操作,例如深度测试、剪刀(scissors)测试、阿尔法混合等。

[0077] 另外,光栅化器68可接收顶点信息并且可产生场景的图元的表示。在一些实例中,光栅化器68将预定义规则应用于接收到的顶点信息以确定哪些图元在最终场景中可见。光栅化器68可剔除或去除场景的任何不可见图元。举例来说,光栅化器68可执行z缓冲(其还可被称作深度测试)以识别由其它图元覆盖的图元,并且因此在最终场景中不可见。

[0078] 可见性单元72可包含固定功能硬件组件和/或可编程处理单元的任何组合。可见性单元72可从光栅化器68接收光栅化数据并且产生一或多个可见性流。为了产生可见性流,可见性单元72可将所述可见图元中的每一者(如由光栅化器68确定)分发给格。每一格可表示经修整场景的平铺块。

[0079] 在一些实例中,可见性单元72可针对所述格中的每一者产生单独的可见性流。举例来说,可见性单元72可针对特定格通过设定旗标产生可见性流以指示特定格的图元的哪些像素可见以及特定格的图元的哪些像素不可见。根据一些方面,可见性单元72可设定指示图元在最终场景中可见的旗标值“1”,以及指示图元在最终场景中不可见的旗标值“0”。在一些实例中,可见性单元72可根据图像的过程光栅化来操作。就是说,可见性单元72可确定关于较粗标度(例如,针对四个像素的块)的可见性信息,而不是指示每一像素的可见性状态。

[0080] 在其它实例中,可见性单元72可使用不同过程来将图元分类为位于特定格中。在另一实例中,可见性单元72可输出仅表示存在于给定格中的图元的“指数”的每格单独列表。举例来说,可见性单元72可最初在一个数据结构中包含所有图元(即顶点)。可见性单元72可产生进入对于仅指向在每一格中可见的图元的每一格的结构中的一组指针。因此,仅针对可见索引的指针包含于中每格索引列表中。

[0081] 根据本发明的方面,命令组合单元76可负责确定包含于命令流中的命令的执行次序。举例来说,命令组合单元76可经配置以接收具有与一或多个渲染目标相关联的多个命令的命令流。命令流可包含呈初始执行次序的多个命令。

[0082] 命令组合单元76可基于命令之间的数据相依性的存在或不存在来确定所述命令流的命令的执行次序。在一些实例中,命令组合单元76可使用若干渲染目标命令列表来确定执行次序,所述列表中的每一者包含与相应渲染目标相关联的命令列表。命令组合单元76可将命令添加到命令列表,直到识别到数据相依性和/或渲染模式(例如直接渲染模式、



分格渲染或类似者)的改变为止。

[0083] 根据本发明的方面,渲染目标命令列表可被称为桶(bucket)。举例来说,命令组合单元76可每渲染目标产生一桶。由于对所述配置的渲染,可将桶描述为单个渲染目标的渲染命令参考链(例如还被称作FBO配置)。所述链可包含命令混合,包含用于准备GPU36来进行渲染的前导命令、实际渲染命令(例如绘制)以及恢复状态命令(例如用于将GPU36恢复到特定的绘制状态)。一般来说,桶含有为特定渲染目标发出的所有渲染命令。桶也可以在本文中被称作“命令列表”。

[0084] 在渲染之前,命令组合单元76可产生多于一个桶。GPU 36接着可在此期间循序地渲染与每一桶相关联的命令。一般来说,在渲染桶N+1的命令之前,GPU 36可执行桶N的渲染命令。

[0085] 在一些实例中,如下文更详细地描述,GPU 36和命令组合单元76可使用第一命令缓冲器和第二命令缓冲器来控制渲染。举例来说,命令组合单元76可为第一缓冲器确定命令,其参考第二缓冲器。所述第一缓冲器通常不包含渲染命令,例如绘制命令或复制命令。

[0086] 第二命令缓冲器可包含对渲染命令(例如包含前导命令、绘制命令、复制命令、恢复状态命令或类似者)的参考。在此实例中,命令组合单元76可通过在第二命令缓冲器中组合(和重新排序)对渲染命令的参考来产生执行次序。在渲染后,GPU 36可即刻执行第一命令缓冲器,其包含对呈适当次序的第二命令缓冲器的命令的参考。

[0087] 虽然出于解释的目的与GPU驱动程序50分开示出,但应理解,命令组合单元76可与GPU驱动程序50集成,使得GPU驱动程序50进行本发明中归于命令组合单元76的技术。然而,虽然将上文的技术描述为由GPU驱动程序50的命令组合单元76执行,但应理解,命令组合单元76可与GPU 36集成,使得GPU 36负责确定渲染命令的执行次序。在此实例中,GPU 36可从GPU驱动程序50接收命令流,并在渲染所述命令流的命令之前,确定执行次序。

[0088] 图3A到3D是说明符合本发明的技术的实例命令组合过程的流程图。举例来说,图3A到3D的最左列说明与第一渲染目标(渲染目标A)和第二渲染目标(渲染目标B)相关联的若干渲染命令。图3A到3D的渲染命令标记为一到六(如由带圆圈编号指示)。在图3A到3D右侧的列说明以执行次序来组合与渲染目标A和渲染目标B相关联的命令。就是说,图3A到3D说明产生每渲染目标桶,其中每一桶含有对命令的参考的列表。

[0089] 举例来说,如上所述,在一些情况下,GPU(例如GPU 36)或GPU驱动程序(例如GPU驱动程序50)可对命令进行排序,以使用第一缓冲器(其可被称为IB1)和第二缓冲器(其可被称为IB2)来执行。IB1可被称为间接缓冲器1,而IB2可被称为间接缓冲器2。IB1和IB2可为层级式的。举例来说,IB1中的命令可调用IB2中的整个命令流。因此,IB1通常包含对IB2的参考,且可在渲染期间使用。IB2可包含对渲染命令(例如包含前导命令、绘制命令、复制命令、恢复状态命令或类似者)的参考。以此方式,GPU 36可在IB2中建立命令列表,且将所述所确定的命令列表冲洗到IB1以供执行。

[0090] 因此,在图3A到3D的实例中,由于对所述配置的渲染,桶可为单个渲染目标的IB2参考链(例如FBO配置)。所述链可包含来自前导IB2、渲染IB2和恢复状态IB2(例如前导IB2的子集)的命令混合,如下文更详细描述。因此,每一桶含有为特定渲染目标发出的所有渲染命令。如上所述,可存在等待调度到IB1的多个桶。一般来说,在渲染桶N+1的命令之前,GPU 36可执行桶N的渲染命令。



[0091] 随着GPU 36渲染到特定渲染目标,GPU 36可在渲染IB2中累加。当从一个渲染目标切换到另一渲染目标(或存在命令到IB1的冲洗)时,GPU 36可将累积的渲染保存到当前渲染目标的现有桶,或可产生新的桶条目。每一渲染目标可映射到渲染桶ID,其识别所述桶中的每一者。

[0092] 根据本发明的方面且如下文更详细地描述,GPU 36可基于渲染目标A和渲染目标B的命令之间的数据相依性来对渲染目标A和渲染目标B的命令从初始次序重新排序。举例来说,假定初始操作次序包含对渲染目标A的渲染、对渲染目标B的渲染、对渲染目标A的渲染以及对渲染目标B的渲染。根据本发明的方面,当渲染目标A与渲染目标B之间不存在数据相依性时,GPU 36从渲染目标B切换回到渲染目标A,GPU 36可继续将渲染目标A的命令累积在其中累积渲染目标A的先前命令的相同桶中。同样地,GPU 36可使用单个桶来累积渲染目标B的命令。GPU 36接着可冲洗所述桶,其中渲染目标B的命令在渲染目标A的命令之后。

[0093] 根据本发明的一些方面,GPU 36可至少部分地基于与渲染命令相关联的时戳来确定数据相依性。举例来说,GPU驱动器50可基于既定执行渲染命令的次序,将时戳发出到命令流的每一渲染命令。因此,GPU 36可通过识别与当前命令相关联的任何命令是否具有比当前命令的时戳早的时戳来确定当前命令的数据相依性。在一些实例中,GPU 36可另外或替代地基于GPU 36存取命令缓冲器的方式来确定数据相依性。举例来说,GPU 36可将命令缓冲器指定为源缓冲器(从中读取数据)或目的地缓冲器(向其写入数据)。如果GPU 36在执行先前命令缓冲器的指令时已将特定命令缓冲器称为来源,那么GPU 36可仅写入到后续命令缓冲器中的特定命令缓冲器(例如在读取之前,GPU 36可不对写入进行重新排序)。同样地,如果GPU 36已参考特定命令缓冲器,那么GPU 36可仅从后续命令缓冲器中的特定缓冲器读取(例如在写入之前,GPU 36可不对所述读取进行重新排序)。

[0094] 上文所述的重新排序可有助于减少与渲染目标A和渲染目标B相关联的加载和存储操作的数目,因为GPU 36无需进行渲染目标A与渲染目标B之间超过单个转变,如上所述。当GPU 36结束一批渲染(例如不存在额外命令要添加到桶)时,GPU 36可确定所述桶的渲染模式。实例渲染模式包含直接渲染模式、使用软件的分格渲染模式、使用GPU 36的硬件的分格渲染模式,或其它渲染模式。在将桶冲洗到IB1中时,GPU 36可使用所述所确定的渲染模式。在一些情况下,渲染模式可被认为是桶的一部分,在创建桶条目时确定,且在指定之后无法改变。

[0095] 作为分桶过程的实例,相对于图3A的步骤1,GPU 36可接收为渲染目标A设定色彩缓冲器(设定色彩缓冲器A)的指令(90),接着接收渲染目标A的第一绘制命令(绘制1-A)(92)。作为响应,GPU 36可开始通过产生渲染目标A的桶ID(获得A的渲染桶ID)(94)以及更新正渲染的图像的格布局(更新格布局)(96)来为渲染目标A建立桶。GPU 36接着可将前导命令添加到桶(产生前导)(98),其可指令GPU 36进入已知状态(例如包含GPU存储器38的状态)来进行绘制。GPU 36接着可添加渲染IB2命令(渲染IB2-0)(100)和绘制命令(绘制1-A)(102)。

[0096] 在步骤2,GPU 36可接收渲染目标A的第二绘制命令(绘制2-A)(104)。GPU 36可将先前在步骤1期间产生的第二绘制命令添加到IB2。就是说,GPU 36将第二命令(绘制2-A)添加到渲染目标A的IB2的结尾(渲染IB2-0(106)、绘制1A(108)、绘制2-A(110))。

[0097] 图3B在步骤3继续所述过程,其中GPU 36接收新渲染目标,被标识为渲染目标B。举

例来说, GPU 36接收为渲染目标B设定色彩缓冲器(设定色彩缓冲器B)的指令(112),接着接收渲染目标B的第一绘制命令(绘制1-B)(114)。作为响应,针对渲染目标A的桶, GPU 36处理A的渲染(处理A的渲染)(116),且确定冲洗模式(确定A冲洗模式)(118),其还可被称作渲染模式。在图3B中所示的实例中, GPU 36确定渲染目标A的分格模式(当前桶ID(A-分格))(120),其与前导命令(前导IB2)相关联(122),以及当前在用于渲染目标A的桶中的命令(渲染IB2-0(124)、绘制1-A(126)、绘制2-A(128))。

[0098] 另外, GPU 36可开始通过产生渲染目标B的桶ID(获得B的渲染桶ID)(130)和更新正渲染的图像的格布局(更新格布局)(132)来为渲染目标B建立新的桶。 GPU 36还可包含用于产生恢复状态(134)以使 GPU 36返回到用于渲染的适当状态的命令。举例来说,当重新排序命令时,在执行新渲染命令之前, GPU 36返回到预期状态(例如将适当的数据存储到 GPU 存储器38)。 GPU 36接着可添加渲染IB2命令(渲染IB2-0)(136),其现在包含渲染目标A的第一绘制命令(绘制1-A)(138),渲染目标A的第二绘制命令(绘制2-A)(140),以及渲染目标B的第一绘制命令(绘制1-B)(142)。

[0099] 图3B在步骤4继续所述过程,其中 GPU 36接收渲染目标B的第二第一绘制命令B绘制2-B)(144)。 GPU 36可将先前在步骤3期间产生的第二绘制命令添加到IB2。就是说, GPU 36将渲染目标B的第二命令(绘制2-B)添加到IB2的结尾,使得IB2现在包含渲染目标A的第一绘制命令(绘制1-A)(148)、渲染目标A的第二绘制命令(绘制2-A)(150)、渲染目标B的第一绘制命令(绘制1-B)(152),以及渲染目标B的第二绘制命令(绘制2-B)(154)。

[0100] 图3C在步骤5继续所述过程,其中对于渲染目标A的第三绘制命令(绘制3-A)(158), GPU 36切换回到渲染目标A(设定色彩缓冲器A)(156)。当切换渲染目标时, GPU 36处理已为先前渲染目标分批的任何工作。因此, GPU 36处理B的渲染(处理B的渲染)(160),且为渲染目标B的桶确定冲洗模式(确定B冲洗模式)(162)。如相对于图3B所描述,对于桶ID 0(与渲染目标A相关联)(164), GPU 36包含渲染目标A的前导命令(前导IB2)(166),以及当前在渲染目标A的桶中的命令,包含渲染IB2-0(168)、绘制1-A(170)、绘制2-A(172)。作为对桶ID 1(与渲染目标B相关联)的新添加(174), GPU 36包含恢复状态命令(恢复状态IB2)(176)以及当前在渲染目标B的桶中的命令,包含渲染IB2-0(178)、绘制1-B(180)和绘制2-B(182)。

[0101] 另外,在图3C的实例中, GPU 36在与渲染目标A的前两个绘制命令分开的IB2中包含渲染目标A的第三绘制命令。举例来说, GPU 36获得渲染目标A的先前所产生的桶ID(获得A的渲染桶ID)(184),且更新正渲染的图像的格布局(更新格布局)(186)。 GPU 36还可包含用于产生恢复状态以使 GPU 36返回到用于渲染的恰当状态(产生恢复状态)的命令(188)。 GPU 36接着可添加渲染IB2命令(渲染IB2-1)(190),其现在包含渲染目标A的第三绘制命令(绘制3-A)(192)。虽然图3C将渲染目标A的第三绘制命令说明为包含于单独的IB2(IB2-1)中,但在其它实例中,可将渲染目标A的第三绘制命令添加到具有渲染目标A的前两个绘制命令的IB2(IB2-0)。

[0102] 图3D在步骤6继续所述过程,其中 GPU 36接收新渲染目标,被标识为渲染目标C。举例来说, GPU 36接收为渲染目标C设定色彩缓冲器(设定色彩缓冲器C)的指令(194),接着接收渲染目标C的第一绘制命令(绘制1-C)(196)。出于说明的目的,假定渲染目标C与渲染目标A或渲染目标B具有数据相依性。在此实例中, GPU 36可将渲染命令冲洗到命令缓冲器(例

如IB1缓冲器)以用于渲染。

[0103] 在另一实例中,应用程序可冲洗IB2 (198),例如以呈现将在显示单元42处渲染的数据。就是说,GPU 36可迫使IB2冲洗,而不管当数据准备好输出时遵循的命令。在任一情况下,如上所述,在切换渲染目标(或冲洗)之前,GPU 36处理已为先前渲染目标分批的任何工作。

[0104] 因此,GPU 36可处理渲染目标A的渲染命令(200)。举例来说,GPU 36为渲染目标A的桶确定冲洗模式(确定A冲洗模式)(202)。如上文所描述,对于桶ID 0(与渲染目标A相关联)(204),GPU 36包含渲染目标A的前导命令(前导IB2)(206),以及当前在渲染目标A的桶中的命令,包含渲染IB2-0(208)、绘制1-A(210)和绘制2-A(212)。作为对桶ID0(与渲染目标A相关联)的新添加,GPU 36还包含恢复状态命令(恢复状态IB2)(214),以及当前在第二IB2(渲染IB2-1)的渲染目标A的桶中的命令(216)以及绘制3-A(218)。另外,对于桶ID 1(与渲染目标B相关联)(220),GPU 36包含恢复状态命令(恢复状态IB2)(222)以及当前在渲染目标B的桶中的命令,包含渲染IB2-0(224)、绘制1-B(226)和绘制2-B(228)。

[0105] 以此方式,GPU 36可以与命令的初始次序不同的次序来确定渲染命令的执行次序。举例来说,GPU 36可使用相对于图3A到3D描述的经排序IB2参考来建立用于渲染的命令缓冲器。

[0106] 图4是说明符合本发明的技术的另一实例命令重新排序过程的流程图。举例来说,根据本发明的方面,GPU(例如GPU 36)可在两个重新排序模式(包含“重新排序打开”模式和“重新排序关闭”模式)下操作。GPU 36可默认在重新排序打开模式下操作,除非重新排序功能明确停用。

[0107] 当重新排序打开时,GPU 36将每一目标地图映射到桶,且用于渲染到所述渲染目标的命令累积在所述桶中。累加可在先前填充的桶中继续,只要累积的渲染命令的冲洗模式(例如渲染模式,例如直接渲染或分格)与传入渲染命令匹配即可。在图4中所示的实例中,渲染目标A与直接渲染模式以及渲染命令240的第一IB2链相关联。渲染目标B与硬件分格渲染模式以及渲染命令242的第二IB2链相关联。渲染目标C与直接渲染模式以及渲染命令244的第三IB2链相关联。

[0108] 在遇到与和已经与桶(例如IB2链,例如链240到244)相关联的命令不同的冲洗模式(例如渲染模式)相关联的渲染命令后,GPU 36可即可关闭重新排序(切换到重新排序关闭模式)。在图4的实例中,并不具有与已经定义的桶相同的冲洗模式的实例命令包含具有分格模式的渲染目标A的渲染命令、具有直接模式的渲染目标B的渲染命令、具有软件分格模式的渲染目标B的渲染命令,或具有分格模式的渲染目标C的渲染命令。如上所述,当遇到所建立的桶的渲染命令之间的数据相依性时,GPU 36也可关闭重新排序。数据相依性的实例可为用正用作渲染目标A的纹理的渲染目标B的数据来渲染渲染目标A。

[0109] 在已关闭重新排序之后,GPU 36可产生渲染目标的新桶。就是说,GPU 36可不组合渲染目标的命令与渲染目标的先前定义桶的命令,即使已经为特定渲染目标建立了桶。在图4中所示的实例中,GPU 36可产生渲染目标A(直接渲染模式)、渲染目标B(硬件分格渲染模式)和渲染目标C(直接渲染模式)的新桶。

[0110] 当GPU 36将渲染命令冲洗到用于渲染的硬件时,GPU 36根据与桶相关联的渲染模式(例如直接渲染或分格)来处理所述桶中的每一者,将桶的命令插件到IB1中,且提交用于

渲染的IB1。可将每一桶的条目概念化为添加到IB1的工作的简写版本。

[0111] 虽然相对于重新排序打开或关闭来描述图4的实例,在其它实例中,GPU 36可使用另一经细化方法,其中可在每桶基础上接通或关闭重新排序。可依靠额外相依性跟踪来实现此类细化。

[0112] 图5A到5B是说明符合本发明的技术的另一实例命令重新排序过程的流程图。在一些情况下,图5A和5B中所示的实例命令可以与上文相对于图3A到3D描述的方式类似的方式以执行次序重新排序。然而,图5A到5B的实例包含若干块传送 (BLT) 命令。

[0113] 一般来说,执行BLT命令将数据从源位置复制到目的地位置。正从所述源位置复制的数据可与目的地位置的数据组合。BLT命令可或可不与当前渲染目标相关联。BLT命令可内联或异步执行。举例来说,内联BLT可连同渲染命令一起执行,且BLT的目的地可与正执行的那批渲染命令的渲染目标相同。异步BLT并不具有对渲染目标(或来源纹理)的相依性。在一些实例中,异步BLT可重新排序为正执行的第一批命令。

[0114] 根据本发明的方面,可以与其它渲染命令相同的方式来处置BLT命令。举例来说,在遇到BLT命令后,GPU (例如GPU 36)可即刻确定BLT命令是否具有与其它渲染目标的数据相依性。如果BLT命令取决于一或多个先前命令的结果,那么GPU 36可以与其它渲染命令相同的方式来处置BLT,如上文所描述。举例来说,GPU 36可冲洗当前桶或创建新桶(而不冲洗),并关闭命令重新排序。

[0115] 相对于图5A中所示的实例,GPU 36可接收渲染目标A的第一渲染命令(FBO A-绘制0)以及渲染目标A的第二渲染命令(FBO A-绘制1)。响应于接收到所述命令,GPU 36可产生新桶(桶A)(260),并将第一和第二渲染命令添加到桶(绘制0、绘制1)(262)。

[0116] GPU 36接着接收BLT命令(异步BLT),其将Y(出于实例的目的,任意位置)的内容复制到Z(另一任意位置)。响应于接收到BLT命令,GPU 36可产生新BLT桶(异步BLT IB2链)(264),并将Y添加到Z BLT(266)。在一些实例中,GPU 36可将异步BLT命令与其它渲染命令分开执行。如上所述,异步BLT并不具有对渲染目标的相依性,且GPU 36可将异步BLT重新排序成正执行的第一批命令。虽然GPU 36通常可以正执行的第一批命令来执行异步BLT,但在一些实例中,GPU 36可创建新桶,并像新桶与对新渲染目标的新绘制相关联那样处理新桶。

[0117] GPU 36接着接收渲染目标B的第一渲染命令(FBO B-绘制0)。响应于接收到所述命令,GPU 36可产生新桶(桶B)(268),并将第一渲染命令添加到所述桶(绘制0)(270)。

[0118] GPU 36接着接收额外异步BLT命令,其将W(任意位置)的内容复制到X(另一任意位置)。响应于接收到BLT命令,GPU 36可将BLT命令添加到异步BLT IB2链(272)。

[0119] GPU 36接着接收渲染目标A的第三渲染命令(FBO A-绘制2)。响应于接收到所述命令,GPU 36可将渲染命令添加到渲染目标A的先前所产生的桶A(274)。在一些情况下,GPU 36可在将命令添加到桶A之前,确定第三渲染命令(绘制2)是否具有任何相关联数据相依性。

[0120] GPU 36接着接收内联BLT命令,其将C(任意位置)的内容复制到渲染目标A。响应于接收到BLT命令,GPU 36可将BLT命令添加到渲染目标A的先前所产生的桶A(276)。并且,GPU 36可在将命令添加到桶A之前,确定BLT命令(C到A)是否具有任何相关联数据相依性。

[0121] GPU 36接着接收渲染目标B的第二渲染命令(FBO B-绘制1)。响应于接收到所述命令,GPU 36可将第二渲染命令添加到渲染目标B的先前所产生的桶B(278)。

[0122] GPU 36接着接收从渲染目标A到位置D的BLT,其包含数据相依性。此时,GPU36可冲洗所产生的命令列表,归因于数据相依性(280)。在另一实例中,GPU 36可关闭命令重新排序,且可产生新的BLT桶(桶BLT)(282),并将BLT命令添加到新的BLT桶(284)。

[0123] 在重新排序关闭的情况下,GPU 36可不将命令添加到先前所产生的桶,即使传入命令与具有先前建立的桶的渲染目标相关联也是如此。因此,当GPU 36接收渲染目标B的第三渲染命令(FBO-绘制2),GPU 36可产生新桶(桶B'),且将第三渲染命令添加到新桶(288)。在其它实例中,可不关闭重新排序。举例来说,关闭重新排序可帮助减少跟踪的相依性的数目。然而,在不重新排序的情况下,GPU 36可对渲染目标B的第三渲染命令(FBO-绘制2)进行重新排序,提供渲染目标B的第三渲染命令(FBO-绘制2)并不具有对渲染目标A或位置D的相依性。在此实例中,GPU 36可对渲染目标B的第三渲染命令(FBO-绘制2)进行重新排序,以遵循渲染目标B的第二渲染命令(绘制1)。

[0124] GPU 36接着可接收将渲染内容冲洗到外部存储器的命令(冲洗)。响应于接收到冲洗命令,GPU 36可使用经重排序的渲染命令来建立命令缓冲器(IB1)。就是说,GPU 36可依序将桶A、桶B、桶BLT和桶B'的所述所确定的命令列表添加到命令缓冲器。以此方式,GPU 36推迟命令缓冲器的建立,直到在接收到并灵活地重新排序命令缓冲器的渲染命令之后。

[0125] 图6是说明符合本发明的技术的另一实例命令重新排序过程的流程图。明确地说,图6的实例说明其中渲染目标之间发生相依性的实例。举例来说,GPU 36可接收渲染目标A的第一渲染命令(FBO A-绘制0)以及渲染目标A的第二渲染命令(FBO A-绘制1)。响应于接收到所述命令,GPU 36可产生新桶(桶A)(300),并将第一和第二渲染命令添加到桶(绘制0、绘制1)(302)。

[0126] GPU 36接着可接收渲染目标B的第一渲染命令(FBO B-绘制0)。响应于接收到所述命令,GPU 36可产生新桶(桶B)(304),并将第一渲染命令添加到所述桶(绘制0)(306)。

[0127] GPU 36接着可接收渲染目标A的第三渲染命令(FBO A-绘制2),其使用渲染目标B作为纹理。在此实例中,渲染目标A的第三渲染命令取决于渲染目标B。如上所述,如果在重新排序后,与当前命令相关联的读取时戳在与当前命令所取决于的命令相关联的写入时戳之后出现,那么可识别到相依性。

[0128] 举例来说,出于说明的目的,对渲染目标A的第三渲染命令进行重新排序(FBO A-绘制2)将致使渲染目标A的第三渲染命令添加到桶A。然而,渲染目标A的第三渲染命令(FBO A-绘制2)取决于渲染目标B。就是说,与渲染目标A的第三渲染命令(FBO A-绘制2)相关联的读取时戳在与渲染目标B相关联的写入时戳之后出现,因为渲染目标B的内容必须可用于将恰当地计算的渲染目标A的第三渲染命令。因此,GPU 36可基于所述时戳来识别此相依性,并关闭重新排序。

[0129] 在关闭重新排序之后,GPU 36可冲洗命令列表,或产生新的桶(桶A')(308),其包含渲染目标A的第三渲染命令(FBO A-绘制2)(310)。

[0130] 图7是说明根据本发明的方面的用于对命令进行重新排序的实例过程的流程图。虽然将图7中(以及本发明中其它地方)所示的过程描述为由CPU 32和/或GPU 36(图1和2)进行,但所述技术可由多种其它处理单元实施。

[0131] GPU 36以初始次序接收命令(320)。举例来说,GPU 36可以其中命令由GPU程序52指定的次序来从GPU驱动程序50接收命令。所述命令可包含多个渲染目标的多个渲染命令。

[0132] 根据本发明的方面, GPU 36可确定所述命令的执行次序(322)。在确定执行次序时, GPU 36可基于命令之间的数据相依性, 以与初始次序不同的次序来对所述命令中的一或多者重新排序。举例来说, 如本文所述, GPU 36可产生每渲染目标命令列表, 且可将渲染命令添加到所述列表, 假定正添加的所述命令并不取决于另一命令的结果。可对此类无关命令(不具有相依性的命令)重新排序, 以消除不必要的渲染目标切换而无较复杂的相依性跟踪, 因而减少GPU 36所执行的加载和存储操作的数目。

[0133] GPU 36接着可使用所述所确定的执行次序来执行所述命令(324)。举例来说, GPU 36可以所述所确定的执行次序将所产生的渲染命令列表冲洗到命令缓冲器。GPU 36接着可执行来自命令缓冲器的命令, 如通常由GPU 36执行那样。

[0134] 还应理解, 取决于实例, 本文中所描述的方法中的任一者的某些动作或事件可以不同顺序执行、可添加、合并或全部省略(例如, 实践所述方法并不需要所有所描述动作或事件)。此外, 在某些实例中, 可(例如)通过多线程处理、中断处理或多个处理器同时而非顺序地执行动作或事件。

[0135] 在一或多个实例中, 所描述功能可硬件、软件、固件或其任何组合来实施。如果以软件实施, 那么功能可作为一或多个指令或代码存储在包括非暂时性计算机可读媒体的制品上。计算机可读媒体可包含计算机数据存储媒体。数据存储媒体可为可由一或多个计算机或一或多个处理器存取以检索用于实施本发明中描述的技术的指令、代码和/或数据结构的任何可用媒体。作为实例而非限制, 此类计算机可读媒体可包括RAM、ROM、EEPROM、CD-ROM或其它光盘存储装置、磁盘存储装置或其它磁性存储装置、快闪存储器或可用来运载或存储呈指令或数据结构形式的所要程序代码且可由计算机存取的任何其它媒体。如本文中所使用, 磁盘和光盘包含压缩光盘(CD)、激光光盘、光学光盘、数字多功能光盘(DVD)、软性磁盘和蓝光光盘, 其中磁盘通常以磁性方式再现数据, 而光盘利用激光以光学方式再现数据。以上各项的组合也应包含在计算机可读媒体的范围内。

[0136] 代码可由一或多个处理器执行, 所述一或多个处理器例如一或多个DSP、通用微处理器、ASIC、FPGA, 或其它等效集成或离散逻辑电路。另外, 在一些方面中, 可在专用硬件和/或软件模块内提供本文中所描述的功能性。而且, 所述技术可完全实施于一或多个电路或逻辑元件中。

[0137] 本发明的技术可在各种各样的装置或设备中实施, 包含无线手持机、集成电路(IC)或一组IC(例如, 芯片组)。本发明中描述各种组件、模块或单元是为了强调经配置以执行所公开的技术的装置的功能方面, 但未必需要由不同硬件单元实现。实际上, 如上文所描述, 各种单元可以结合合适的软件及/或固件组合在编码解码器硬件单元中, 或者通过互操作硬件单元的集合来提供, 所述硬件单元包含如上文所描述的一或多个处理器。

[0138] 已描述了各种实例。这些和其它实例在所附权利要求书的范围内。

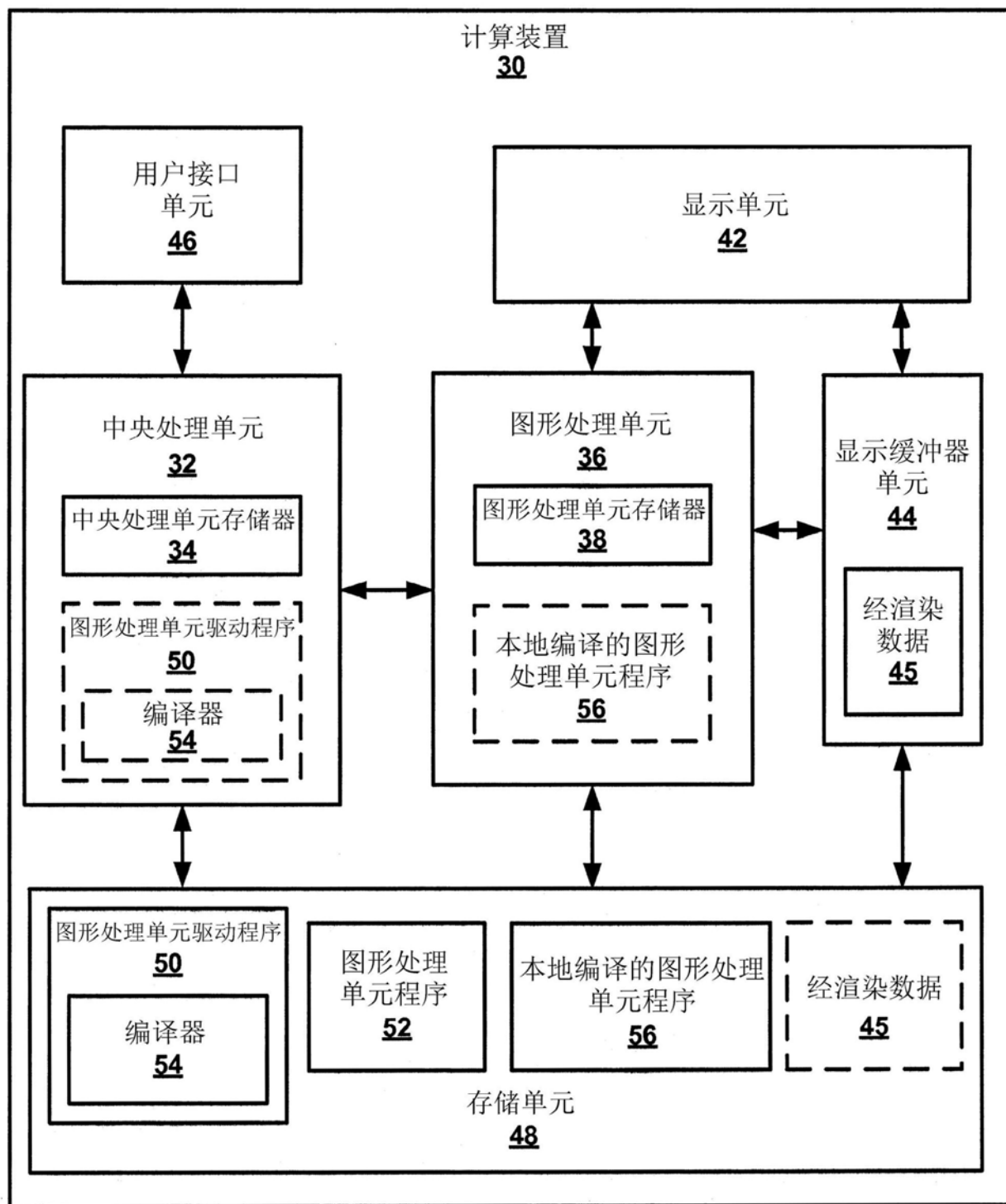


图1

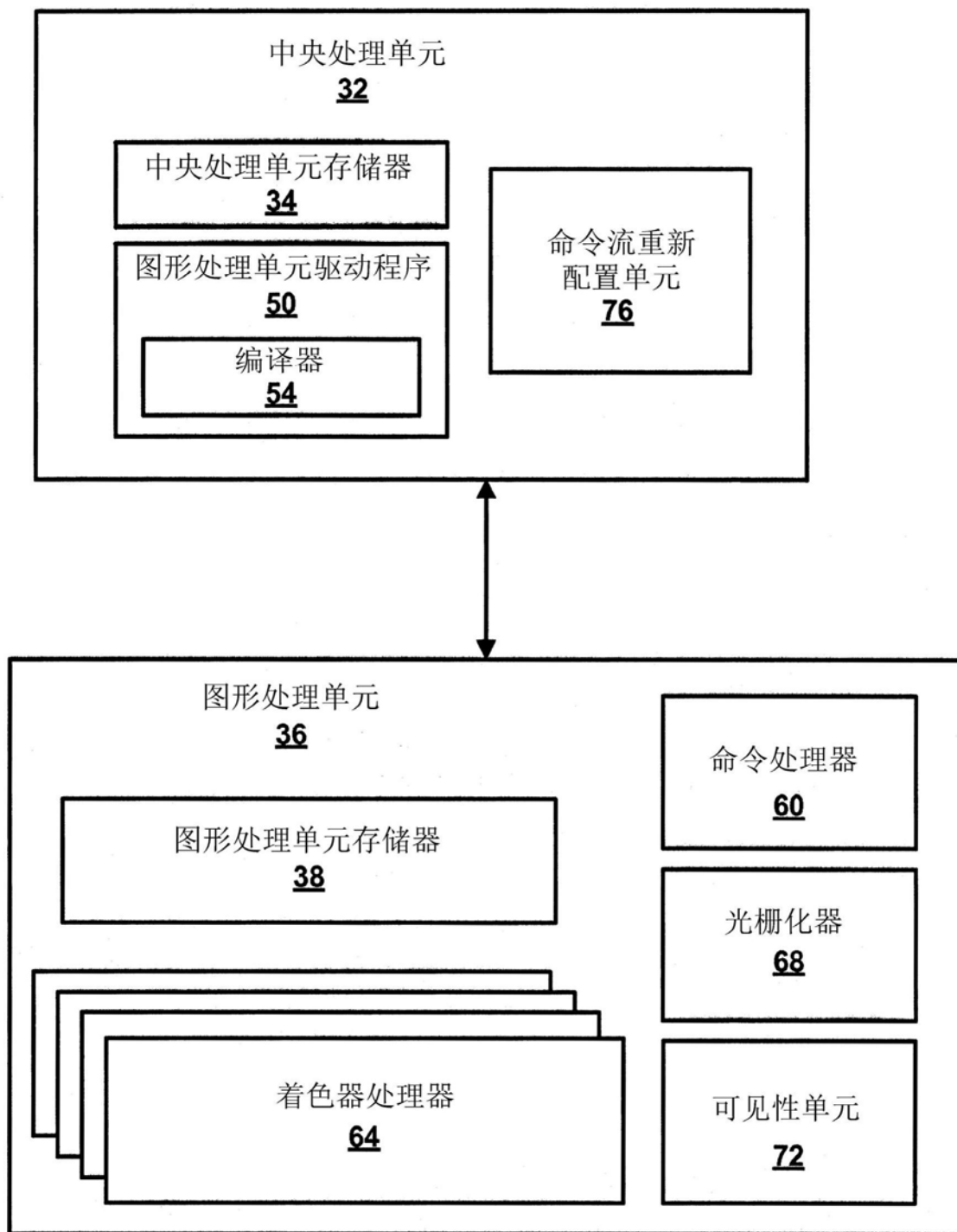


图2



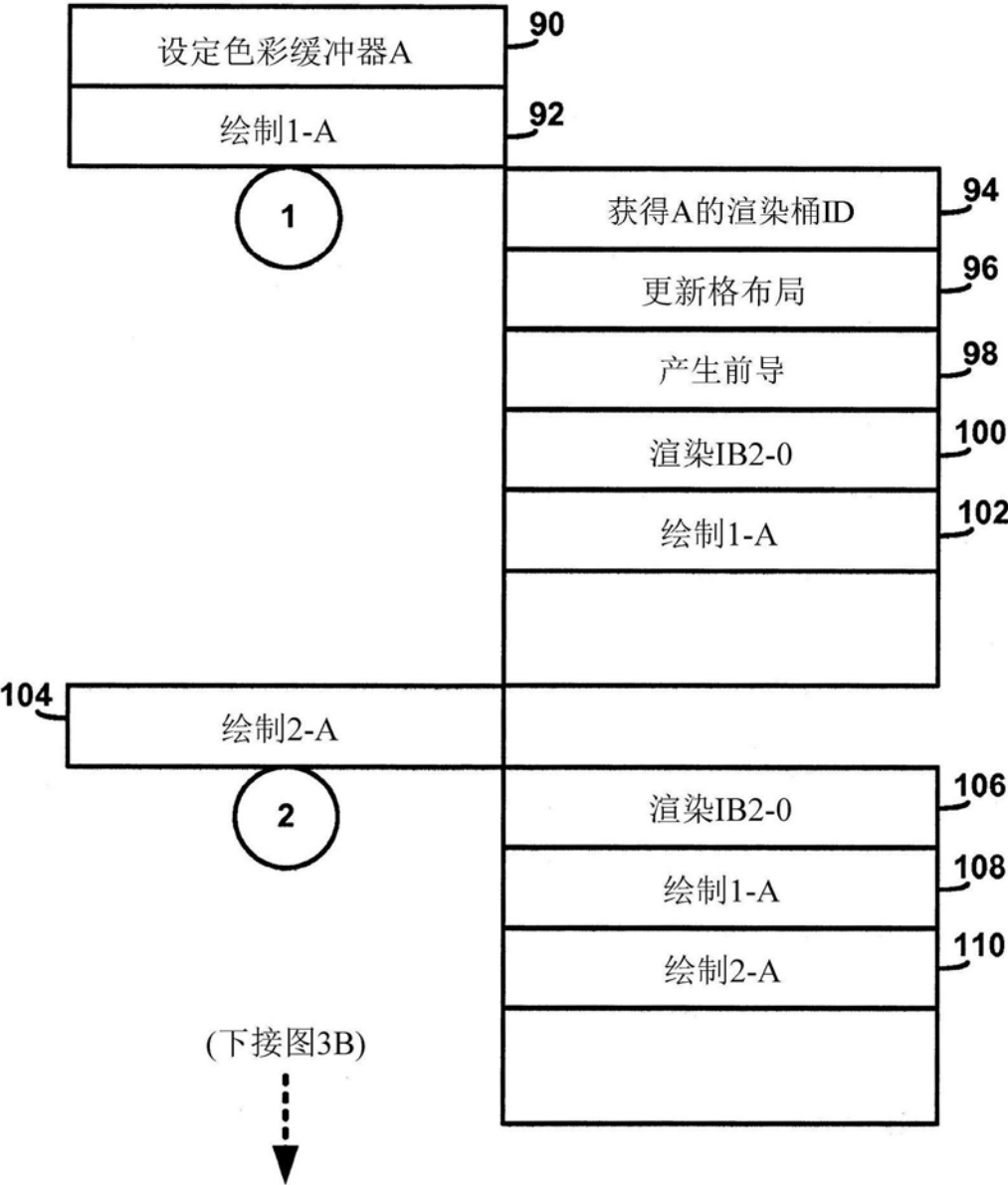


图3A

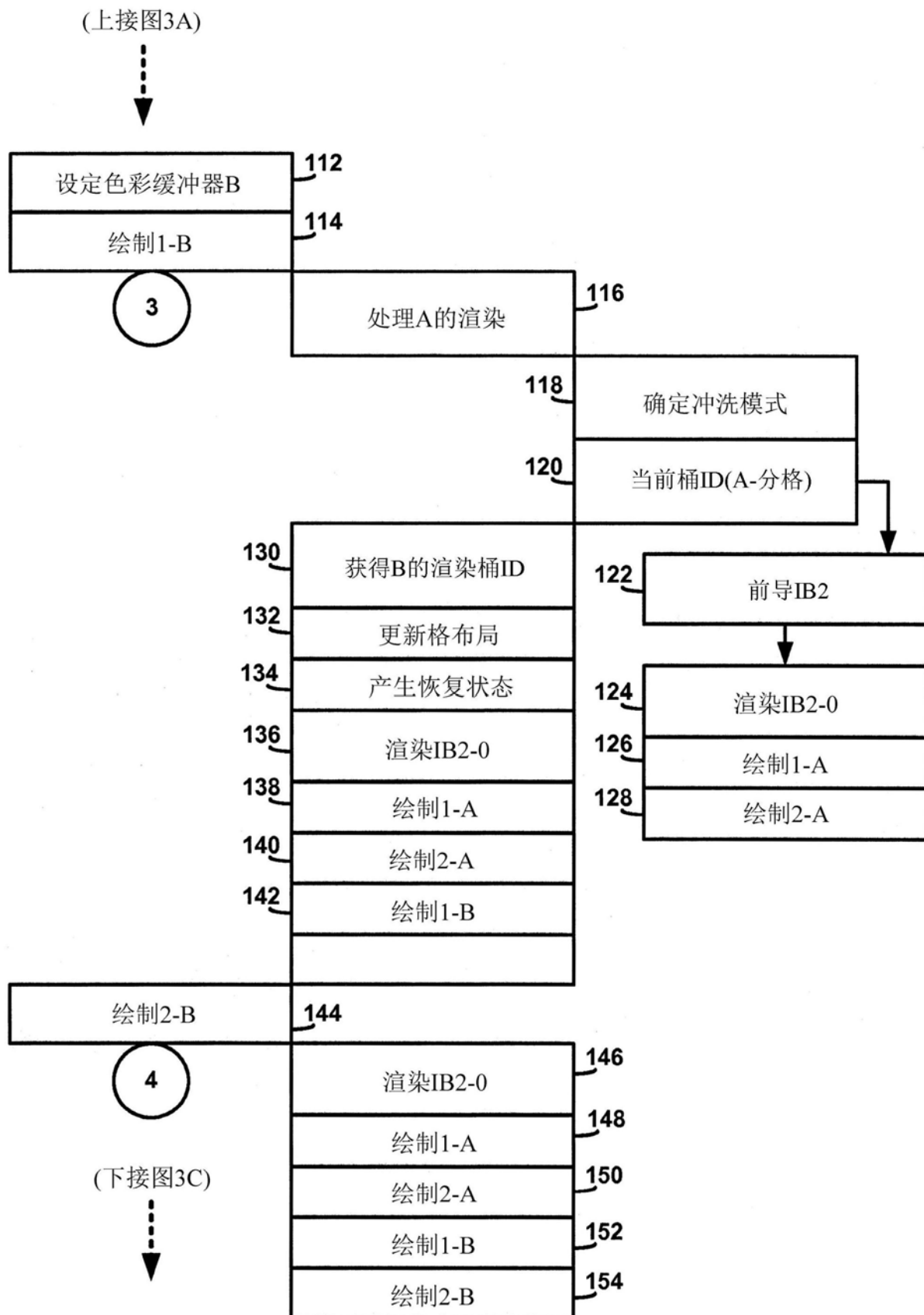


图3B

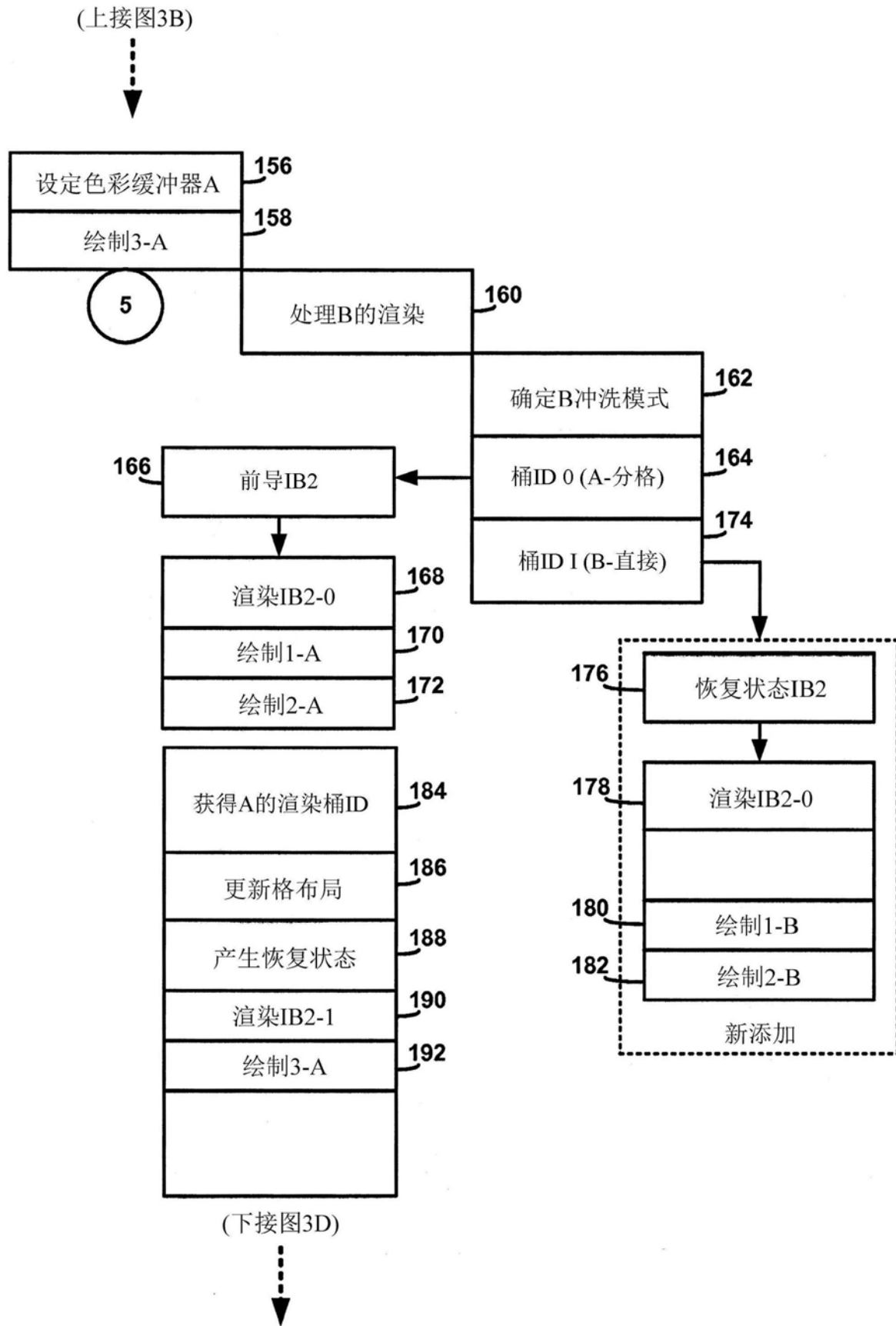


图3C

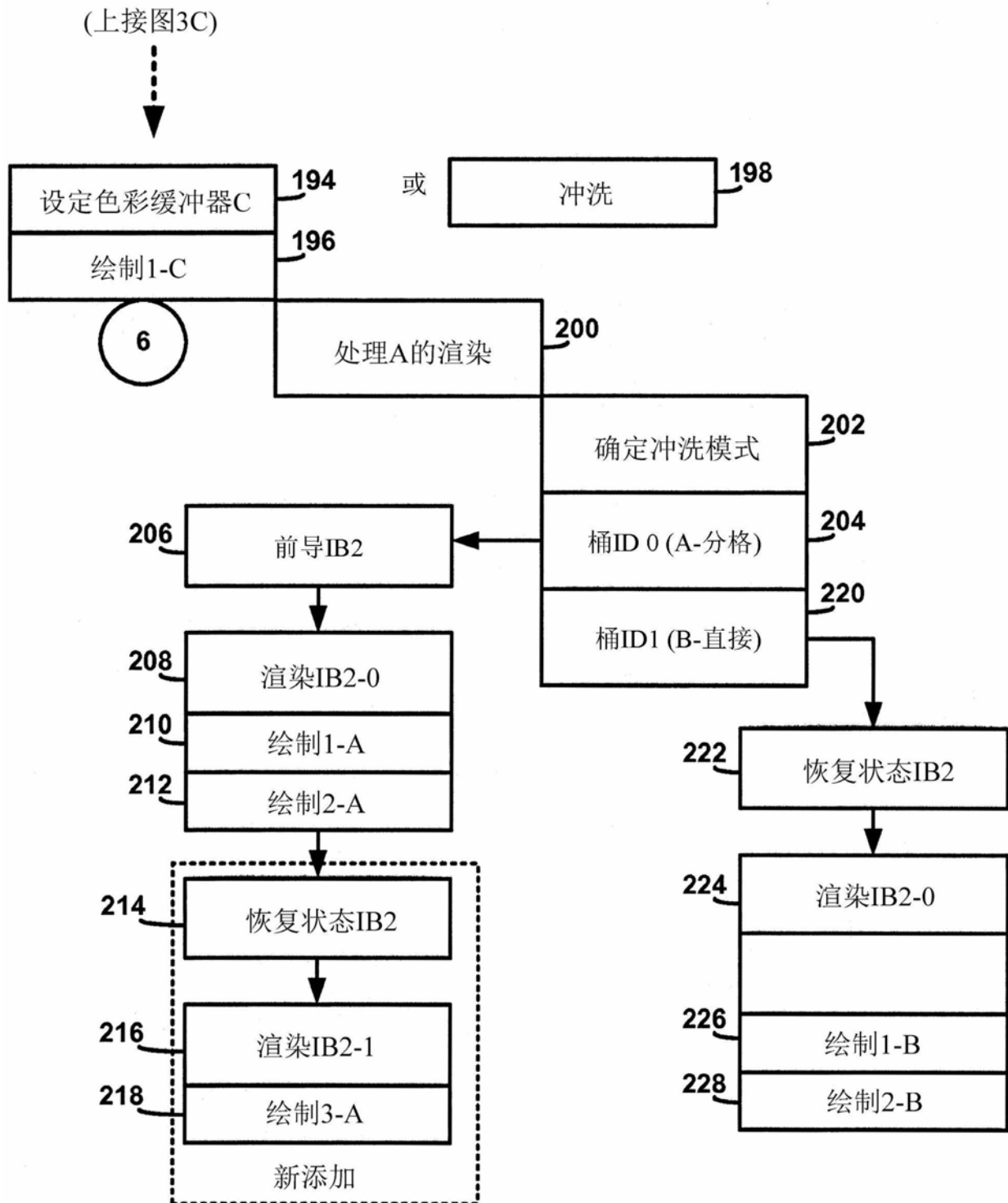


图3D

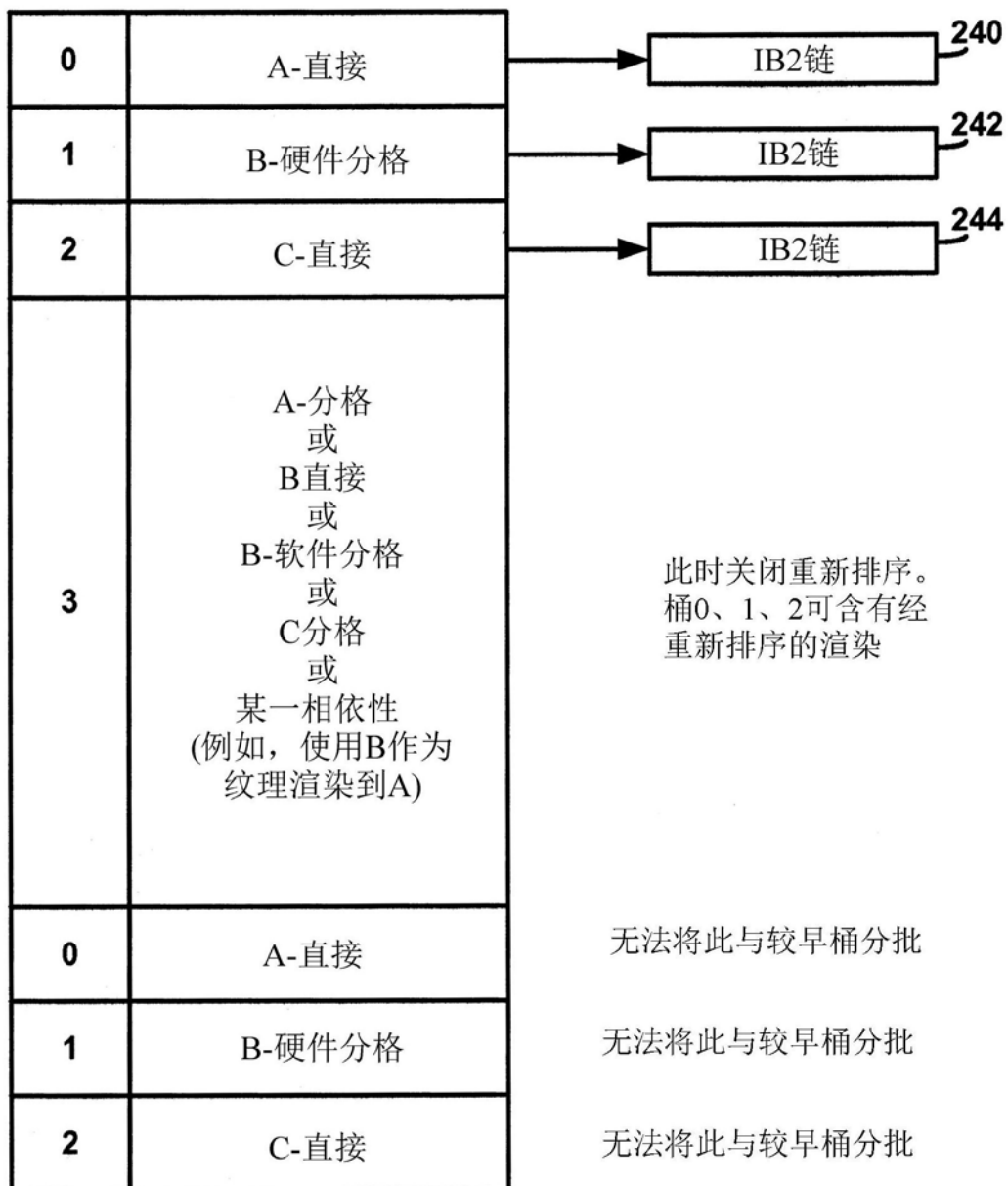


图4

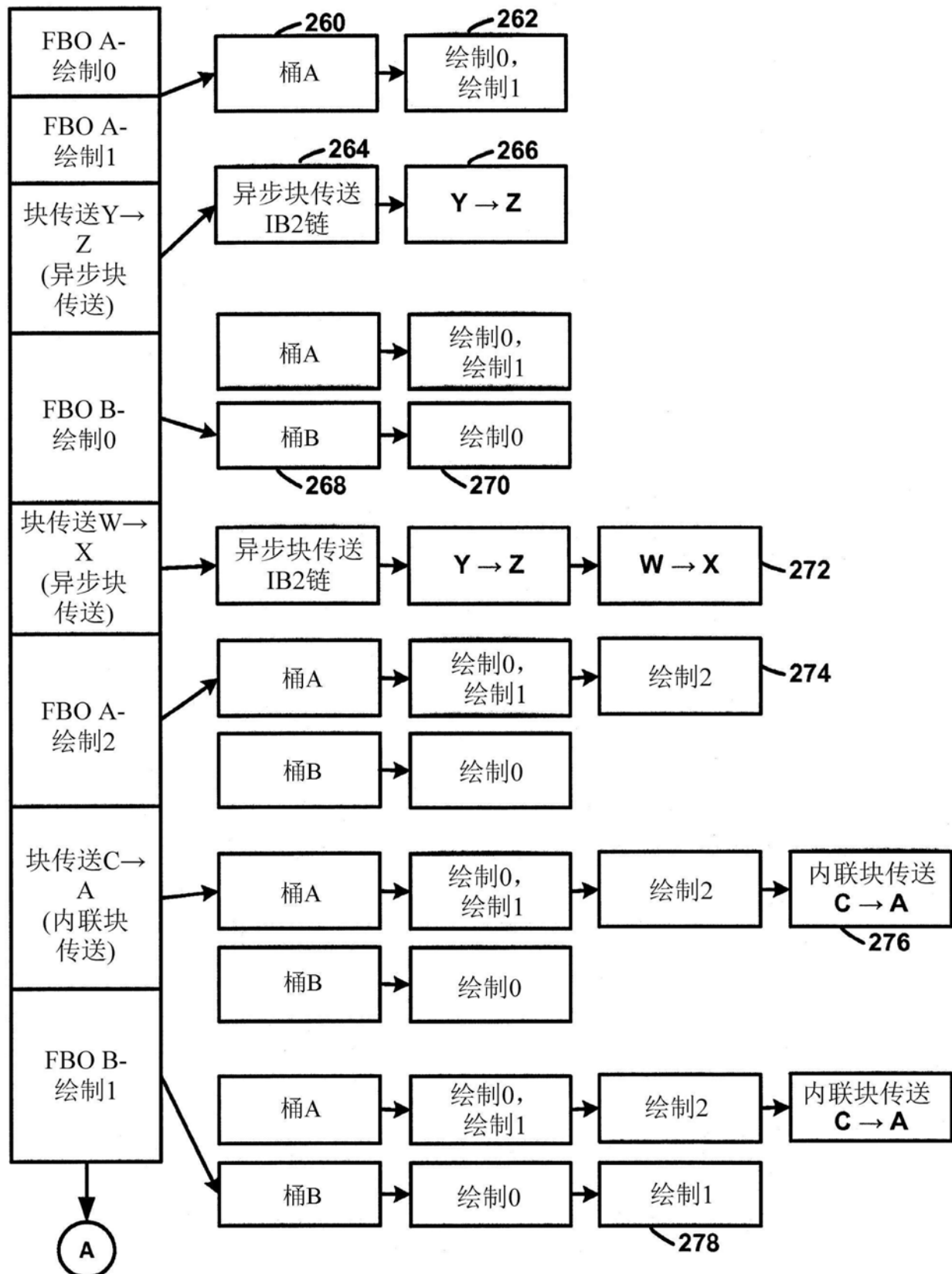


图5A

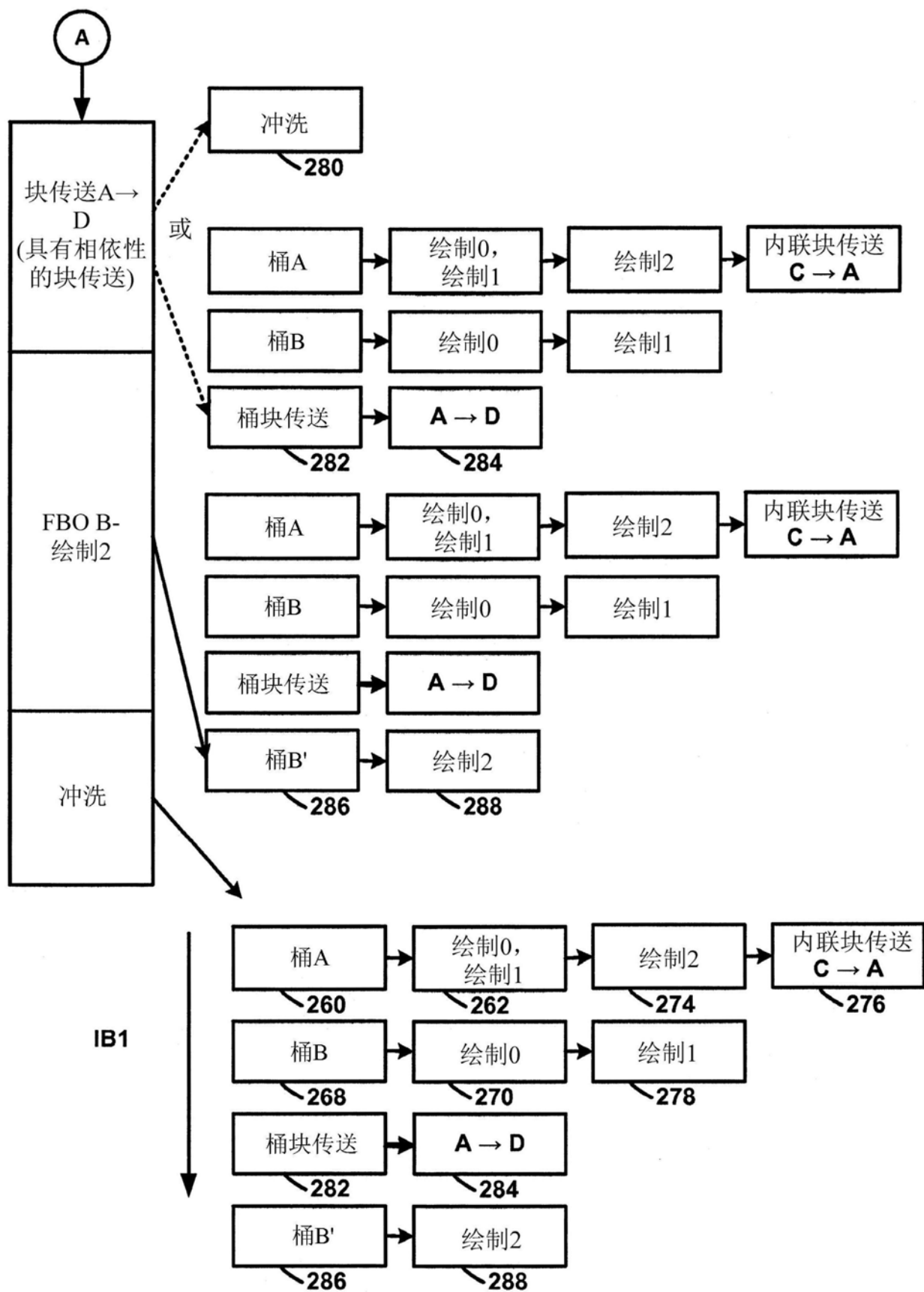


图5B

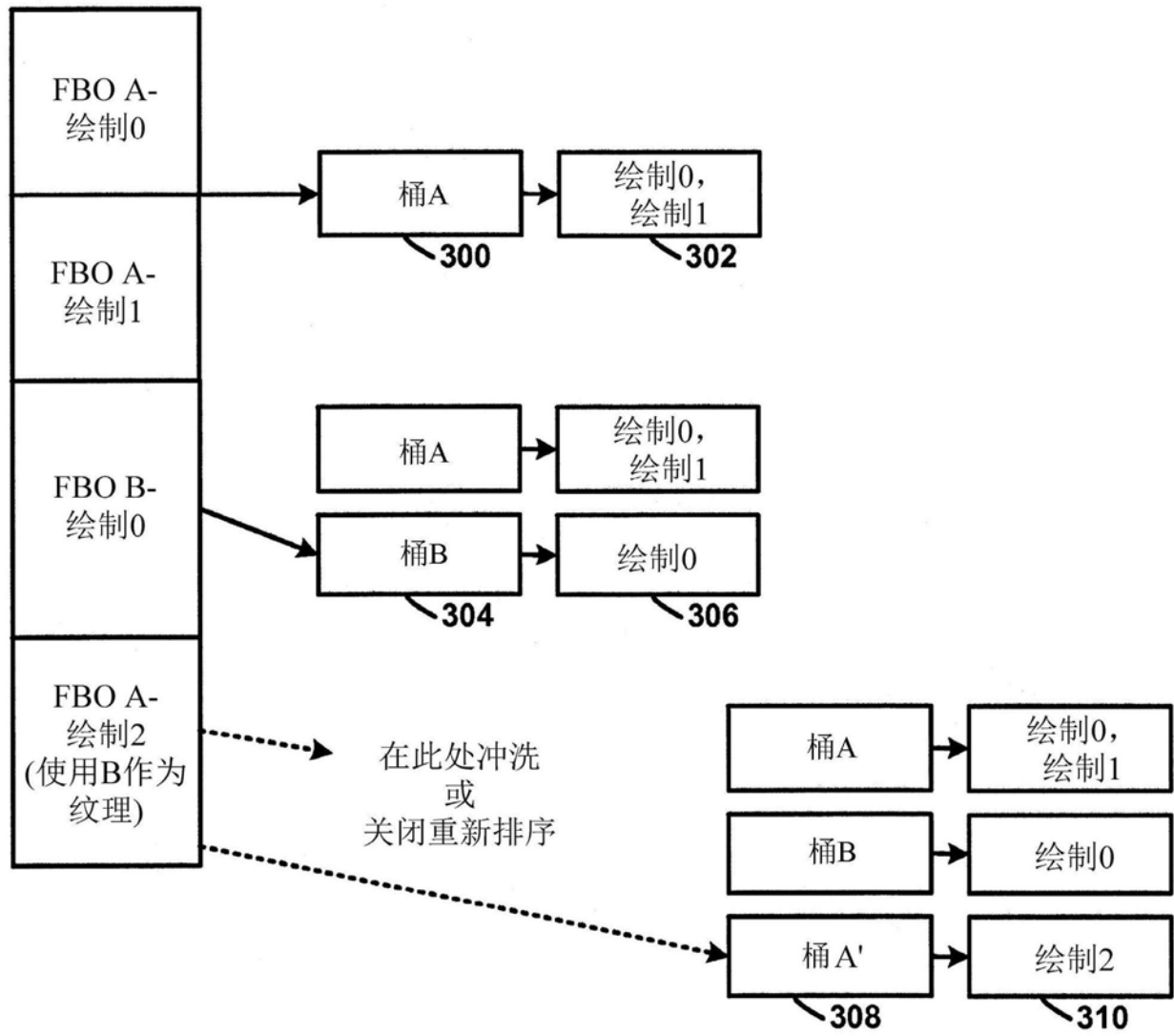


图6



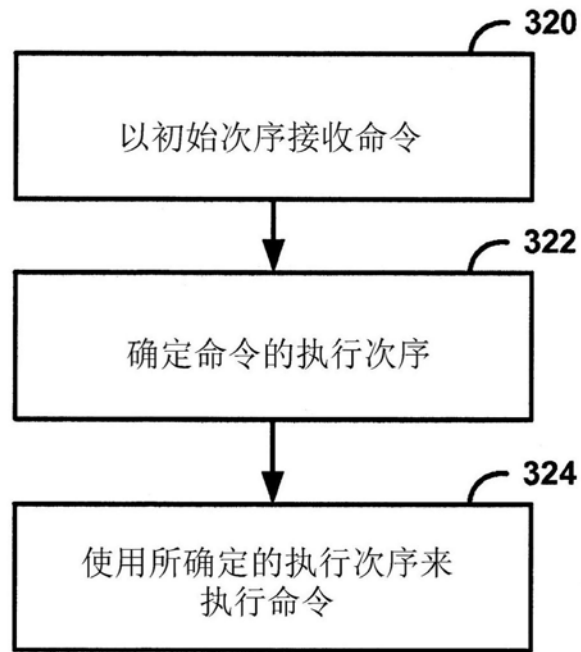


图7