



(12) 发明专利申请

(10) 申请公布号 CN 101965610 A

(43) 申请公布日 2011. 02. 02

(21) 申请号 200880125863. 4

代理人 程伟 王锦阳

(22) 申请日 2008. 12. 01

(51) Int. Cl.

(30) 优先权数据

G09G 5/36 (2006. 01)

11/987, 559 2007. 11. 30 US

G09G 5/395 (2006. 01)

(85) PCT申请进入国家阶段日

2010. 07. 29

(86) PCT申请的申请数据

PCT/US2008/085160 2008. 12. 01

(87) PCT申请的公布数据

W02009/073617 EN 2009. 06. 11

(71) 申请人 ATI 技术有限责任公司

地址 加拿大安大略省

(72) 发明人 J·程 T·拉维奥莱特 J·黄

R·扎布聚茨 J·朗 X·翁

S·马林科维奇 P·穆玛 M·简

M·特里西德 R·萨尔特切瓦 G·谢

I·利特奇马诺夫

(74) 专利代理机构 北京戈程知识产权代理有限

公司 11314

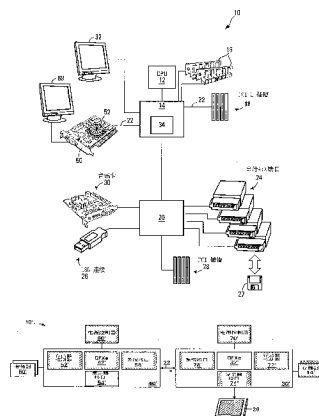
权利要求书 2 页 说明书 11 页 附图 13 页

(54) 发明名称

跨越高速外围互连总线的视频渲染

(57) 摘要

由一个图形处理器产生的图形经由高速互连总线而传送至帧缓冲器。来自该帧缓冲器的该渲染过的帧经由与该帧缓冲器通信的显示器接口而呈现在显示器上。另一个存在的(例如,集成的)图形适配器/子系统的显示器接口可用来将该渲染过的帧呈现于互连的显示器上。



1. 一种操作计算器件的方法,该计算器件包括处理器、与该处理器通信的存储器、将该处理器互连至图形处理器的外围互连总线、和显示器接口,该方法包括:

在该存储器中建立帧缓冲器;

指示该图形处理器渲染视频帧;

将该视频帧经由该外围互连总线传送至该帧缓冲器内;以及

将来自该帧缓冲器的该帧经由该显示器接口呈现在显示器上。

2. 如权利要求 1 所述的方法,其中,该指示该图形处理器包括将帧渲染至第二帧缓冲器内,以及将来自该第二帧缓冲器的该帧传送至该帧缓冲器。

3. 如权利要求 2 所述的方法,其中,该图形处理器是第一图形适配器的一部分,而该第二帧缓冲器是在第二图形适配器的本地存储器中。

4. 如权利要求 1 所述的方法,其中,该传送包括指示该图形处理器包括经由该外围互连总线直接渲染至该帧缓冲器内。

5. 如权利要求 1 所述的方法,其中,该指示包括指示该图形处理器将该视频帧经由该外围互连总线传送至该帧缓冲器内。

6. 如权利要求 1 所述的方法,其中,该指示包括指示直接存储器访问控制器,以将该视频帧传送至该帧缓冲器内。

7. 如权利要求 1 所述的方法,进一步包括编程该显示器接口,以将该帧缓冲器中的该帧呈现在该显示器上。

8. 如权利要求 3 所述的方法,其中,该呈现包括编程该显示器接口,以显示来自该第二帧缓冲器的图像。

9. 如权利要求 1 所述的方法,其中,该指示该图形处理器包括指示该图形处理器以渲染与该图形处理器通信的缓冲器中的该帧,以及其中,该传送包括直接存储器访问将该帧从该缓冲器传送至该帧缓冲器。

10. 一种计算器件,包括:

中央处理器;

存储器,与该中央处理器通信;

外围互连总线,将该中央处理器互连至图形处理器;

显示器接口;

计算机可执行指令,储存在该存储器中,适配该计算器件:

使该图形处理器渲染视频帧;

将该视频帧经由该外围互连总线传送至帧缓冲器内;

将来自该帧缓冲器的该渲染过的帧经由该显示器接口呈现在显示器上。

11. 如权利要求 10 所述的计算器件,其中,该显示器接口形成外围扩充接口的一部分。

12. 如权利要求 11 所述的计算器件,其中,该图形处理器形成与该外围互连总线通信的外围扩充插槽中的外围扩充卡的一部分。

13. 如权利要求 10 所述的计算器件,其中,该图形处理器形成图形子系统的一部分,该部分缺少帧缓冲存储器和显示器接口。

14. 如权利要求 10 所述的计算器件,其中,该视频帧通过将该视频帧直接渲染至该帧缓冲器的方式,经由该外围互连总线传送至该帧缓冲器内。

15. 一种计算器件,包括:
中央处理器;
存储器,与该处理器通信;
第一帧缓冲器;

外围互连总线,将该处理器互连至图形子系统,该图形子系统缺少将帧缓冲器本地界定至该图形子系统的存储器;以及

显示器接口,与该存储器通信;

该存储器储存适配该图形处理器的计算机可执行指令,以渲染视频帧,并将该视频帧经由该外围互连总线传送至该第一帧缓冲器内。

16. 如权利要求 15 所述的计算器件,其中,该图形子系统形成在与该外围互连总线互连的外围扩充卡上。

17. 如权利要求 15 所述的计算器件,进一步该显示器接口形成外围扩充接口的一部分。

18. 如权利要求 15 所述的计算器件,其中,该显示器接口形成外围扩充接口上的该图形子系统的一部分,该部分将该中央处理器互连至该外围互连总线。

19. 一种图形子系统,包括:

图形处理器、和外围互连总线接口,用来将该图形子系统互连至主处理器,该图形子系统缺少将帧缓冲器本地界定至该图形子系统的存储器;以及

该图形子系统操作以渲染视频帧,并将该视频帧经由该外围互连总线传送至帧缓冲器内。

20. 一种计算机可读介质,储存计算可执行指令,该计算可执行指令适配计算器件以执行如权利要求 1 所述的方法。

跨越高速外围互连总线的视频渲染

[0001] 相关申请案的交互参考

[0002] 本发明是在 2005 年 5 月 24 日提出申请的美国专利申请案第 11/136,483 号和在 2006 年 5 月 30 日提出申请的美国专利申请案第 11/421,005 号的部分连续案 (continuation-in-part), 所述专利申请案的内容在此并入本发明中作为参考。

技术领域

[0003] 本发明通常涉及计算器件, 特别是涉及包含图形处理器的计算器件, 该图形处理器互连于高速外围总线。

背景技术

[0004] 近年来, 计算硬件已经变得越来越集成了。计算机外围通常形成如集成芯片组中与中央处理器共同操作的部分。通常, 计算机音讯、图形适配器和接口形成该集成芯片组的一部分。这种集成降低该计算器件的成本和尺寸。

[0005] 在此同时, 图形处理器是稳定地发展, 以提供增强的效能和特征。先进的图形处理器已几乎与中央处理器同样复杂了。因此, 尽管集成的潮流, 高效能图形处理器仍通常形成在外围卡上。集成图形处理器比较起来相当简单, 并且没有提供形成在外围卡上的高效能图形处理器的特征或效能。在附加 (add-on) 外围卡出现的情况下, 集成图形组件 (当出现时) 通常是多余的并因此是失能的 (disabled)。

[0006] 类似地, 当图形适配器 (不论是集成的或外部的) 升级时, 较不具效力的适配器通常没有适用余地, 因此, 这些会被抛弃或失能。

[0007] 因此, 仍需降低组件的多余性。

发明内容

[0008] 依据本发明的方面, 由一个图形处理器产生的图形是跨越高速扩充 / 互连总线传送至帧缓冲器。来自该帧缓冲器的该渲染过的帧是经由与该帧缓冲器通信的显示器接口而呈现在显示器上。

[0009] 方便地, 另一个现存的 (集成的) 图形适配器 / 子系统的显示器接口可用来将该渲染过的帧呈现在互连显示器上。

[0010] 以这种方式, 该图形处理器可形成图形适配器中不需包含显示器接口或可能甚至帧缓冲器的部分。同样地, 该图形处理器可被更流行的图形处理器取代。

[0011] 该图形处理器可形成图形适配器中包含在外围扩充卡上的部分。

[0012] 在一个实施例中, 计算器件可包含第一和第二图形适配器。将被显示在显示器上的图像被传送至可被该第二图形适配器访问的存储器。该第二适配器的显示器接口是用来将该第二图形适配器的存储器中的图像呈现在该多个显示器中至少一者。以这种方式, 形成该显示器接口和该第二适配器的端口 (port) 的器件电子设备可因此利用。渲染过的图形可以一次传送一帧或一图形基元 (primitive) 的方式被传送。在后者的例子中, 图形是

直接地由该第二图形适配器所访问的存储器中被有效地渲染。该存储器可为系统存储器或适配器存储器。

[0013] 依据另一个实施例,该第二适配器的图形处理器可失能或置于低电源模式,而该第一图形处理器则渲染跨越该高速互连总线传送的帧。

[0014] 依据本发明的另一个方面,形成第一图形适配器的一部分的图形处理器是操作以分别在第一和第二显示器上绘制第一和第二图像。该第一显示器可与该第一图形适配器的第一显示器接口互连。该第二显示器可与第二图形适配器的第二显示器接口互连。该方法包含渲染该第一图形适配器的存储器中的该第一和第二图像;将来自于该第一适配器的该存储器的该第二图像传送至该第二图形适配器的帧缓冲器;以及,使用该第二图形适配器的该第二显示器接口,将来自于该第二图形适配器的该帧缓冲器的该第二图像显示在该第二显示器上。

[0015] 依据本发明的另一个方面,计算器件包含:处理器;与该处理器通信的计算机可读取存储器;第一图形适配器;与该第一图形适配器通信的第一适配器存储器;用来将图像呈现在第一显示器上的第一显示器接口;第二图形适配器;用来将图像呈现在第二显示器上的第二显示器接口;与该第一和第二图形适配器互连的高速总线。程序代码是储存在该计算机可读取存储器中,并可被该处理器执行。该程序代码当执行时,使该第一图形适配器渲染该第一适配器存储器中的第一和第二图像;使来自于该第一适配器存储器的该第二图像传送至该第二图形适配器的帧缓冲器;以及,使该第二图形适配器的该第二显示器接口被编程以将来自于该第二图形适配器的该帧缓冲器的该第二图像显示在该第二显示器上。

[0016] 依据另一个方面,该第一适配器将第一图像直接渲染至可由该第二适配器访问的该系统存储器中。该第二适配器将该第二图像渲染至该相同的存储器中。该第二适配器显示来自于该系统存储器的图像。

[0017] 依据本发明的另一个方面,提供一种操作计算器件的方法,该计算器件包含处理器、与该处理器通信的存储器、将该处理器互连至图形处理器的外围互连总线、以及显示器接口。而该方法包含在该存储器中建立帧缓冲器;指示该图形处理器渲染视频帧;将该视频帧经由该外围互连总线传送至该帧缓冲器中;以及,将来自于该帧缓冲器的该帧经由该显示器接口呈现在显示器上。

[0018] 依据本发明的再一个方面,提供一种计算器件。该计算器件包含中央处理器、与该中央处理器通信的存储器、将该中央处理器互连至图形处理器的外围互连总线、显示器接口、储存在该存储器中的计算机可执行指令,适配该计算器件:使该图形处理器渲染视频帧;将该视频帧经由该外围互连总线传送至帧缓冲器中;以及将来自于该帧缓冲器的帧经由该显示器接口显示在显示器上。

[0019] 依据本发明的另一个方面,提供一种计算器件。该计算器件包含中央处理器;与该处理器通信的存储器;第一帧缓冲器;将该处理器互连至图形子系统的外围互连总线,该图形子系统缺少本地界定帧缓冲器至该图形子系统的存储器;以及,与该存储器通信的显示器接口。该存储器储存计算机可执行指令,该计算机可执行指令适配该图形处理器渲染视频帧,并将该视频帧经由该外围互连总线传送至该第一帧缓冲器中。

[0020] 依据本发明的另一个方面,提供一种图形子系统,包含:图形处理器、及外围互连

总线接口,用来将该图形子系统互连至主处理器,该图形子系统缺少本地界定帧缓冲器至该图形子系统的存储器;以及,该图形子系统操作以渲染视频帧,并将该视频帧经由该外围互连总线传送至帧缓冲器中。

[0021] 依据本发明的另一个方面,提供一种操作第一图形处理器以分别在第一和第二显示器上绘制第一和第二图像的方法。该第二显示器是与第二图形适配器的显示器接口互连。而该方法包含渲染与该第一图形处理器通信的存储器中的该第一和第二图像;将该第二图像传送至该第二图形适配器的帧缓冲器;以及,编程该第二图形适配器的该显示器接口,以将来自于该第二图形适配器的该帧缓冲器的该第二图像显示在该第二显示器上。

[0022] 本领域中具有通常技术者,在看过接下来的本发明特定实施例的描述与伴随图式后,本发明的其它方面和特征将变得明显。

附图说明

[0023] 在以下仅通过范例来显示本发明的实施例的图式,其中,

[0024] 图 1 为计算器件的简化方框图,其为本发明的实施例的范例;

[0025] 图 2 为图 1 的该计算器件的软件的简化逻辑方框图;

[0026] 图 3 为图 1 的该计算器件的一部分的简化方框图;

[0027] 图 4 为图 1 的该计算器件的一部分的进一步示意方框图;

[0028] 图 5A 为显示在图 1 的该计算器件所实施的步骤的流程图,其为本发明的实施例的范例;

[0029] 图 5B 示意地描绘图 5A 的步骤在图 1 的器件的组件上的作用;

[0030] 图 6A 为显示在图 1 的计算器件所实施的步骤的流程图,其为本发明的实施例的范例;

[0031] 图 6B 示意地描绘图 6A 的步骤在图 1 的器件的组件上的作用;

[0032] 图 7 和图 8 为显示在图 1 的计算器件所实施的步骤的流程图,其为本发明的进一步实施例的范例;

[0033] 图 9 为计算器件的一部分的进一步部分简化示意方框图,其为本发明的进一步实施例的范例;

[0034] 图 10 显示在图 9 的计算器件所实施的步骤的流程图,其为本发明的实施例的范例;以及

[0035] 图 11A、图 11B 为显示图 9 的器件的操作的简化方框图;以及

[0036] 图 12 为计算器件的一部分的进一步部分简化示意方框图,其为本发明的进一步实施例的范例。

具体实施方式

[0037] 图 1 为计算器件的简化示意方框图,其为本发明的实施例的范例。计算器件 10 是基于传统的 Intel x86 架构。然而,将会变得明显,本发明可容易地实作在具有 PowerPC、AMD x86 或其它架构的的计算器件中。

[0038] 计算器件 10 包含处理器 12,通过集成接口电路 14 和 20 互连至系统存储器 16 和外围。

[0039] 在范例计算器件 10 中的处理器 12 为传统的中央处理单元,并且可例如为与该 INTEL™x86 家族微处理器兼容的微处理器。

[0040] 集成接口电路 14 和 20 通常分别指北桥和南桥,且提供处理器 12 至外围和存储器的接口。如所显示的,北桥 14 互连该处理器 12;存储器 16;经由高速互连总线 22 的复数个扩充插槽 18;以及南桥 20。南桥 20 互连至低速外围和互连,例如经由集成 IDE/SATA 24 的磁盘驱动器、经由集成通用串行总线 (USB) 端口 26 的通用串行总线器件;以及,经由低速互连总线 28 的例如符合 PCI 或 ISA 标准的其它外围。南桥 20 可包含集成音讯组件 30、网络接口 (未显示)、等等。

[0041] 范例的互连总线 22 为 PCI express (PCIe) 总线,其带宽在每秒百万字节的范围内,并允许数据传送在此带宽读取和写入。举例来说,现有的 PCIe 总线速度允许双向数据在高达 266MHz 或 533MHz 的总线速度下传送,以允许高达每秒 2.1 百万字节或每秒 4.2 百万字节的数据速率。

[0042] 在绘示的实施例中,北桥 14 进一步包含集成图形适配器 34,集成图形适配器 34 适于将计算器件 10 互连至第一显示器 32,第一显示器 32 的形式为监视器、LCD 面板、电视、等等。

[0043] 如以下所详述的,经由互连总线 22 互连且形成例如互连总线 22 上的扩充插槽 18 中的外围扩充卡 50 的一部分的额外图形适配器 52 可进一步为计算器件 10 的一部分。

[0044] 在绘示的实施例中,计算器件 10 执行储存在存储器 16 中的软件。如图 2 所显示的,范例软件 100 包含储存在存储器 16 中的操作系统 102、图形库 104 和应用软件 106。范例的操作系统包含 Windows Vista、Windows XP;Windows NT4.0、Windows ME;Windows 98、Windows 2000、Windows 95、或 Linux 操作系统。范例的图形库包含 Microsoft DirectX libraries 和 OpenGL libraries。计算机存储器 16 (图 1) 和互连的磁盘驱动器 (未显示) 包含随机访问存储器、只读存储器和磁盘储存存储器的适合组合,其被器件 10 用来以本发明的实施例的范例方式储存和执行和器件 10 适配的操作系统和图形适配器驱动器程序。范例的软件 100 (图 2) 例如可储存在只读存储器中或从外在的外围 (例如与 IDE/SATA 端口 24 通信的磁盘驱动器) 加载。计算机可读介质 27 可为光学储存介质、磁性软盘 (magnetic diskette)、磁带、ROM 盒带 (cartridge)、等等。

[0045] 图形库 104 或操作系统 102 进一步包含图形驱动器软件 108,用来与图形适配器 34 和 52 低阶通信。该软件具有数层,较高阶层使用较低阶层以提供特定功能。应用可利用操作系统 102 和图形库 104 以渲染 2D 或 3D 图形。渲染从上下文看是包含绘制、不然就是创造用于呈现的图形图像,且可例如包含多边形渲染 (polygon rendering)、光线追踪 (ray-tracing)、图像译码 (image decoding)、线绘制 (line drawing)、等等。将会变得明显,本发明的实施例的软件范例可形成图形库 104 和 / 或驱动器软件 108 的一部分。在范例实施例中,范例软件可形成 OpenGL library 和相关的驱动器的一部分。OpenGL 在 The Official Guide to Learning OpenGL,第 1.1 版有特别的详述,该内容并入于此作为参考。然而,具有通常技术者将了解本发明的实施例的范例方法可形成 Microsoft Direct3D libraries、不使用惯用库或惯用程序的应用 (例如游戏)、或其它任何 3D 应用或库的一部分。

[0046] 描绘集成图形适配器 34 的计算器件 10 的一部分 (和尤其是北桥 14) 的进一步简

化方框图是显示在图 3。如所显示的,北桥 14 包含高速总线接口 36 和存储器控制器 38,将桥 14 互连至互连总线 22 和系统存储器 16。北桥 14 进一步包含集成图形适配器 34,集成图形适配器 34 接着包含图形处理器 40、命令处理器 42 和显示器接口 44。

[0047] 图形处理器 40 可采用三维 (3D) 图形处理器的形式,该三维图形处理器可渲染配置在系统存储器 16 内的帧缓冲器 46 中的三维图像,并可包含用于加速渲染图形的多重管线 (pipeline) 和管线阶段。

[0048] 在正常单一监视操作中,命令是以传统的方式由命令处理器 42 传给图形处理器 40。举例来说,命令处理器 42 可包含用来界定存储器 16 中的命令队列 (command queue) 47 的寄存器 47。应用软件 106 或操作系统 102 使用库 104 来渲染图形图像。驱动器软件 108 产生图形处理器特定的命令,并将这些放置在该命令队列 47 中,且同时更新命令处理器 42 的寄存器,以反映该队列的尾端。图形处理器 42 以管线的方式异步处理队列的命令,以为了渲染帧缓冲器 46 内的图像。当图形处理器 40 能够处理该队列中的下一个命令时,他提供讯号至命令处理器 42,命令处理器 42 接着抓取该队列 47 中的下一个命令,并将队列指示字前进至命令处理器 42 的寄存器中。

[0049] 显示器接口 44 经由存储器控制器 38 采样该帧缓冲器 46 并将图像呈现在具有 VGA 端口、复合的视频端口、DVI 端口、HDMI、显示器端口、等等形式的一个或多个视频输出端口 48 上,用以在例如具有电视、监视器、等等形式的视频器件 (例如图 1 的显示器 32) 显示一个或多个图像。以这种方式,可呈现帧缓冲器 46 中由图形处理器 40 所渲染的图像。

[0050] 将会了解到,显示器接口 44 可为任何用来转换缓冲器中用以在显示器器件上显示的数据的适合接口。举例来说,显示器接口 44 可采用 RAMDAC 的形式。显示器接口 44 通常是可编程的,例如通过复数个寄存器,以允许驱动器软件 108 或类似的软件或图形处理器 40 设定存储器 16 中的开始地址,以在其显示器输出呈现。同样地,接口 44 图素深度 (例如,每图素有几个位) 和屏幕分辨率通常也是可编程的。

[0051] 将会了解到,可在存储器 16 中放置两个帧缓冲器,以取代放置单一帧缓冲器 46。该缓冲器可交替地使用以更平顺地呈现渲染过的图像。显示器接口 44 在任何时候仅显示该两个缓冲器中的一个缓冲器的内容。被显示的缓冲器通常称为前缓冲器。图形处理器 40 渲染图像,以显示至该所谓的后缓冲器,亦即没有被显示器接口 44 正在显示的帧缓冲器。在渲染帧完成后,前和后缓冲器对调:显示器接口 44 被编程以呈现该先前的后缓冲器。以这种方式,由图形处理器 40 所渲染的接续帧间的转换可看起来较平顺。

[0052] 值得注意的是,寄存器和被适配器 34 所使用的存储器 46 的帧缓冲器均对应至全存储器空间中的地址。

[0053] 本发明的范例;然而,计算器件 10 进一步包含形成在外围扩充卡 50 上的图形适配器 52,外围扩充卡 50 可经由高速互连总线 22 上的扩充插槽插置于器件 10 中,以与北桥 14 通信,如图 1 所显示的。图形适配器 52 当互连至北桥 14 时,是示意地显示于图 4 中。如所显示的,图形适配器 52 包含图形处理器 54、视频存储器 56、命令处理器 58、存储器控制器 60、及显示器接口 62。一个或多个端口 64、66 允许适配器 52 互连至一个或多个显示器器件,例如显示器 68 (图 1)。

[0054] 命令处理器 58 和显示器接口 62 分别与命令处理器 42 和显示器接口 44 以相同的方式运作。图形处理器 54 与图形处理器 40 以相同的方式操作,以其正常的操作模式。图

形适配器 52 包含其自己的存储器 56 及存储器控制器 60, 以提供至存储器 56 和系统存储器 16 的访问。本地存储器 56 是使用来保留一个或多个用于由图形处理器 54 所使用的图形适配器 52 的缓冲器 70, 缓冲器 70 可作为帧缓冲器。用于图形处理器 54 的命令队列 57 也可建立在系统存储器 16 中。

[0055] 再次地, 本地存储器 56 可通过存储器控制器 60 寻址在全系统存储器空间中的地址。同样地, 由命令处理器 54 和显示器接口 62 所使用的寄存器也可寻址在全地址空间的地址内。如已经注意到的, 由适配器 34 所使用的寄存器和帧缓冲器 46 是对应至全存储器空间。因此, 命令处理器 58 和图形处理器 54 可写入至由显示器接口 44 和命令处理器 42 所使用的帧缓冲器 46 和寄存器。

[0056] 此外, 图形处理器 54 与存储器控制器 60 一起操作以实施块传送操作, 以将数据在系统和适配器存储器 16、56 的存储器地址之间传送。

[0057] 为了努力提供节约的集成组件, 集成图形适配器 34 提供有限的功能。举例来说, 集成图形控制器的分辨率、3D 图形能力、等等是相当的有限。因此, 相较于图形适配器 52, 集成图形适配器 34 通常具有相当少的图形处理能力。举例来说, 相较于适配器 52, 适配器 34 可更缓慢地操作, 以提供较少的特征、等等。

[0058] 现在, 集成适配器 34 或适配器 52 其自己可 (也可不) 允许经由多个端口 (例如端口 48) 互连至多个实体显示器器件。然而, 每一个适配器可互连至至少一个实体显示器。所以, 本发明的实施例的范例, 加载至存储器 16 的适配器驱动器软件 108 使图形适配器 52 作为主要 (或第一) 图形适配器, 主要图形适配器渲染图形, 以呈现在显示器 32 和 68 上, 而图形适配器 34 则作为第二或从属适配器。

[0059] 将会变得明显, 在此主要 / 从属操作模式中, 集成图形适配器 34 实质上是失能的。然而, 板上 (on-board) 图形适配器 34 的组件 (和尤其是显示器接口 44) 仍可被使用来驱动互连的监视器和其它显示器器件。像这样, 卡 50 不需包含多重显示器端口和显示器驱动器。取而代之的是, 显示器接口 44 及其端口 48 可用来驱动至少一个监视器或显示器 32, 除了显示器 68 或由卡 50 所驱动的显示器 68 外。如果适配器 52 包含多重端口 (例如端口 64、66), 处理器 52 可额外地渲染图像, 以通过这些端口来加以呈现。当然, 驱动器软件 108 也可允许适配器 34 以其传统的操作模式操作, 如上述。也可由或通过应用软件 106 来选择适配器 34 (和因此适配器 52) 的操作模式。

[0060] 具体言之, 在主要 / 从属操作模式中, 本发明的实施例的范例, 存储器 16 中驱动器软件 108 将作用的帧缓冲器 70 和次要适配器缓冲器 72 放置于适配器 52 的存储器 56 中, 如图 4 所示意地描绘。该作用的帧缓冲器 70 如用于适配器 52 的传统帧缓冲器作用着, 并因此储存帧, 以显示于适配器 52 的显示器 68 上。像这样, 图形处理器 54 以传统的方式渲染存储器 56 中的作用的帧缓冲器 70 中的二维或三维图形。卡 50 的显示器接口 62 转换作用的帧缓冲器 70 的内容, 以显示在互连的监视器或显示器器件上。选择性地, 作用的帧缓冲器 70 可被前和后缓冲器所取代, 如上述。

[0061] 图形处理器 54 进一步渲染图像, 以在次要适配器帧缓冲器 72 的显示器器件 32 上显示。总结渲染次要适配器帧缓冲器 72 中的帧, 次要适配器帧缓冲器 72 的内容被传送至帧缓冲器 46, 以由图形适配器 34 的显示器接口 44 呈现, 如下述。

[0062] 在第一实施例中, 存储器 16 中的图形适配器驱动器软件 108 实施图 5A 所描绘的

步骤 S500。具体言之,驱动器软件 108 产生命令,以在步骤 S502 中引导图形处理器 54 渲染存储器 56 的次要适配器缓冲器 72。这些命令是放置在适配器 52 的命令队列 57,并由图形处理器 54 以与命令将被队列于队列 47 内几乎相同的方式执行,如上述。一旦该帧被渲染后,驱动器软件 108 进一步产生一个命令(或数个命令),以使处理器 54 将次要适配器帧缓冲器 72 的内容位块传送(BITBLT)存储器 56 中放置如用于图形适配器 34 的帧缓冲器 46 的那个区域中(步骤 S504)。

[0063] 图形适配器 34 的显示器接口 44 接着已经预编程以将其帧缓冲器 46 的内容显示至互连的监视器或端口 48 处的显示器。方便地接下来,互连至端口 48 和端口 46 的显示器 32 和 68 上所显示的图像由图形处理器 54 渲染或绘制。步骤 S504 在缓冲器 72 和帧缓冲器 46 上的效应是显示于图 5B 中。

[0064] 在第二实施例中,实施描绘于图 6A 中的步骤 S600。起初,前和后帧缓冲器是放置在适配器 52 的存储器 16 中。为了方便起见,这些是指定为缓冲器 46a 和 46b,并且绘示在图 6B 中。在步骤 S602 中,处理器 54 再次地渲染任何帧,以由如上述参考步骤 S502 的存储器 56 的次要适配器帧缓冲器 72 中的显示器接口 44 所呈现。

[0065] 总结渲染帧以显示于器件 32 上,驱动器软件 108 编程图形处理器 54,以将存储器 56 中次要适配器帧缓冲器 72 的内容位块传送(BITBLT)至存储器 16 的该之后现在的后缓冲器,以用于步骤 S604 中的图形适配器 34。在该 BITBLT 完成后,驱动器编程处理器 54,以编程显示器接口 44 的寄存器对调适配器 34 的后缓冲器和前缓冲器(也就是,使用该后缓冲器作为该前缓冲器),在步骤 S606 中。此可藉由直接地再编程显示器接口 44 的寄存器、以确认由显示器接口 44 用来呈现数据所使用的起始地址来加以完成。图形适配器 34 的显示器接口 44 接着将其缓冲器的内容呈现至在端口 48 处的互连监视器(也就是,显示器 68)。该之前的前缓冲器接着使用作为用于适配器 34 的后缓冲器,而由处理器 54 所渲染用于显示于显示器 32 上的下一个帧将初始地被传送至步骤 S604 的此后缓冲器。当然,驱动器软件 108 维持每一个缓冲器 46a 和 46b 的起始位置作为前和后缓冲器。步骤 S604 和 S606 在缓冲器 46a 和 46b 上的效应是示意地显示在图 6B 中。

[0066] 在没有同步的情况下,使用单一缓冲器 46 或直接编程适配器 34 的显示器接口 44 可导致看得见的撕裂。也就是,显示器接口 44 的寄存器编程的垂直回扫期间(vertical blanking interval)以外的任何时间,均看得到撕裂,这是因为显示器参数在由接口 44 所输出的帧的中间会改变。

[0067] 因此,在第三实施例中,可实施图 7 中所描绘的步骤 S700。再次地,前和后缓冲器 46a、46b 是初始地放置在由适配器 34 所使用的存储器 16 中。处理器 54 再次地渲染图像,以在本地存储器 56 的缓冲器 72 内的器件 32 显示,如上述关于步骤 S502 和 S602。

[0068] 在完成渲染帧以显示在显示器 32 上后,存储器 16 中的驱动器软件 108 编程图形处理器 54,以将存储器 56 中次要适配器帧缓冲器 72 的内容位块传送至步骤 S704 的图形适配器 34 的存储器 16 中的之后现有后缓冲器。驱动器软件 108 进一步将命令置放在适配器 34 的命令队列 47 中,以对调前和后缓冲器(步骤 S706)。然而,命令处理器 42 的寄存器并非由驱动器软件 108 更新,以反映命令队列 47 中尚未处理的命令。取而代之的是,驱动器软件 108 提供命令至图形处理器 54(步骤 S708),以在步骤 S706 中所启动的位块传送完成时更新命令处理器 42 的命令队列寄存器。此接着使命令处理器 42 提供指令至处理器 40,

以将其后缓冲器对调成前缓冲器。图形处理器 40 接着执行队列的命令,以对调适配器 34 的后和前缓冲器(也就是,藉由再编程显示器接口 44 的寄存器,以将其之前后缓冲器的内容呈现至互连监视器或端口 48 处的显示器)。

[0069] 在第四实施例中,实施图 8 中所描绘的步骤 S800。再次地,前和后缓冲器初始是放置在适配器 34 的存储器 16 中。处理器 54 再次地渲染图像,以用于本地存储器 56 的缓冲器 72 中的显示器 32,如上述。

[0070] 驱动器软件 108 进一步将命令放置在适配器 34 的队列 47 中,以使图形处理器 40 将存储器 56 中次要帧缓冲器 72 的内容位块传送 (BITBLT) 至系统存储器 16 中放置作为用于图形适配器 34 的现有后帧缓冲器的那个区域。驱动器软件 108 也将命令放置在队列 47 中,以使图形处理器 40 通过编程显示器接口 44 以对调适配器 34 的前和后缓冲器,以使用该后缓冲器作为前缓冲器。然而,命令处理器 42 中界定该队列指示字的寄存器直到帧已经在缓冲器 72 中被渲染后才更新。

[0071] 所以,在总结渲染用于显示器 32 的帧,图形处理器 54 提供命令来更新界定用于命令处理器 42 的该队列指示字的命令寄存器,以反映适配器 34 的命令队列中的尚未处理的命令。图形处理器 40 接着执行该队列的命令,以将缓冲器 72 的内容位块传送至其后缓冲器,以对调其后和前缓冲器(也就是,藉由再编程显示器接口 44 的寄存器,以将其之前的后缓冲器的内容呈现至互连监视器或端口 48 处的显示器)。

[0072] 如将要了解到的,上述实施例允许主图形处理器担负在多重独立的帧缓冲器上绘制二或三维图形图像的责任。从属图形适配器可用来显示在一个帧缓冲器中所创造的图像。驱动额外的显示器所需要的器件电子组件和端口可由从属图形适配器 34 提供。以这种方式,包含在从属图形控制器中的电子组件可有效地用来允许驱动多重显示器。

[0073] 传统上,缓冲器 70 和 72 可用来界定单一表面,在该单一表面上,应用软件 106 可将单一图形图像渲染成显示跨越两个或更多个与图形适配器 52 和 34 互连的显示器 48 和 64 上。驱动器软件 108 可相应地将可用的屏幕尺寸报告给应用软件 106,相同于缓冲器 70 和缓冲器 72 的尺寸。因此,如果缓冲器 70 的分辨率为 $m_1 \times n$,而缓冲器 72 的分辨率为 $m_2 \times n$,则驱动器软件 108 可将可用的屏幕尺寸 $(m_1 + m_2) \times n$ 报告给应用软件 106 的应用。如果缓冲器 70 和 72 占据存储器 56 中相邻的 $(m_1 + m_2) \times n$ 个图素位置,则由驱动器软件 108 所实施的渲染会被简化。应用软件 106 可接着使用驱动器软件 108,以使图形处理器 40 渲染较大的图像,该较大的图像的分辨率为组成由缓冲器 70 和 72 所界定的平面的 $(m_1 + m_2) \times n$ 个图素。缓冲器 70 和 72 因此储存图像对,其中,该图像对的每个图像形成该较大的图像中由该应用所渲染的一部分。总结渲染每一个较大的图像,驱动器软件 108 可依据上述的步骤 S500、S600、S700 或 S800 将缓冲器 72 中的图像传送给帧缓冲器 46,用于显示在显示器 32 上。缓冲器 70 中的该图像可以正常的方式显示在显示器 68 上。观看放置在一起的显示器 32 和 68 的最后用户认识到由应用 106 所渲染的该较大的图像。将会了解到,具有游戏形成的应用 106 可利用多重显示器,而不需要修正。

[0074] 如果前和后缓冲器为显示器 68 所使用,则可放置和使用两个不同的缓冲器,以代替缓冲器 72,以确保保留将被显示在显示器 32 上的该表面部分的该缓冲器在存储器 56 中是邻接该缓冲器,其中,将被显示在显示器 68 上的该表面部分会被渲染。步骤 S602、S604、S702、S704 或 S802、S704 可相应地被修正为将保留该刚完成的图像的该缓冲器位块传送至

后缓冲器 46a 或 46b。方便地,现有的应用软件 106 不需修正。取而代之的是,该应用软件可只是依赖该报告的较大的屏幕尺寸。驱动器软件 108 使该渲染表面跨越多重显示器。

[0075] 也将会被了解到,虽然该描绘的实施例利用集成图形控制器作为从属图形适配器,但该从属图形适配器可轻易地是经由总线(该总线具有足够的带宽,以允许将图形帧传送至该从属帧缓冲器内)而与图形适配器 52 通信的另一个图形适配器。像这样,该从属图形处理器可形成如另一个与 PCIe 兼容的扩充卡。所以,举例来说,当图形卡升级时,较旧的扩充卡可能仍被使用,以他们的能力来驱动互连的显示器。类似地,该互连总线不需要是 PCIe 互连总线,但可为任何其它适合的总线,只要该总线的带宽允许数据以与显示器接口 44 所使用的刷新率的速率在帧缓冲器之间传送即可。同样地,虽然帧缓冲器 46(和前和后缓冲器 46a 和 46b)已经描述成为形成在系统存储器 16 中,他们可容易地形成如存储器中本地至适配器 34 的一部分。

[0076] 在本发明的又一个实施例中,渲染跨越总线 22 可允许由高速总线 22 所互连的图形适配器(或子系统)取代集成适配器/子系统。

[0077] 为了达到此目的,图 9 为本发明的另一个实施例的计算器件 10' 范例的一部分的简化方框图范例。计算器件 10' 详述于美国专利申请案第 11/421,005 号。

[0078] 简言之,然而,器件 10' 包含两个图形子系统 30' 和 40'。图形子系统 30' 包含图形引擎/处理器 32'、存储器控制器 72'、显示器接口 74' 和总线接口 78'。第二图形子系统 40' 是经由高速总线 22' (例如 PCIe 总线)而与图形子系统 30' 通信。图形子系统 40' 包含其自己的图形引擎/处理器 42';存储器控制器 52';显示器接口 54'。图形子系统 40' 进一步与图形存储器 50' 通信。将会变得明显,子系统 40' 是适配来将图形渲染跨越总线 22' 而至存储器 14'。

[0079] 器件 10' 可方便地形成如具有膝上型或较小计算器件的形式的便携式计算器件。像这样,单一外罩可包含 DC 电源、显示器 26' 和上述的主机板和组件。该第二图形子系统 40' 可附加至罩住该计算器件的其余部分的单一外罩,或可形成系泊部位(docking station)中只形成部分器件 10' 的一部分(当器件 10' 实体互连至该部分时)。

[0080] 器件 10' 可操作在至少两种电源消耗模式:较高电源消耗模式和较低电源消耗模式。在描绘的实施例中,当器件 10' 是由连接至 AC(主要)供应器的电源供应电源时,器件 10' 可假定为较高电源模式;当器件 10' 是由使用一个或多个电池、燃料电池、等等的 DC 电源供应电源时,器件 10' 是假定为较低电源消耗模式。或者,电源消耗模式可依据例如,用户偏好、被执行的软件应用的类型、电池电位、等等或其它选择,而由用户选择或由软件控制。

[0081] 控制器件 10' 的操作的软件的机构是揭露在美国专利申请案第 11/421,005 号,并且分享类似的软件 100,如上述。

[0082] 部分该软件控制器件 10' 的操作如同器件 10' 在高和低电源消耗状态转换。具体言之,图 10 绘示本发明的实施例的范例软件方框 S800', 软件方框 S800' 由器件 10' 的系统存储器中的软件控制的中央处理器(例如图 1 中的处理器 12-)所实施。方框 S800' 可在每一次器件 10' 进行状态改变时实施,因此,其子系统 30' 和 40' 应相应地组构。如所显示的,在方框 S802' 中,该软件判定器件 10' 是否应假定为较高电源消耗模式、或为其较低电源消耗模式。

[0083] 当器件 10' 假定 (或转换) 为其高电源消耗模式时, 执行方框 S804' -S810'。在方框 S804 中, 图形子系统 40' 是置于其全操作 (高电源消耗) 模式, 如果其尚未在此模式。此可通过将适当的讯号经由控制图形子系统 40' 的驱动器提供给电源控制器 60' 而加以执行。接下来, 图形子系统 40' 在方框 S806 和 S808 中致能。再次地, 此可通过将互连关联于图形子系统 30' 的任何显示器逻辑地失能 (方框 S804)、及将连接于图形子系统 40' 的该显示器逻辑地致能 (方框 S808) 而加以执行。方框 S806 和 S808 可再次地由适当的操作系统 API calls (例如, 美国专利申请案第 11/421, 005 号所描述的 EnumDisplayDevices() 和 ChangeDisplaySettingsEX() calls) 来加以执行、或透过与硬件直接通信来加以执行。

[0084] 需注意的是, 没有实体的显示器是连接至图形子系统 40'。控制图形子系统 40' 的操作的驱动器软件是组构以渲染图形子系统 30' 的缓冲器中的图像, 而非渲染步骤 S810' 中的关联存储器 50' 中的图像。方便地, 在高速总线 22' (实作为例如 PCIe 总线) 出现的情况下, 这种渲染有可能跨越总线 22', 部分是因为由该总线所致能的传送速度。

[0085] 该渲染可一个基元一个基元地跨越该总线而被引导至该帧缓冲器中, 以因此跨越总线 22' 传送该渲染的图像。此可通过将缓冲器 14' 放置在可由子系统 40' 访问的存储器、及提供缓冲器 14' 的地址给用于子系统 40' 的该驱动器软件来加以促成。或者, 将会变得明显, 该帧可在一个缓冲器中被渲染, 并由直接存储器访问 (等等) 而传送跨越总线 22'。

[0086] 并且, 用于图形子系统 30' 的驱动器进一步组构以使图形子系统 30' 的显示器接口 74' 采样存储器 14' 中的该帧缓冲器, 以将由存储器 14' 中的该帧缓冲器内的图形子系统 40' 所渲染的图像呈现在互连的显示器 26' 上。同时, 用于图形子系统 30' 的驱动器可引导图形子系统 30' 的图形引擎 32' 维持实质的休眠或闲置。此操作模式是示意地显示在图 11A 中, 其中, 仅图形子系统 40' 和图形子系统 30' 的作用的方框划有斜线。

[0087] 将为明显的, 在图 11A 的实施例, 没有使用存储器 50' 和显示器接口 54'。像这样, 这些功能方框可从子系统 40' 消除, 以允许成本降低。该结果的子系统 40' 不需包含显示器接口, 并且将缺少界定帧缓冲器本地至子系统 40' 的存储器。制造这种图形子系统可能是有利的, 因为子系统 40' 可被制造以补足由子系统 30' 所提供的功能。举例来说, 子系统可提供有提供三维图形或视频译码能力的图形引擎 42'。图形引擎 32' 可不包含这些能力。同时, 由图形引擎 32' 所提供的二维图形能力不需要包含在子系统 40' 中。消费者接着可于需要额外功能时才附加图形子系统 30'。

[0088] 当器件 10' 转换至、或假定为其低电源消耗模式时, 执行方框 S812' -S818'。广而言之, 图形子系统 40' 是部分地或完全地失能, 并且是置于其低电源消耗模式, 而渲染是再次地由图形子系统 30' 所实施。这样做后, 互连关联于图形子系统 30' 的任何显示器可在方框 S812' 中致能, 而实体地连接至图形子系统 40' 的任何显示器则可在方框 S814' 中逻辑地失能。接下来, 控制图形子系统 30' 的操作的驱动器软件再次地被组构, 以使图形子系统 30' 渲染存储器 14' 中的图像。显示器接口 74' 继续采样存储器 14', 以将图像呈现在与端口 78' 互连的显示器 26' 上。同样的, 处理器 12' 首先提供适当的讯号至电源控制器 60' (方框 S818'), 以将图形子系统 40' 置于其低电源状态。以其最简单的形式, 电源控制器 (未显示) 断开至图形子系统 40' 的电源、或将图形子系统 40' 置于较低电源睡眠模式。再次地, 在此较低电源消耗模式中, 电压被节流 (throttled), 和 / 或所有或部分

的图形子系统 40' 的电源下降和 / 或由图形子系统 40' 所使用的选择时钟减慢。具体言之,图形子系统 40' 的图形引擎 42' 维持闲置或实质地闲置(例如,其可被减慢、失能或降低电源)。此操作模式是示意地绘示在图 9B 中,其中,仅适配器 40' 和图形子系统 30' 的作用的功能方框划有斜线。该不作用 / 闲置的功能方框可完全地失能、或操作在降低的电压或时钟速度。

[0089] 选择性地,一部分的图形子系统 30' 可于图形引擎 32' 没有使用时失能。在图形子系统 40' 负责渲染图像的任何时刻,此可通过将图形引擎 32' 和其它组件置于一个或多个电压岛(该电压岛可通过 GPIO 或类似电路来加以失能)来加以促成。

[0090] 其它的变化也应是明显的。举例来说,在图 11A 中所绘示的高电源模式中,图形子系统 30' 和图形子系统 40' 两者均可渲染至存储器 14' 或存储器 50'。以这种方式,这两个图形子系统 30' 和 40' 可和谐地操作,每一者渲染存储器 14' 中交替的图像或渲染存储器 14' 中每一帧的交替部分(例如,扫描线)。

[0091] 在其它实施例中,额外的显示器可连接至图形子系统 30' 和 40', 以允许同时使用以高电源消耗模式操作的多重显示器,如上述。以这种方式,显示器接口 54' 可使用来驱动第二显示器。在转换至较低电源消耗模式后,器件 10' 可组构为如图 11B 所绘示的操作。

[0092] 类似地,器件 10' (或 10) 可包含连接至总线 22' (或 22) 的多重额外的图形子系统,所有这些图形子系统均可作用在该高电源消耗模式,并且可通过图形子系统 30' 的显示器接口 74' 渲染图形。在转换至该较低电源消耗模式后,这些可被失能,而渲染可留给图形子系统 30' 的图形引擎 32'。

[0093] 在图 12 所绘示的另一个实施例中,计算器件 10' 可包含直接存储器访问 (DMA) 控制器 90。DMA 控制器 90 可将数据从存储器 50' 传送至存储器 14'。以这种方式,在较高电源消耗模式的器件 10' 中,图形子系统 40' 可渲染图像至存储器 50'。这些渲染的图像可接着由 DMA 控制器 90 传送至存储器 14' 中的帧缓冲器。DMA 控制器 90' 可形成部分的图形子系统 30' 或 40' (例如图形引擎 32' 或 42' 的 DMA 引擎),或为置放于计算器件 10' 中。数据可从存储器 50' 经由跨越总线 20' 或直接传送至存储器 14'。显示器接口 74' 将继续如上述的操作,以采样存储器 14' 中的帧缓冲器,以将该渲染的图像呈现在显示器 26' 上。再次地,以其较高电源消耗模式操作的图 10 的器件 10' 的作用方框在图 12 中划有斜线。

[0094] 当然,上述的实施例只是用来显示,而决不是用来限制的。实现本发明的该描述的实施例可接受许多形式的修正、部件的配置、操作的细节或顺序。而本发明意欲将所有这种修正涵盖在其范围中,如权利要求书所界定的。

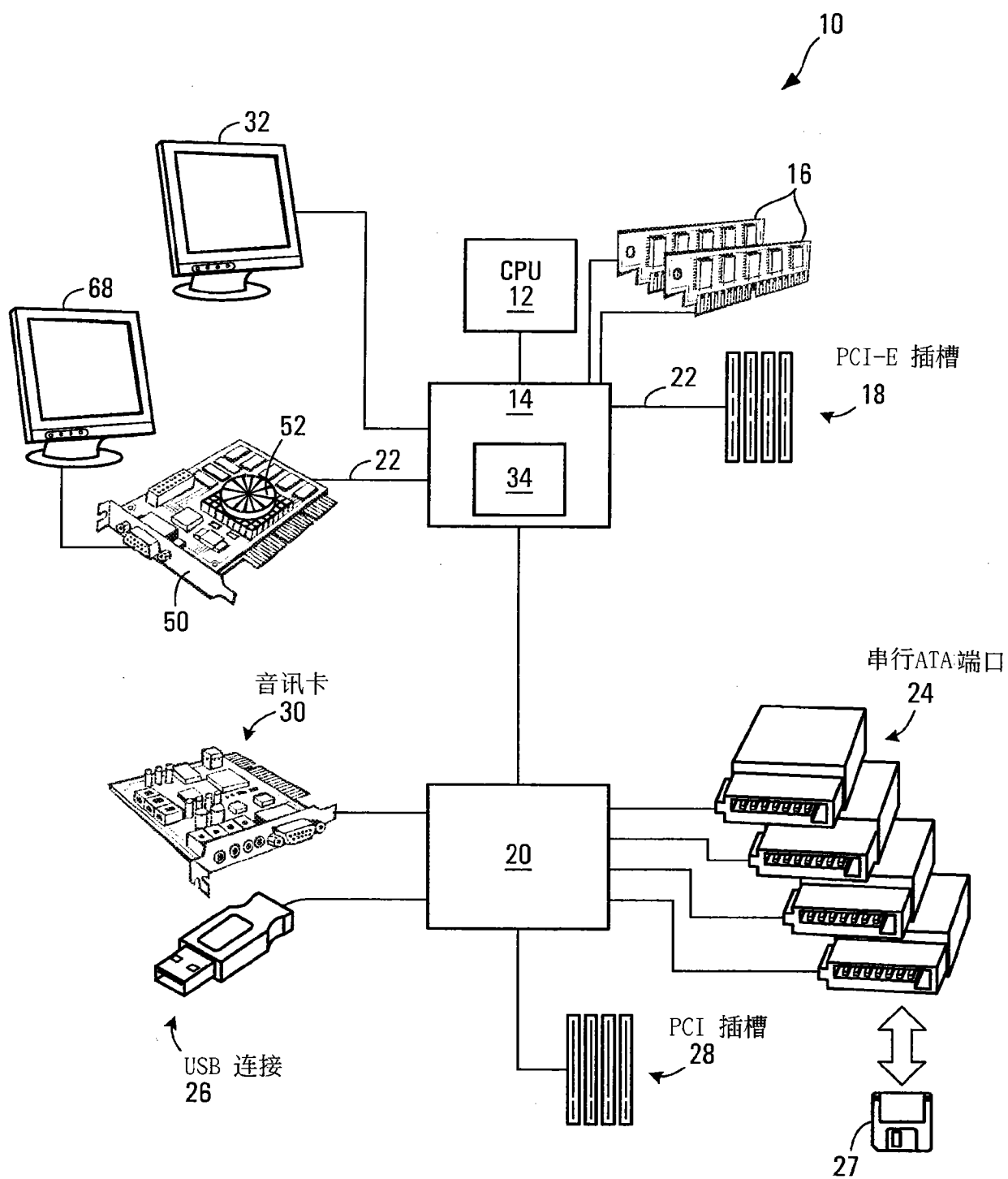


图 1

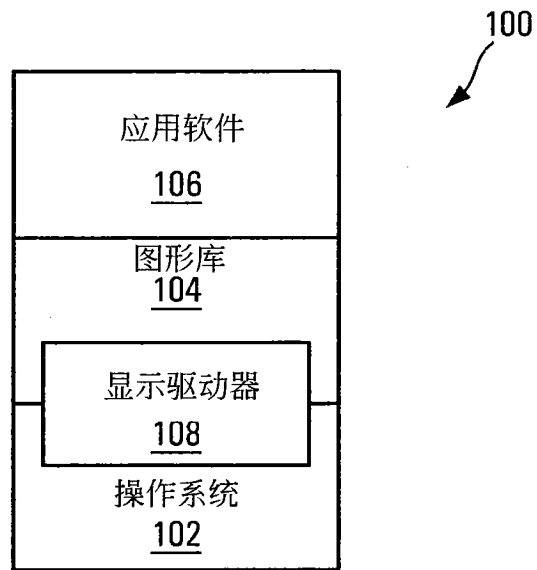


图 2

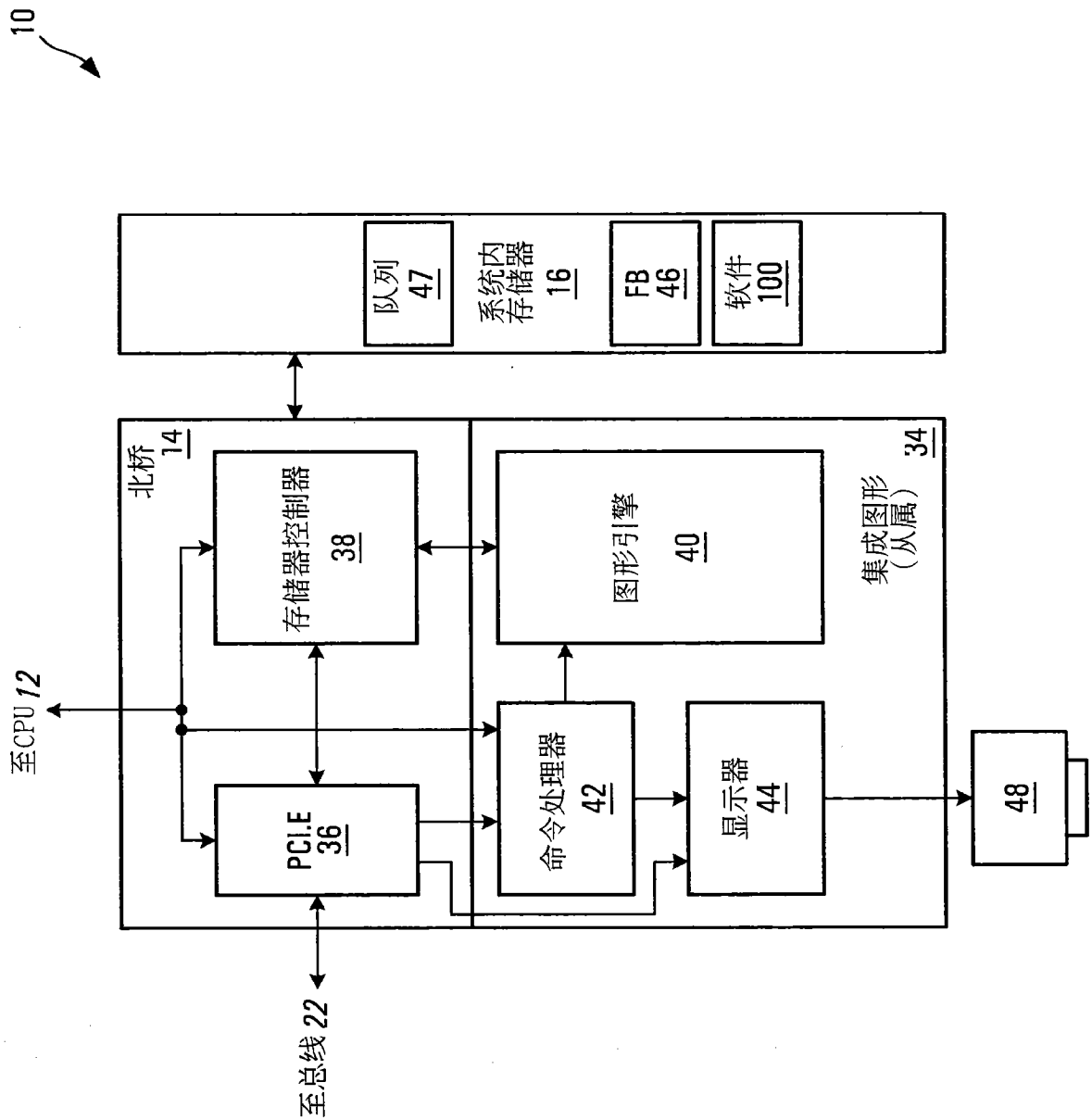


图 3

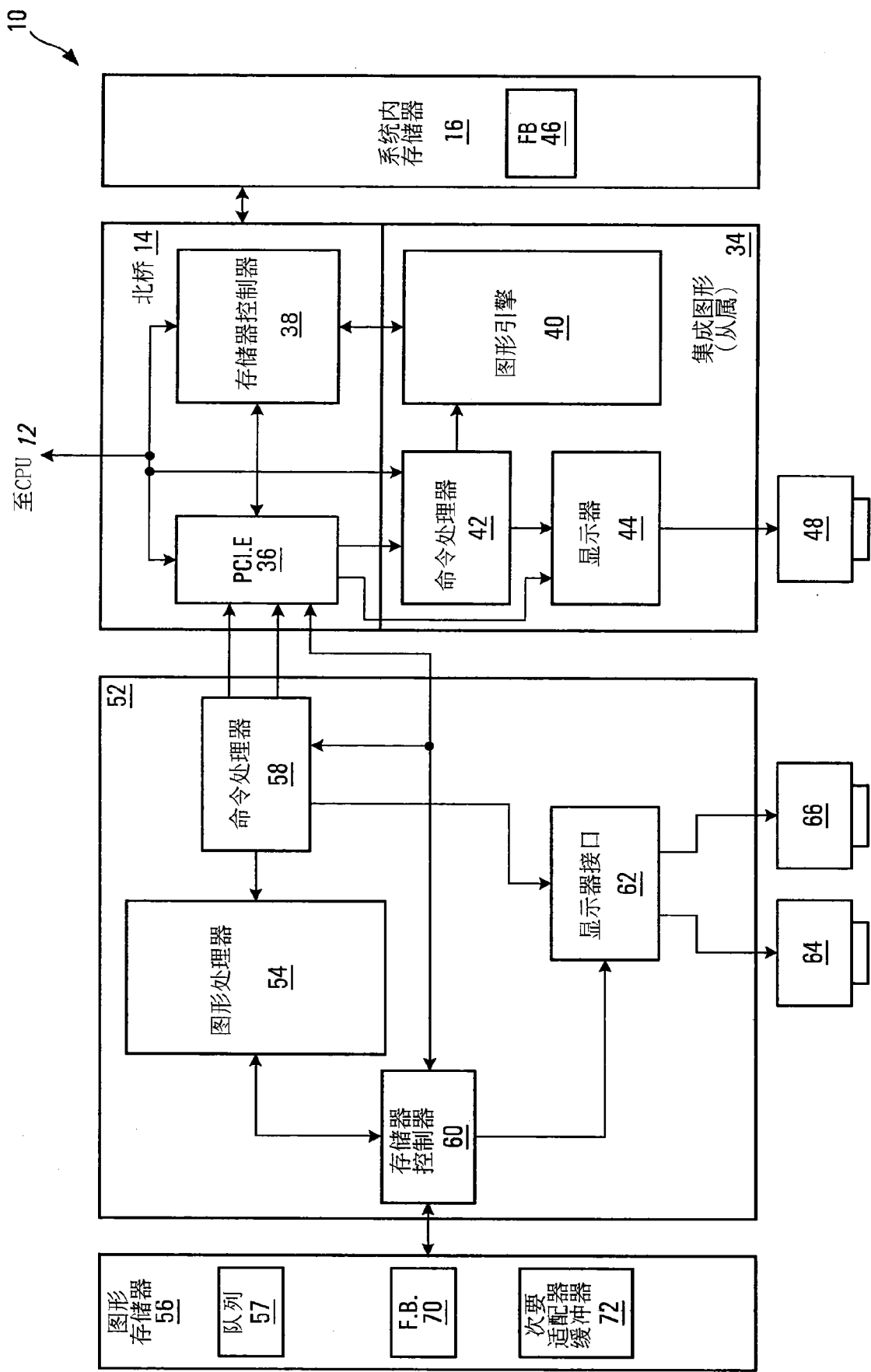


图 4

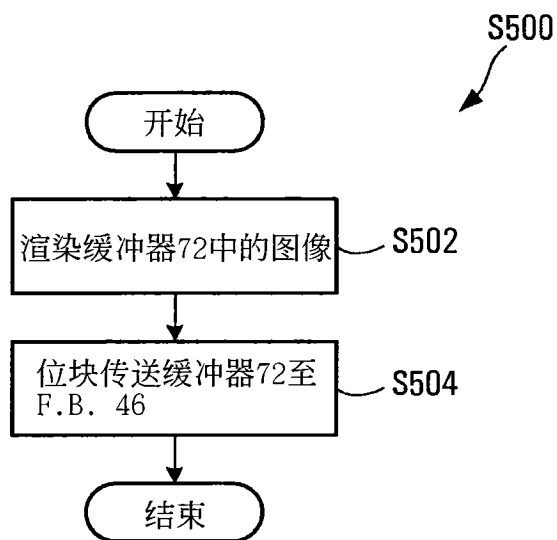


图 5A

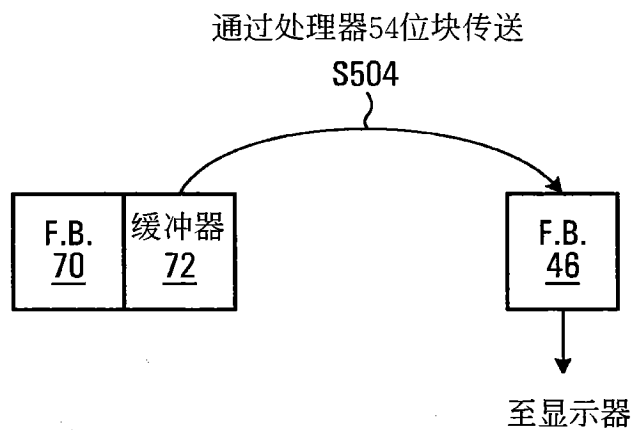


图 5B

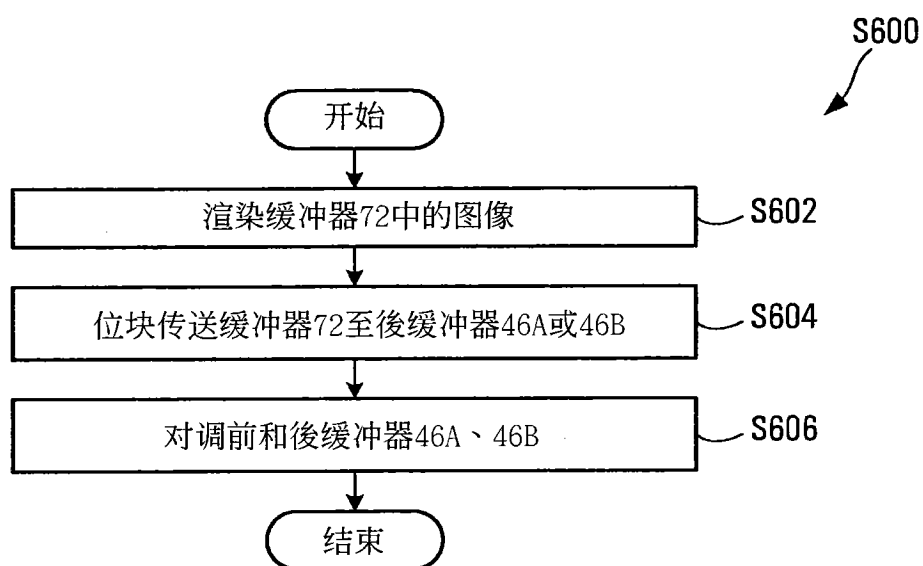


图 6A

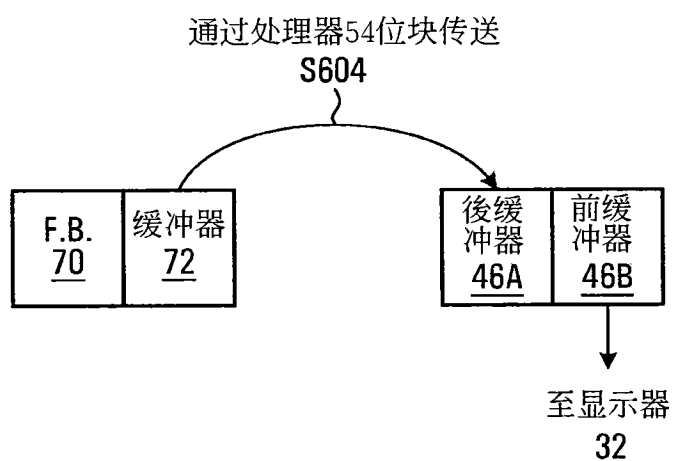


图 6B

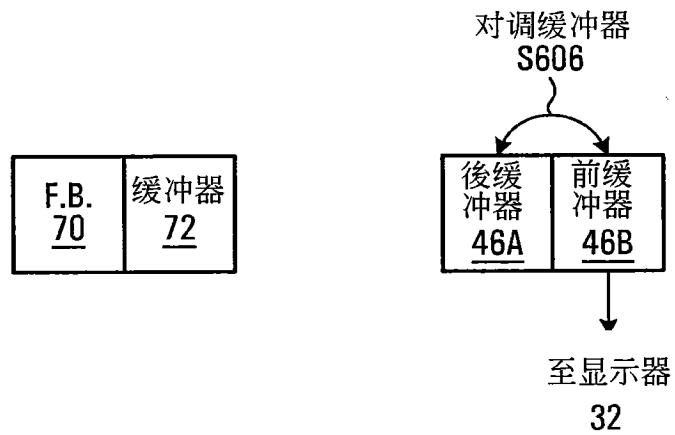


图 6C

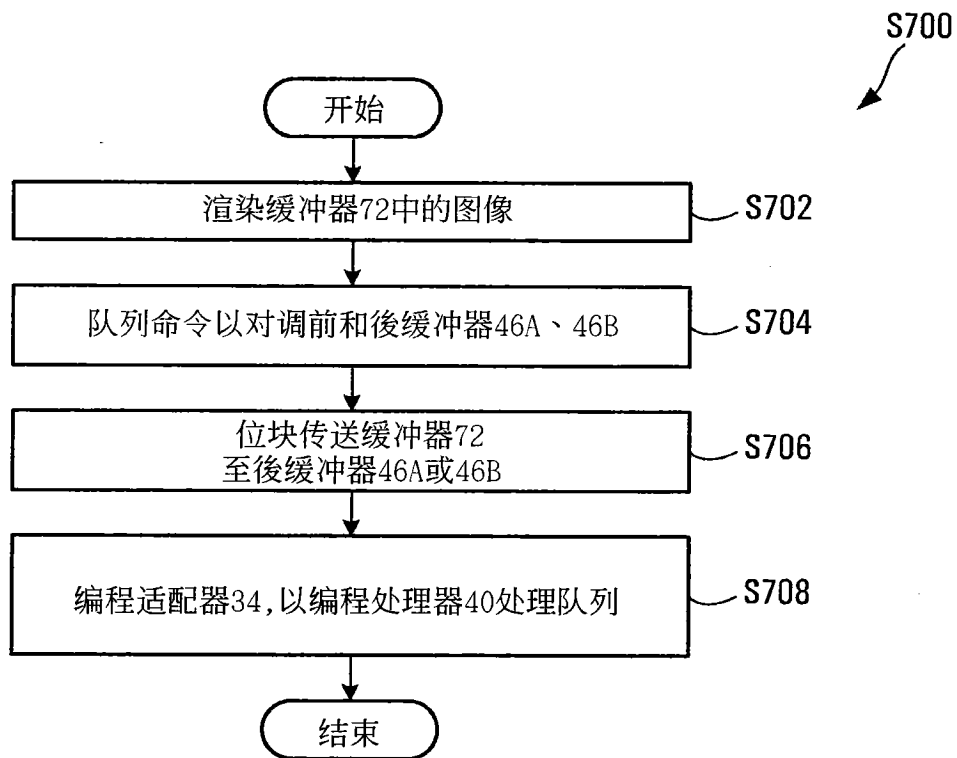


图 7

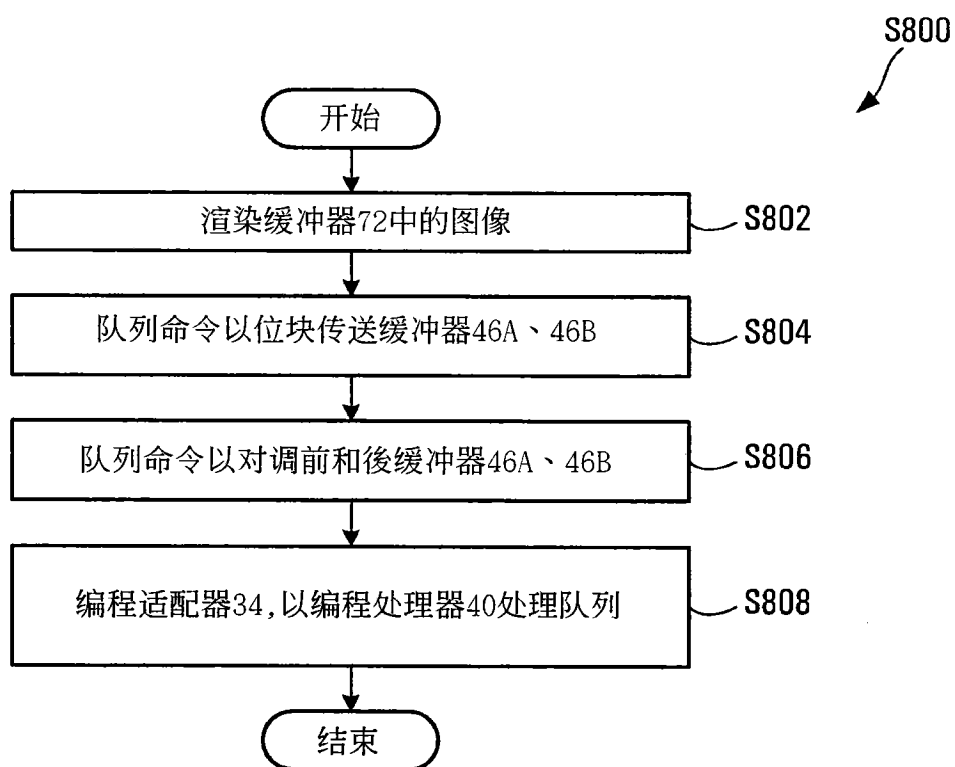


图 8

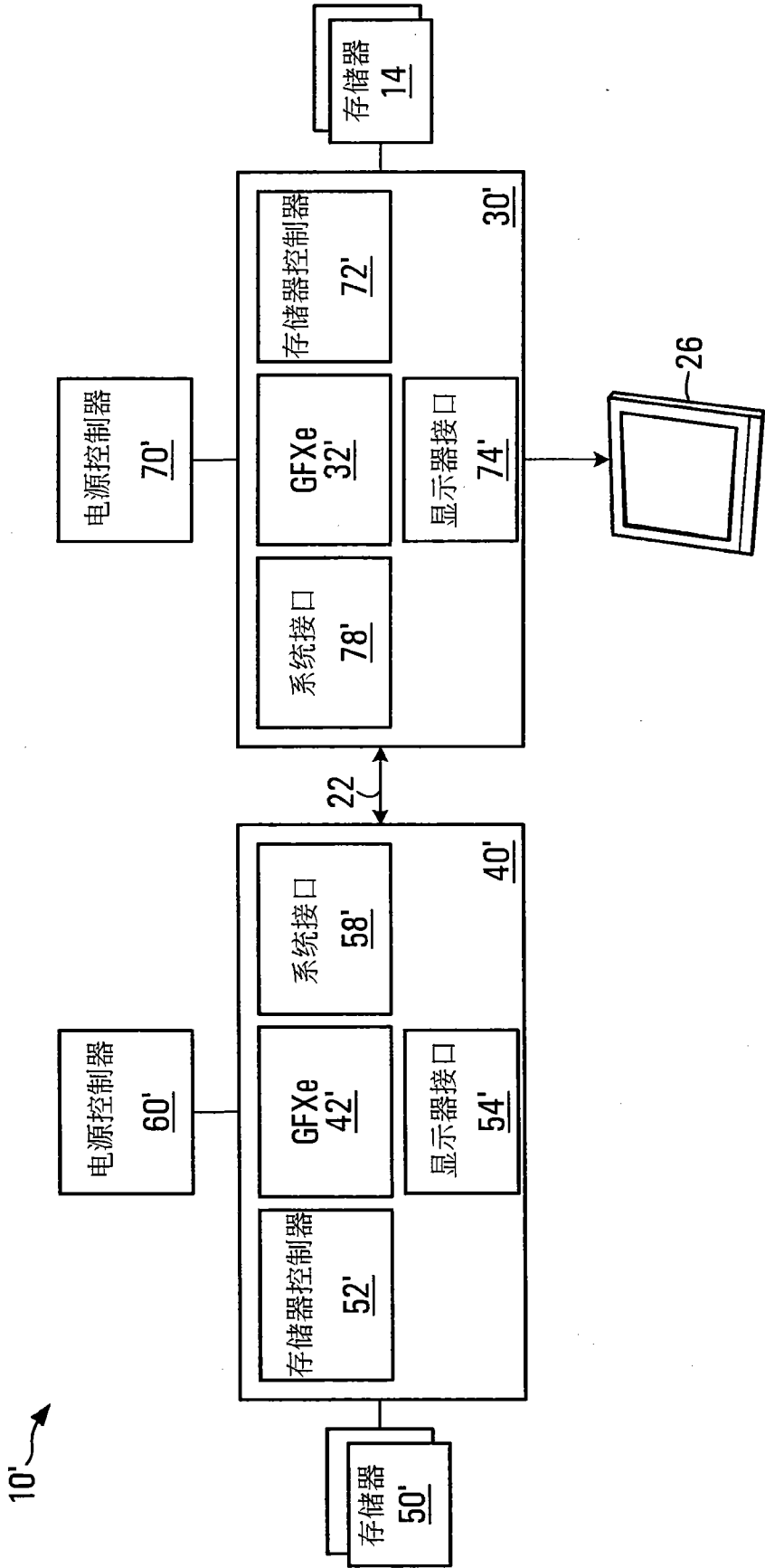


图 9

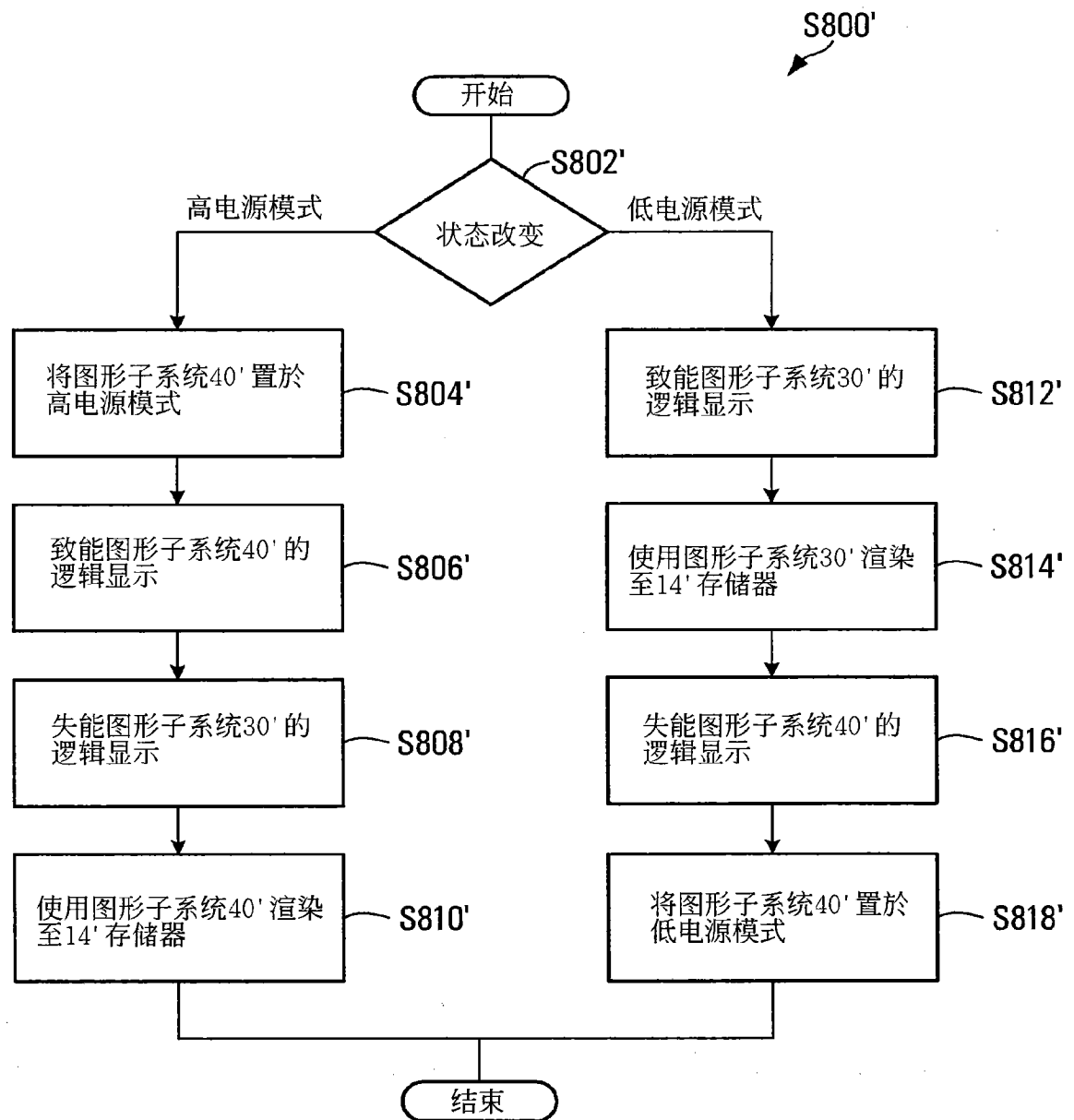


图 10

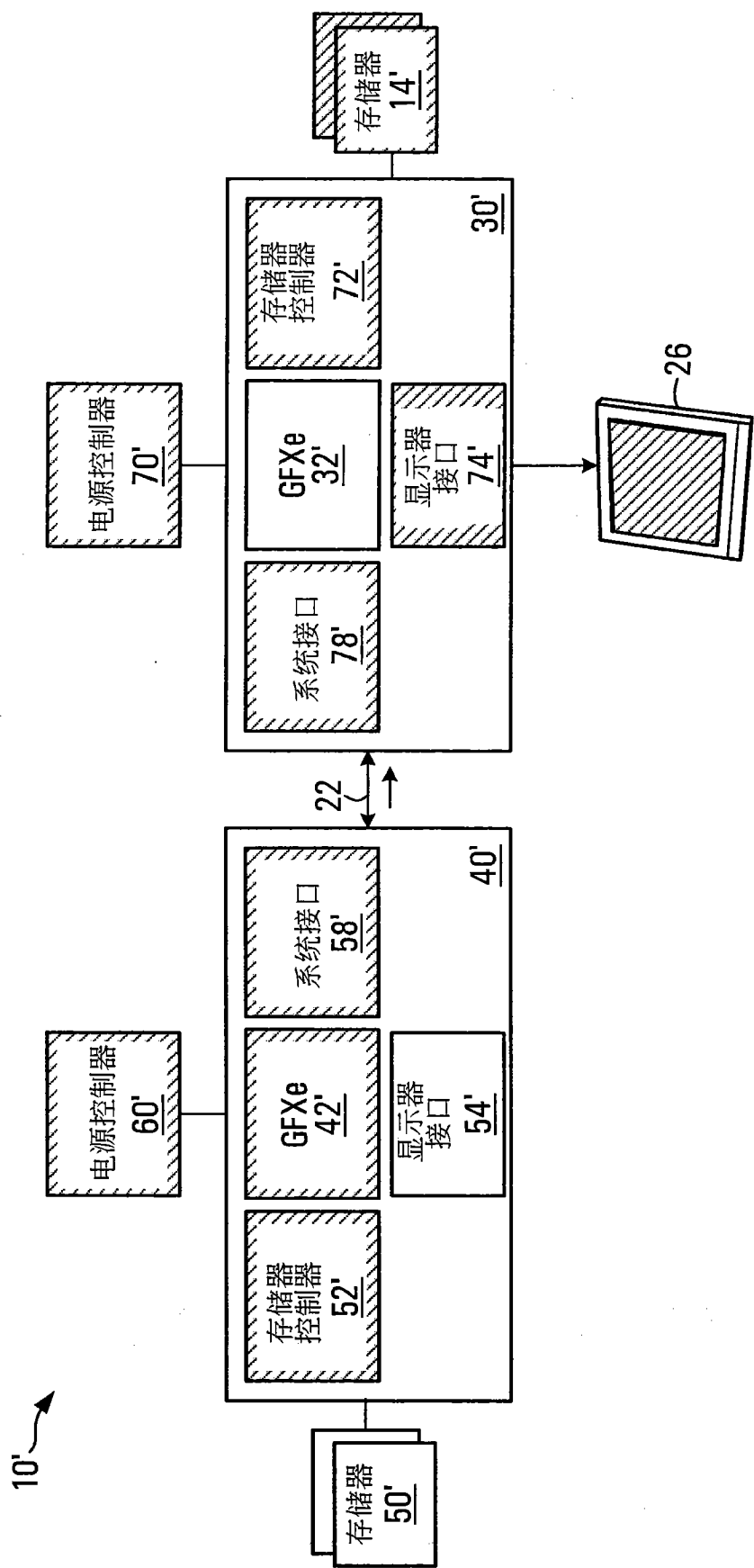


图 11A

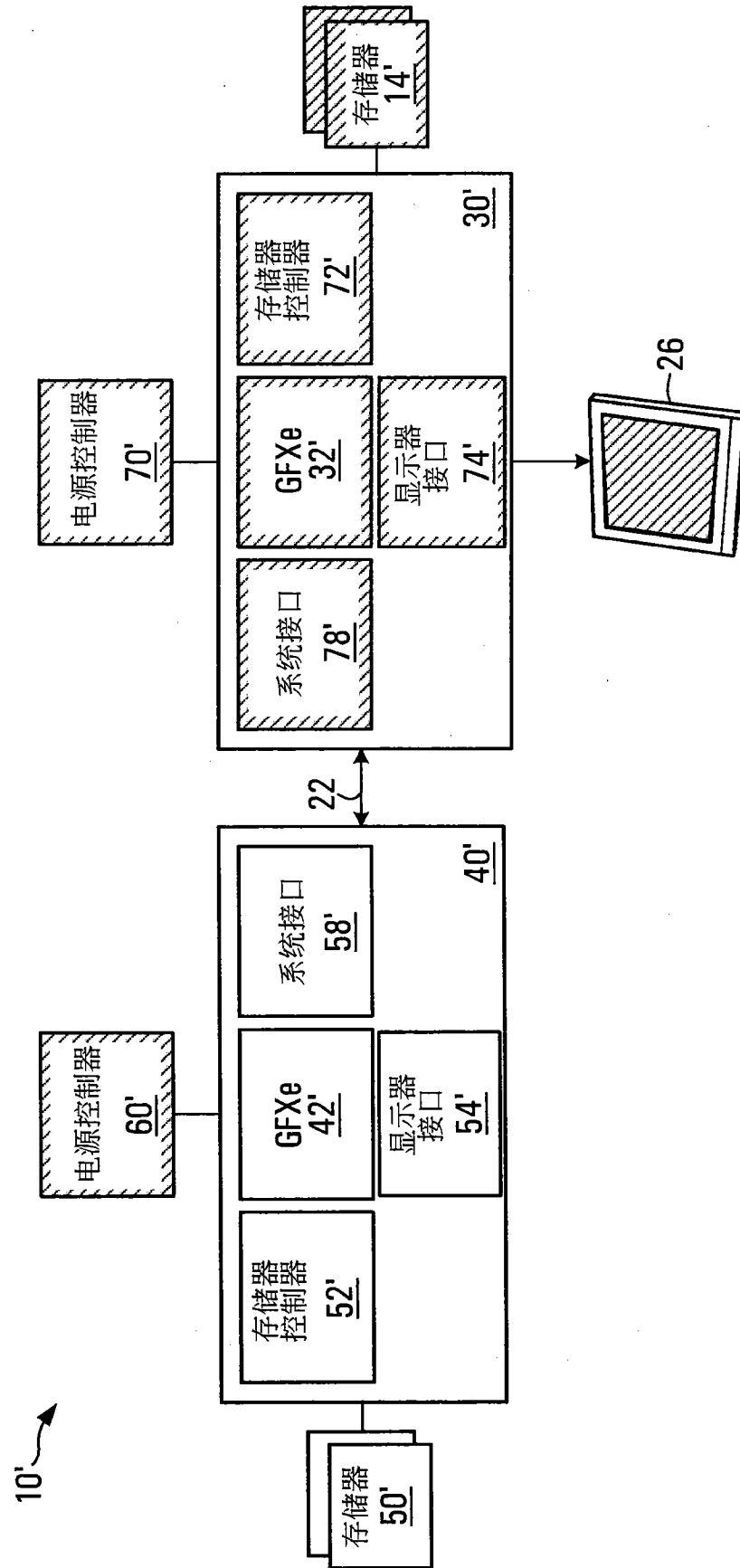


图 11B

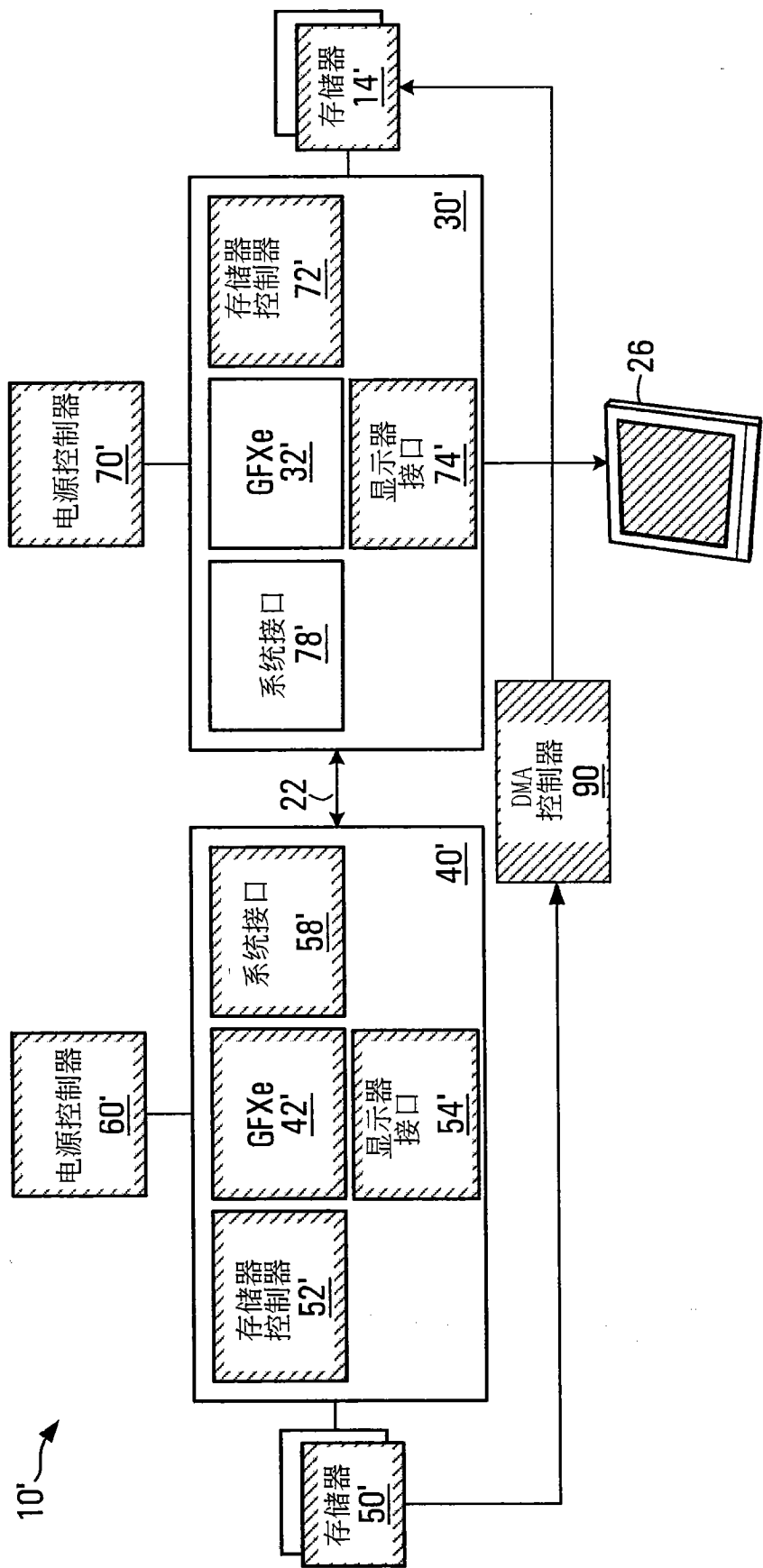


图 12