I794064

(19)中華民國智慧財產局

(12)發明說明書公告本 **(11)證書號數：TW I794064 B**

(45)公告日： 中華民國 112 (2023) 年 02 月 21 日

(21)申請案號：111111342 (22)申請日： 中華民國 107 (2018) 年 03 月 13 日

(51)Int. Cl.： ***G16B40/00 (2019.01)*** ***G06F16/00 (2019.01)***

(30)優先權：2017/04/24 美國 15/495,112

(71)申請人：美商英特爾股份有限公司 (美國) INTEL CORPORATION (US)
美國

(72)發明人：亞布 亞奇雪克 APPU, ABHISHEK (IN)；金德煥 KIM, DUKHWAN (KR)；傑哈吉達 桑吉夫 JAHAGIRDAR, SANJEEV (US)；寇克 歐塔 KOKER, ALTUG (US)；雷岡納森 華森士 RANGANATHAN, VASANTH (US)；麥佛森 麥克 MACPHERSON, MIKE (US)；凡佈 巴拉吉 VEMBU, BALAJI (US)；雷 喬伊迪普 RAY, JOYDEEP (US)；赫德 琳達 HURD, LINDA (US)；威斯特 約翰 WEAST, JOHN C. (US)

(74)代理人：林志剛

(56)參考文獻：

US 2014/0007111A1 US 2015/0187042A1

US 2016/0321777A1

審查人員：李惟任

申請專利範圍項數：18 項 圖式數：43 共 193 頁

(54)名稱

用於在自主機器之機器學習的屏障和同步

(57)摘要

說明促進用於在自主機器之機器學習的屏障和同步的機制。如於此的說明，實施例的方法包括偵測關於與一或多個處理裝置關聯的機器學習的執行緒群。該方法更包括跨多個晶粒促進該執行緒群的屏障同步，使得在執行緒群中的各個執行緒跨與該多個晶粒關聯的成組的計算元件而被排程，其中各個晶粒代表該一或多個處理裝置其中的處理裝置，該處理裝置包括圖形處理器。

A mechanism is described for facilitating barriers and synchronization for machine learning at autonomous machines. A method of embodiments, as described herein, includes detecting thread groups relating to machine learning associated with one or more processing devices. The method may further include facilitating barrier synchronization of the thread groups across multiple dies such that each thread in a thread group is scheduled across a set of compute elements associated with the multiple dies, where each die represents a processing device of the one or more processing devices, the processing device including a graphics processor.

指定代表圖：

符號簡單說明：

600:計算裝置

604:輸入/輸出來源

606:作業系統

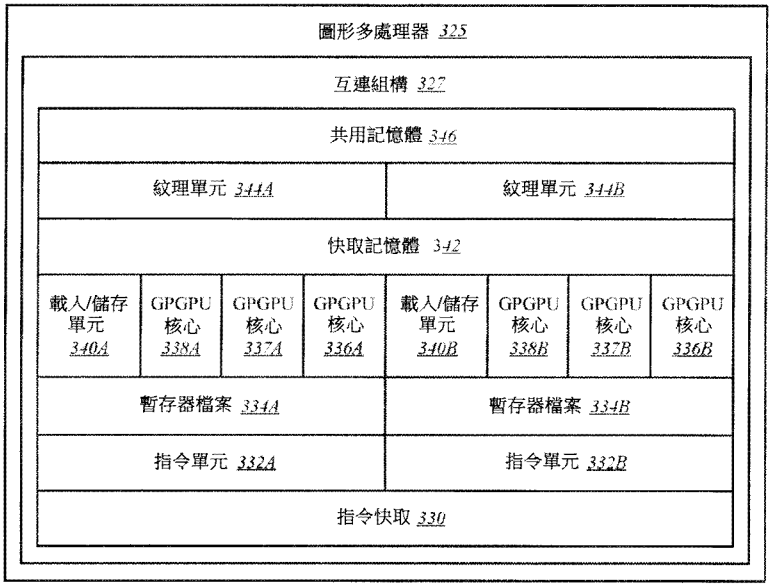
608:記憶體

610:屏障和同步機制

612:應用處理器

614:圖形處理器

616:驅動器



【圖 3A】

專利案號：111111342

公告本

I794064

【發明摘要】

【中文發明名稱】

用於在自主機器之機器學習的屏障和同步

【英文發明名稱】

BARRIERS AND SYNCHRONIZATION FOR MACHINE LEARNING AT AUTONOMOUS MACHINES

【中文】

說明促進用於在自主機器之機器學習的屏障和同步的機制。如於此的說明，實施例的方法包括偵測關於與一或多個處理裝置關聯的機器學習的執行緒群。該方法更包括跨多個晶粒促進該執行緒群的屏障同步，使得在執行緒群中的各個執行緒跨與該多個晶粒關聯的成組的計算元件而被排程，其中各個晶粒代表該一或多個處理裝置其中的處理裝置，該處理裝置包括圖形處理器。

【英文】

A mechanism is described for facilitating barriers and synchronization for machine learning at autonomous machines. A method of embodiments, as described herein, includes detecting thread groups relating to machine learning associated with one or more processing devices. The method may further include facilitating barrier synchronization of the thread groups across multiple dies such that each thread in a thread group is scheduled across a set of compute elements associated with the multiple dies, where each die represents a processing device of the one or more processing devices, the processing device including a graphics processor.

【指定代表圖】圖6

【代表圖之符號簡單說明】

600:計算裝置

604:輸入/輸出來源

606:作業系統

608:記憶體

610:屏障和同步機制

612:應用處理器

614:圖形處理器

616:驅動器

【特徵化學式】無

【發明說明書】

【中文發明名稱】

用於在自主機器之機器學習的屏障和同步

【英文發明名稱】

BARRIERS AND SYNCHRONIZATION FOR MACHINE LEARNING AT AUTONOMOUS MACHINES

【技術領域】

【0001】於此說明的實施例一般係關於資料處理，且更特別的是關於促進用於在自主機器之機器學習的屏障和同步。

【先前技術】

【0002】目前平行圖形資料處理包括發展來在圖形資料上進行特定運算的系統和方法，例如像是線形內插(linear interpolation)、曲面細分(tessellation)、光柵化(rasterization)、紋理映射(texture mapping)、深度測試(depth testing)等。傳統上來說，圖形處理器曾使用固定功能計算單元來處理圖形資料；然而，近年來，圖形處理器的部分已被做成可程式、使這類處理器能支援用於處理頂點和片段資料之更加廣泛的運算。

【0003】為了進一步增加效能，圖形處理器典型地實施處理技術，像是嘗試平行處理遍及圖形管線之不同部分

盡可能多的圖形資料。利用單指令多執行緒(SIMT；single instruction, multiple thread)架構的平行圖形處理器被設計來將在圖形管線中平行處理量最大化。在SIMT架構中，成群的平行執行緒嘗試盡可能時常地同步一起執行程式指令以增加處理效率。用於SIMT架構之軟體及硬體的總體概述能在Shane Cook著作的*CUDA Programming*，第3章第37~51頁(2013年)及/或Nicholas Wilt著作的*Handbook, A Comprehensive Guide to GPU Programming*，第2.6.2到3.1.2節(2013年6月)中找到。

【0004】機器學習已在解決許多種任務上成功。當訓練及使用機器學習演算法(例如，神經網絡(neural network))時引起的計算自然的將他們自已轉借至有效的平行實施。據此，像是通用圖形處理單元(GPGPU；general-purpose graphic processing unit)的平行處理器在深度神經網路的實際實施中扮演了重要的角色。利用單指令多執行緒(SIMT；single instruction, multiple thread)架構的平行圖形處理器被設計來將在圖形管線中平行處理量最大化。在SIMT架構中，成群的平行執行緒嘗試盡可能時常地同步一起執行程式指令以增加處理效率。由平行機器學習演算法實施所提供的效率允許高容量網路且使該些網路能在較大的資料集(dataset)上被訓練。

【0005】傳統屏障及同步技術係嚴重受限於任何數目的原因，像是全部軟體相依的、缺乏接入來行跨多個執行緒群以及錯失通用功能性等，因而，這類技術在系統資源

(像是時間、記憶體、功率及頻寬)方面是低效的。

【圖式簡單說明】

【0006】實施例係藉由範例的方式而非藉由限制的方式來闡述於附隨圖式的圖中，其中相似的參考號碼指的是類似的元件。使得能藉由參考實施例具有上述記載的特徵能以之詳細了解、更特定的描述、在上方簡要總結的方式，實施例中之一些係闡述於所附的圖式中。然而，要注意的是，所附圖式僅闡述典型的實施例，因而不會被考慮來限制其範圍，因為圖述可闡述其它等效實施例。

【0007】[圖1]為闡述組態以實施於此說明的實施例之一或多個態樣的電腦系統的方塊圖。

【0008】[圖2A~2D]依據實施例闡述平行處理器組件。

【0009】[圖3A~3B]為依據實施例的圖形多處理器的方塊圖。

【0010】[圖4A~4F]闡述在其中複數個圖形處理單元通訊地耦接至複數個多核心處理器的示範性架構。

【0011】[圖5]為依據實施例的圖形處理管線的概念圖。

【0012】[圖6]闡述依據一實施例主控屏障和同步機制的計算裝置。

【0013】[圖7]闡述依據一實施例的屏障和同步機制。

【0014】[圖8A]闡述依據一實施例用於採用及使用用於機器學習的多晶粒屏障的架構設定。

【0015】[圖8B]闡述依據一實施例用於在機器學習中促進執行緒群之同步的框架。

【0016】[圖8C]闡述依據一實施例用於在機器學習中使用表面庫促進跨圖形處理器共用的資料的框架。

【0017】[圖9A]闡述依據一實施例用於在機器學習中沒有屏障或共用局部記憶體下促進執行緒群之最佳化排程的圖形處理器。

【0018】[圖9B]闡述依據一實施例用於在機器學習中基於屏障促進執行緒群搶先的框架。

【0019】[圖10]闡述依據實施例的機器學習軟體堆疊。

【0020】[圖11]為依據實施闡述高度平行通用圖形處理單元。

【0021】[圖12]闡述依據實施例的多GPU計算系統。

【0022】[圖13A~13B]闡述示範性深度神經網路的層。

【0023】[圖14]闡述深度神經網路的訓練及部署。

【0024】[圖15]闡述深度神經網路的訓練及部署。

【0025】[圖16]為闡述分散式學習的方塊圖。

【0026】[圖17]闡述適於使用訓練的模型進行推論的示範性推理晶片上系統(SOC)。

【0027】[圖18]為有著具有一或多個處理器核心及圖

形處理器的電腦系統之實施例的方塊圖。

【0028】[圖19]為具有一或多個處理器核心、積體記憶體控制器以及積體圖形處理器之一實施例的方塊圖。

【0029】[圖20]為可為獨立圖形處理單元的圖形處理器或為與複數個處理核心整合的圖形處理器之一實施例的方塊圖。

【0030】[圖21]為用於圖形處理器的圖形處理引擎之實施例的方塊圖。

【0031】[圖22]為圖形處理器之另一實施例的方塊圖。

【0032】[圖23]為包括處理元件之陣列的執行緒執行邏輯之方塊圖。

【0033】[圖24]依據實施例闡述圖形處理器執行單元指令格式。

【0034】[圖25]為包括圖形管線、媒體管線、顯示引擎、執行緒執行邏輯以及算繪輸出管線之另一實施例的方塊圖。

【0035】[圖26A]為依據實施例闡述圖形處理器命令格式的方塊圖。

【0036】[圖26B]為依據實施例闡述圖形處理器命令序列的方塊圖。

【0037】[圖27]闡述依據實施例用於資料處理系統的示範性圖形軟體架構。

【0038】[圖28]為闡述依據實施例可被使用來生產用

以進行運算的積體電路之IP核心發展系統的方塊圖。

【0039】[圖29]為依據實施例闡述可使用一或多個IP核心來製造的示範性晶片上系統積體電路的方塊圖。

【0040】[圖30]為闡述晶片上系統積體電路之示範性圖形處理器的方塊圖。

【0041】[圖31]為闡述晶片上系統積體電路之額外的示範性圖形處理器的方塊圖。

【發明內容】及【實施方式】

【0042】實施例為了新穎的技術而準備以用於對於機器學習以及以在執行緒群內每執行緒同步來給予多晶粒屏障。實施例更為了使用屏障促進跨執行緒群的同步來準備，其中對於未具有共用局部記憶體(SLM；shared local memory)或屏障使用的該些執行緒群而言，他們係跨串流多處理器(SM；streaming multiprocessor)及/或跨圖形處理器而排程。實施例更為了基於屏障搶先執行緒而準備。

【0043】要注意的是，術語或縮寫，像是「卷積神經網路」、「CNN」、「神經網路」、「NN」、「深度神經網路」、「DNN」、「遞歸神經網路」、「RNN」及/或類似者可遍及本文件而可互換地參考。進一步，術語像是「自主機器」或簡單為「機器」、「自主車輛」或簡單為「車輛」、「自主代理器」或簡單為「代理器」、「自主裝置」或「計算裝置」、「機器人」及/或類似者，可遍及本文件而可互換地參考。

【0044】在一些實施例中，圖形處理單元(GPU；graphics processing unit)係通訊地耦接至主機/處理器核心以加速圖形運算、機器學習運算、圖樣分析運算以及各種通用GPU(GPGPU)功能。GPU可在匯流排或另外的互連(例如，高速互連，像是PCIe或NVLink)之上通訊地耦接至主機處理器/核心。在其它實施例中，GPU可在相同的封裝上或晶片上被整合為核心且在內部處理器匯流排/互連(亦即，內部於該封裝或晶片)之上通訊地耦接至該核心。無論GPU被連接於其中的方式為何，處理器核心可以包含在工作描述符(work descriptor)中的命令/指令之序列的形式分配工作給GPU。接著GPU使用專屬電路/邏輯以用於有效地處理這些命令/指令。

【0045】在下列發明說明中，提出了眾多的特定細節。然而，如於此的說明，實施例可不以這些特定細節來實踐。在其它實例中，周知的電路、結構及技術未被詳細地繪示以為了不去模糊本發明說明的了解。

系統概觀I

【0046】圖1為闡述組態以實施於此說明的實施例之一或多個態樣的計算系統100的方塊圖。計算系統100包括處理子系統101，其具有一或多個處理器102和經由可包括記憶體中樞105的互連路徑通訊的系統記憶體104。記憶體中樞105在晶片組組件內可為分開的組件或可被整合在一或多個處理器102內。記憶體中樞105經由通訊連結106與

I/O子系統111耦接。I/O子系統111包括I/O中樞107，其能夠致能計算系統100來從一或多個輸入裝置108接收輸入。此外，I/O中樞107能致能顯示控制器，其可被包括在一或多個處理器102內，用以提供輸出給一或多個顯示裝置110A。在一實施例中，與I/O中樞107耦接的一或多個顯示裝置110A能包括局部、內部或嵌入的顯示裝置。

【0047】在一實施例中，處理子系統101包括一或多個平行處理器112，其經由匯流排或其它通訊連結113耦接至記憶體中樞105。通訊連結113可為任何數目的基於標準的通訊連結科技或協定其中一者，像是(但不限於)PCI Express或可為供應商特定通訊介面或通訊組構(communications fabric)。在一實施例中，該一或多個平行處理器112形成計算聚焦平行或向量處理系統，其能包括大數目的處理核心及/或處理叢集，像是眾積體核心(MIC；many integrated core)處理器。在一實施例中，該一或多個平行處理器112形成圖形處理子系統，其能將像素輸出至經由I/O中樞107耦接的一或多個顯示裝置110A之其中一者。該一或多個平行處理器112亦能包括顯示控制器以及顯示介面(未繪示)，用以對一或多個顯示裝置110B致能直接連接。

【0048】在I/O子系統111內，系統儲存單元114能連接至I/O中樞107以提供用於計算系統100的儲存機制。能使用I/O切換器116以提供介面機制來致能在I/O中樞107與其它組件之間的連接，像是網路配接器118及/或可被整合

到平台中的無線網路配接器119，以及能經由一或多個外加裝置(add-in device)120添加的各種其它裝置。網路配接器118能為乙太配接器或另一種有線網路配接器。無線網路配接器119能包括Wi-Fi、藍牙、近場通訊(NFC；near field communication)或包括一或多個無線射頻的其它網路裝置之一或多者。

【0049】計算系統100能包括未明白繪示的其它組件，包括USB或其它埠連接、光學儲存裝置、視訊捕捉裝置及類似者，亦可被連接至I/O中樞107。在圖1中將各種組件互連的通訊路徑可使用任何合適的協定來實施，像是基於PCI(周邊組件互連)的協定(例如，快速PCI(PCI-Express))或任何其它匯流排或點對點通訊介面及/或協定，像是NV-Link高速互連或在本領域已知的互連協定。

【0050】在一實施例中，一或多個平行處理器112包含對於圖形和視訊處理最佳化的電路(例如包括視訊輸出電路)，並且構成圖形處理單元(GPU；graphics processing unit)。在另一實施例中，一或多個平行處理器112包含對於通用處理最佳化的電路，同時保存底層計算架構，如於此所更詳細說明的。又在另一實施例中，計算系統100的組件可與一或多個其它系統元件整合在單一積體電路上。例如，一或多個平行處理器112、記憶體中樞105、處理器102及I/O中樞107能被整合到晶片上系統(SoC；system on chip)積體電路中。或者，計算系統100的組件能被整合到單一封裝中以形成封裝中系統(SIP；system in package)組

態。在一實施例中，計算系統100的組件之至少一部分能被整合到多晶片模組(MCM；multi-chip module)中，其能與其它多晶片模組互連到模組化計算系統中。

【0051】將理解的是，於此繪示的計算系統100為例示性的且變異和修改是可能的。連接拓撲，包括橋接的數目和佈置、處理器102的數目以及平行處理器112的數目，可如所欲的進行修改。舉例而言，在一些實施例中，系統記憶體104直接連接至處理器102而非透過橋接，同時其它裝置經由記憶體中樞105和處理器102與系統記憶體104通訊。在其它替代的拓撲中，平行處理器112連接至I/O中樞107或直接連接至一或多個處理102其中一者，而非連接至記憶體中樞105。在其它實施例中，I/O中樞107和記憶體中樞105可被整合到單一晶片中。一些實施例可包括經由多個插座附接的二或多組處理器102，其能與二或多個平行處理器112的實例耦接。

【0052】於此繪示的特別組件其中一些為可選的且可不被包括在計算系統100之所有實施中。例如，可支援任何數目的外加卡或周邊，或是可消除一些組件。更進一步而言，一些架構可對於類似於在圖1中闡述的該些者的組件使用不同的術語。例如，記憶體中樞105在一些架構中可被稱為北橋(Northbridge)，同時I/O中樞107可被稱為南橋(Southbridge)。

【0053】圖2A依據實施例闡述平行處理器200。平行處理器200的各種組件可使用一或多個積體電路裝置來實

施，像是可程式處理器、特定應用積體電路(ASIC；application specific integrated circuit)或場可程式閘陣列(FPGA；field programmable gate array)。依據實施例，闡述的平行處理器200為在圖1中繪示的一或多個平行處理器112的變體。

【0054】在一實施例中，平行處理器200包括平行處理單元202。平行處理單元包括I/O單元204，其致能與其它裝置的通訊，包括平行處理單元202的其它實例。I/O單元204可直接連接至其它裝置。在一實施例中，I/O單元204經由使用中樞或交換介面(像是記憶體中樞105)與其它裝置連接。在記憶體中樞105與I/O單元204之間的連接形成通訊連結113。在平行處理單元202內，I/O單元204與主機介面206及記憶體交叉(memory crossbar)216連接，其中主機介面206接收關於進行處理運算的命令，而記憶體交叉216接收關於進行記憶體運算的命令。

【0055】當主機介面206經由I/O單元204接收命令緩衝時，主機介面206能將工作運算引導至前端208以進行該些命令。在一實施例中，前端208與排程器210耦接，其被組態以將命令或其它工作項目分布到處理叢集陣列212。在一實施例中，排程器210確保在任務被分布到處理叢集陣列212的處理叢集之前，處理叢集陣列212被適當地組態且在有效狀態中。

【0056】處理叢集陣列212能包括至多「N」個處理叢集(例如，叢集214A、叢集214B到叢集214N)。處理叢集

陣列212之各個叢集214A~214N能執行大數目的並行執行緒。排程器210能使用各種排程及/或工作分布演算法來將工作分配給處理叢集陣列212的叢集214A~214N，其可非常取決於針對程式或計算之各個類型引起的工作量(workload)。排程能由排程器210動態地操控，或在為了由處理叢集陣列212執行所組態的程式邏輯之編譯期間部分地由編譯器邏輯協助。

【0057】在一實施例中，不同的處理叢集陣列212之叢集214A~214N能被分配以用於處理不同類型的程式或用於進行不同類型的計算。

【0058】處理叢集陣列212能組態以進行各種類型的平行處理運算。在一實施例中，處理叢集陣列212係組態以進行通用平行計算運算。例如，處理叢集陣列212能包括用以執行處理任務的邏輯，該任務包括過濾視訊及/或聲頻資料、進行模型化運算(包括物理運算)以及進行資料變換。

【0059】在一實施例中，處理叢集陣列212係組態以進行通用平行圖形處理運算。在平行處理器200組態以進行圖形處理運算的實施例中，處理叢集陣列212能包括額外的邏輯以支援這類圖形處理運算之執行，包括(但不限於)用以進行紋理(texture)運算的紋理取樣邏輯以及曲面細分邏輯和其它頂點處理邏輯。此外，處理叢集陣列212能組態以執行關於著色器(shader)程式的圖形處理，像是(但不限於)頂點著色器、曲面細分著色器、幾何著色器以及

像素著色器。平行處理單元202能經由I/O單元204從系統記憶體傳輸資料以用於處理。在處理期間，傳輸的資料能被儲存到晶片上記憶體(例如，平行處理器記憶體222)一段處理期間，接著寫回到系統記憶體。

【0060】在一實施例中，當使用平行處理單元202以進行圖形處理時，排程器210能組態以將處理工作量分成約相等大小的任務，用以較佳致能分布圖形處理運算到處理叢集陣列212的多個叢集214A~214N。在一些實施例中，處理叢集陣列212的部分能組態以進行不同類型的處理。例如，第一部分可組態以進行頂點著色和拓撲產生、第二部分可組態以進行曲面細分和幾何著色以及第三部分可組態以進行像素著色或其它螢幕空間運算，用以生成用於顯示的算繪影像(rendered image)。由叢集214A~214N之一或多者生成的中間資料(intermediate data)可被儲存在緩衝器中以允許中間資料在叢集214A~214N之間傳送以用於進一步的處理。

【0061】在運算期間，處理叢集陣列212能接收要經由排程器210執行的處理任務，其從前端208接收界定處理任務的命令。對於圖形處理運算，處理任務能包括要被處理的資料之索引，例如表面(補丁(patch))資料、基元資料(primitive data)、頂點資料及/或像素資料以及界定資料要如何被處理(例如什麼程式要被執行)的狀態參數和命令。排程器210可組態以提取對應於任務的索引或可從前端208接收索引。前端208能組態以確保處理叢集陣列212在啟動

由即將來臨命令緩衝器(例如，批緩衝器(batch-buffer)、推緩衝器(push buffer)等)明定的工作量之前處理叢集陣列212被組態成有效狀態。

【0062】平行處理單元202的一或多個實例之各者能與平行處理器記憶體222耦接。平行處理器記憶體222能經由記憶體交叉216來存取，其能從處理叢集陣列212以及I/O單元204接收記憶體請求。記憶體交叉216能經由記憶體介面218存取平行處理器記憶體222。記憶體介面218能包括多個分割單元(例如，分割單元220A、分割單元220B到分割單元220N)，其各者能耦接至平行處理器記憶體222之部分(例如，記憶體單元)。在一實施中，分割單元220A~220N之數目能組態以等於記憶體單元的數目，使得第一分割單元220A具有對應的第一記憶體單元224A、第二分割單元220B具有對應的記憶體單元224B以及第N分割單元220N具有對應的第N記憶體單元224N。在其它實施例中，分割單元220A~220N的數目可能不等於記憶體裝置的數目。

【0063】在各種實施例中，記憶體單元224A~224N能包括各種類型的記憶體裝置，各種類型的記憶體裝置包括動態隨機存取記憶體(DRAM；dynamic random access memory)或圖形隨機存取記憶體，像是同步圖形隨機存取記憶體(SGRAM；synchronous graphics random access memory)，同步圖形隨機存取記憶體包括圖形雙資料率(GDDR；graphics double data rate)記憶體。在一實施例

中，記憶體單元224A~224N亦可包括3D堆疊記憶體，包括(但不限於)高頻寬記憶體(HBM；high bandwidth memory)。本領域具有通常知識者將理解，記憶體單元224A~224N之特定實施能變化，且能選自各種傳統設計其中一者。算繪目標，像是框緩衝器或紋理映射(texture map)可跨記憶體單元224A~224N而被儲存，其允許分割單元220A~220N平行地將各個算繪目標之部分寫入以有效地使用平行處理器記憶體222之可用頻寬。在一些實施例中，可執行平行處理器記憶體222之區域實例，偏好於利用系統記憶體協同局部快取記憶體的統合記憶體設計。

【0064】在一實施例中，處理叢集陣列212之叢集214A~214N其中一者能處理在平行處理器記憶體222內將被寫到記憶體單元224A~224N之任一者的資料。記憶體交叉216能被組態以將各個叢集214A~214N之輸出傳輸到任一分割單元220A~220N或到另一個叢集214A~214N，其能在輸出上進行額外的處理運算。各個叢集214A~214N能透過記憶體交叉216與記憶體介面218通訊或用以從各種外部記憶體裝置讀取或寫入至各種外部記憶體裝置。在一實施例中，記憶體交叉216具有對記憶體介面218的連接，用以與I/O單元204通訊，以及具有對平行處理器222之局部實例的連接，其致能在不同處理叢集214A~214N內的處理單元與系統記憶體通訊或與對平行處理單元202而言非局部的其它記憶體通訊。在一實施例中，記憶體交叉216能使用虛擬通道來將叢集214A~214N與分割單元220A~220N之

間的流量串流分開。

【0065】在當平行處理單元202之單一實例係闡述於平行處理器200內的同時，能包括任何數目的平行處理器202之實例。例如，能在單一外加卡上設置平行處理單元202之多個實例，或是能互連多個外加卡。不同的平行處理器202之實例能組態以互操作，即使不同的實例具有不同數目的處理核心、不同的局部平行處理器記憶體之量及/或其它組態差異。例如且在一實施例中，平行處理單元202之一些實例能包括相關於其它實例的更高精度浮點單元。包括平行處理單元202之一或多個實例或平行處理器200的系統能在各種組態及形式因子中實施，包括(但不限於)桌上型電腦、膝上型電腦或手持個人電腦、伺服器、工作站、遊戲機(game console)及/或嵌入式系統。

【0066】圖2B為依據實施例的分割單元220之方塊圖。在一實施例中，分割單元220為圖2A之分割單元220A~220N其中一者的實例。如所闡述的，分割單元220包括L2快取221、框緩衝器介面225以及ROP 226(光柵運算單元(raster operations unit))。L2快取221為讀取/寫入快取，其組態以進行自記憶體交叉216及ROP 226接收的載入及儲存運算。讀取不中和緊急寫回請求係由L2快取221輸出到框緩衝器介面225以用於處理。髒污(dirty)更新亦能經由框緩衝器介面225發送至框緩衝器以用於機會性處理(opportunistic processing)。在一實施例中，框緩衝器介面225與在平行處理器記憶體中的記憶體單元其中一者介

接，像是圖2A之記憶體單元224A~224N(例如，在平行處理器記憶體222內)。

【0067】在圖形應用中，ROP 226為進行光柵運算的處理單元，像是模板(stencil)、z測試(z test)、混合(blending)等等。ROP 226接著輸出被儲存在圖形記憶體中的經處理的圖形資料。在一些實施例中，ROP 226包括壓縮邏輯，用以將寫入記憶體的z或色彩資料壓縮且將自記憶體讀出的z或色彩資料解壓縮。在一些實施例中，ROP 226被包括在各個處理叢集內(例如，圖2A之叢集214A~214N)而不是在分割單元220內。在這類實施例中，對於像素資料之讀取及寫入請求在記憶體交叉216之上傳送，而不是像素片段資料。

【0068】經處理的圖形資料可在顯示裝置上顯示，像是圖1之一或多個顯示裝置110其中一者，其針對由處理器102進一步處理來選路(route)，或是針對在圖2A之平行處理器200內由處理實體其中一者進一步處理來選路。

【0069】圖2C為依據實施例在平行處理單元內的處理叢集214之方塊圖。在一實施例中，處理叢集為圖2A之處理叢集214A~214N其中一者的實例。處理叢集214能組態以平行地執行許多執行緒，其中術語「執行緒(thread)」指的是在特別組的輸入資料上執行的特別程式之實例。在一些實施例中，單指令多資料(SIMD；single-instruction, multiple-data)指令問題技術被使用來支援大數目的執行緒的平行執行，而不提供多個獨立的指令單元。

在其它實施例中，單指令多執行緒(SIMT；single-instruction, multiple-thread)技術被使用以支援大數目的一般同步執行緒之平行執行，其使用組態以將指令發出給在處理叢集之各者內成組的處理引擎的共用指令單元。不像其中所有處理引擎典型地執行相同指令的SIMD執行管轄，SIMT執行允許不同的執行緒透過給定執行緒程式更輕易地遵循發散執行路徑。本領域具有通常知識者將了解的是，SIMD處理管轄代表SIMT處理管轄之功能子集。

【0070】處理叢集214之運算可經由將處理任務分布給SIMT平行處理器的管線管理器232來控制。管線管理器232從圖2A之排程器210接收指令且經由圖形多處理器234及/或紋理單元236管理該些指令之執行。闡述的圖形多處理器234為SIMT平行處理器之示範性實例。然而，各種類型的不同架構之SIMT平行處理器可被包括在處理叢集214內。圖形多處理器234之一或多個實例能被包括在處理叢集214內。圖形多處理器234能處理資料且能使用資料交叉240以將處理的資料分布到多個可能的目的地其中一者，包括其它著色器單元。管線管理器232能藉由針對用以經由資料交叉240分布的處理資料明定目的地來促進處理資料之分布。

【0071】在處理叢集214內的各個圖形多處理器234能包括相同組的功能執行邏輯(例如，算術邏輯單元、載入儲存單元等)。功能執行邏輯能以管線的方式組態，在其中在先前指令完成之前能發出新的指令。可提供功能執行

單元。功能邏輯支援各種運算，整數及浮點算術比較運算、布林運算位元位移(bit-shift)及各種代數函數之計算。在一實施例中，能善用相同功能單元硬體來進行不同的運算且可能出現功能單元之任何組合。

【0072】傳送至處理叢集214的指令組成執行緒。跨成組的平行處理引擎執行之成組的執行緒為執行緒群(thread group)。執行緒群在不同單元輸入資料上執行相同的程式。在執行緒群內各個執行緒能被指定到在圖形多處理器234內不同的處理引擎。執行緒群可包括比在圖形多處理器234內的處理引擎之數目更少的執行緒。當執行緒群包括比處理引擎之數目更少的執行緒時，處理引擎之一或多者在正處理執行緒群的週期期間可為閒置的。執行緒群亦可包括比在圖形多處理器234內的處理引擎之數目更多的執行緒。當執行緒群包括比在圖形多處理器234內處理引擎之數目更多的執行緒時，能在連序的時鐘週期之上進行處理。在一實施例中，在圖形多處理器234上能並行地執行多個執行緒群。

【0073】在一實施例中，圖形多處理器234包括內部快取記憶體，用以進行載入和儲存運算。在一實施例中，圖形多處理器234能放棄內部快取而使用在處理叢集214內的快取記憶體(例如，L1快取308)。各個圖形多處理器234亦具有對在分割單元(例如，圖2A之分割單元220A~220N)內的L2快取的存取，該些分割單元係共用於所有處理叢集214之間且可被使用以於執行緒之間傳輸資料。圖形多處

理器234亦可存取晶片外全域記憶體(global memory)，其能包括局部平行處理器記憶體及/或系統記憶體之一或多者。外部於平行處理單元202的任何記憶體可被使用為全域記憶體。在其中處理叢集214包括圖形多處理器234之多個實例中的實施例能共用共同指令及資料，其可被儲存於L1快取308中。

【0074】各個處理叢集214可包括MMU 245(記憶體管理單元)，其組態以將虛擬位址映射至實體位址。在其它實施例中，MMU 245之一或多個實例可常駐於圖2A之記憶體介面218內。MMU 245包括成組的頁表項(PTE；page table entry)，其被使用來將虛擬位址映射至磚(更多是談論鋪磚)之實體位址且可選地映射至快取線索引。MMU 245可包括位址轉譯旁視緩衝器(TLB；translation lookaside buffer)或可常駐於圖形多處理器234內的快取或L1快取或處理叢集214。實體位址經處理以分布表面資料存取局部性(surface data access locality)以允許分割單元之間有效請求交插(interleaving)。可使用快取線索引以決定用於快取線的請求是否為命中或不中。

【0075】在圖形及計算應用中，可組態處理叢集214使得各個圖形多處理器234耦接至紋理單元236以用於進行紋理映射運算，例如，決定紋理樣本位置、讀取紋理資料以及過濾紋理資料。紋理資料係讀取自內部紋理L1快取(未繪示)或在一些實施例中讀取自在圖形多處理器234內的L1快取且如需要，提取自L2快取、局部平行處理器記憶

體或系統記憶體。各個圖形多處理器234輸出處理的任務給資料交叉240以提供經處理的任務給另一個處理叢集214，以用於進一步處理或用以在L2快取、局部平行處理器記憶體或經由記憶體交叉216的系統記憶體中儲存經處理的任務。預ROP 242(預光柵運算單元(pre-raster operations unit))組態以從圖形多處理器234接收資料，將資料引導至ROP單元，其可如於此所述以分割單元來定位(例如，圖2A之分割單元220A~220N)。預ROP 242單元能進行用於混色(color blending)的最佳化、組織像素色彩資料以及進行位址轉譯。

【0076】將理解的是，於此繪示的核心架構為例示性的且變異和修改是可能的。任何數目的處理單元，例如圖形多處理器234、紋理單元236、預ROP 242等，可被包括在處理叢集214內。進一步，在當僅繪示一處理叢集214的同時，於此所述的平行處理單元可包括任何數目的處理叢集214之實例。在一實施例中，各個處理叢集214能組態以使用分開且相異的處理單元、L1快取等獨立地操作其它處理叢集214。

【0077】圖2D依據一實施例繪示圖形多處理器234。在這類的實施例中，圖形多處理器234與處理叢集214之管線管理器234耦接。圖形多處理器234具有執行管線，其包括(但不限於)指令快取252、指令單元254、位址映射單元256、暫存器檔案258、一或多個通用圖形處理單元(GPGPU)核心262以及一或多個載入/儲存單元266。

GPGPU核心262和載入/儲存單元266係經由記憶體和快取互連268與快取記憶體272及共用記憶體270耦接。

【0078】在一實施例中，指令快取252接收指令之串流，用以執行自管線管理器232。指令被快取於指令快取252中且針對由指令單元254的執行調度。隨著執行緒群之各個執行緒指定至在GPGPU核心262內不同的執行單元，指令單元254能將指令調度為執行緒群(例如，線程束(warp))。藉由明定在統合位址空間內的位址，指令能存取局部、共用或全域位址空間之任一者。能使用位址映射單元256以將在統合位址空間中的位址轉譯到能由載入/儲存單元266存取的相異記憶體位址。

【0079】暫存器檔案258提供用於圖形多處理器324之功能單元的成組的暫存器。暫存器檔案258提供用於連接至圖形多處理器324之功能單元(例如，GPGPU核心262、載入/儲存單元266)的資料路徑的暫時儲存。在一實施例中，暫存器檔案258在功能單元之各者之間被劃分，使得各個功能單元被分配暫存器檔案258之專屬部分。在一實施例中，暫存器檔案258在正由圖形多處理器324執行的不同線程束之間被劃分。

【0080】GPGPU核心262各者能包括浮點單元(FPU；floating point unit)及/或整數算術邏輯單元(ALU；arithmetic logic unit)，其被使用來執行圖形多處理器324的指令。依據實施例，GPGPU核心262能在架構上類似或能在架構上不同。例如及在一實施例中，GPGPU核心262

之第一部分包括單精度FPU及整數ALU，同時GPGPU核心之第二部分包括雙精度FPU。在一實施例中，FPU能實施用於浮點算術的IEEE 754-2008標準或致能可變精度浮點算術。圖形多處理器324能額外地包括一或多個固定功能或特別功能單元用以進行特定功能，像是複製矩形或像素混合運算。在一實施例中，GPGPU核心之一或多者亦能包括固定或特別功能邏輯。

【0081】記憶體及快取互連268為將圖形多處理器324之功能單元之各者連接到暫存器檔案258和連接到共用記憶體270的互連網路。在一實施例中，記憶體及快取互連268為交叉互連，其允許載入/儲存單元266實施在共用記憶體270與暫存器檔案258之間的載入及儲存運算。暫存器檔案258能在與GPGPU核心262相同的頻率上操作，因此在GPGPU核心262與暫存器檔案258之間的資料傳輸為非常低潛時(latency)。共用記憶體270能被使用來致能在圖形多處理器234內之功能單元上執行的執行緒之間的通訊。快取記憶體272能被使用為例如資料快取，用以快取在功能單元與紋理單元236之間通訊的紋理資料。共用記憶體270亦能被使用為程式管理快取。除了自動地快取在快取記憶體272內儲存的資料以外，在GPGPU核心262上執行的執行緒能程式化地儲存在共用記憶體內的資料。

【0082】圖3A~3B依據實施例闡述額外的圖形多處理器。闡述的圖形多處理器325、350為圖2C之圖形多處理器234之變體。闡述的圖形多處理器325、350能組態為串流

多處理器(SM；streaming multiprocessor)，其能夠同時執行大數目的執行執行緒。

【0083】圖3A依據額外的實施例繪示圖形多處理器325。圖形多處理器325包括關於圖2D之圖形多處理器234的執行資源單元的多個額外實例。例如，圖形多處理器325能包括指令單元332A~332B、暫存器檔案334A~334B以及紋理單元344A~344B之多個實例。圖形多處理器325亦包括多組的圖形或計算執行單元(例如，GPGPU核心336A~336B、GPGPU核心337A~337B、GPGPU核心338A~338B)以及多組的載入/儲存單元340A~340B。在一實施例中，執行資源單元具有共同指令快取330、紋理及/或資料快取記憶體342以及共用記憶體346。各種組件能經由互連組構(interconnect fabric)327進行通訊。在一實施例中，互連組構327包括一或多個交叉開關(crossbar switch)，用以致能在圖形多處理器325之各種組件之間的通訊。

【0084】圖3B依據額外的實施例繪示圖形多處理器350。圖形處理器包括多組的執行資源356A~356D，其中執行資源之各組包括多個指令單元、暫存器檔案、GPGPU核心以及載入儲存單元，其如在圖2D及圖3A中所闡述。執行資源356A~356D能與紋理單元360A~360D協力工作以用於紋理運算，同時共用指令快取354及共用記憶體362。在一實施例中，執行資源356A~356D能共用指令快取354及共用記憶體362以及紋理及/或資料快取記憶體358A~358B之多個實例。各種組件能經由類似於圖3A之互連組

構327的互連結構352來進行通訊。

【0085】本領域具有通常知識者將了解，在圖1、2A~2D及3A~3B中所述的架構為描述性的且不限於本案實施例的範圍。因此，在不悖離於此說明的實施例之範圍下，於此所述的技術可在任何適當組態的處理單元上實施，包括(但不限於)一或多個行動應用處理器、一或多個桌上型電腦或伺服器中央處理單元(CPU)(包括多核心CPU)、一或多個平行處理單元，像是圖2A的平行處理單元202，以及一或多個圖形處理器或特殊用途處理單元。

【0086】在一些實施例中，如於此所述的平行處理器或GPGPU係通訊地耦接至主機/處理器核心以加速圖形運算、機器學習運算、圖樣分析運算以及各種通用GPU(GPGPU)功能。GPU可在匯流排或其它互連(例如，高速互連，像是PCIe或NVLink)之上通訊地耦接至主機處理器/核心。在其它實施例中，GPU可在相同的封裝上或晶片上被整合為核心且在內部處理器匯流排/互連(亦即，內部於該封裝或晶片)之上通訊地耦接至該核心。無論GPU被連接於其中的方式為何，處理器核心可以包含在工作描述符(work descriptor)中的命令/指令之序列的形式分配工作給GPU。接著GPU使用專屬電路/邏輯以用於有效地處理這些命令/指令。

用於GPU對主機處理器互連的技術

【0087】圖4A闡述複數個GPU 410~413在高速連結

440~443(例如，匯流排、點對點互連等)之上通訊地耦接至複數個多核心處理器405~406。在一實施例中，高速連結440~443支援4GB/s、30GB/s、80GB/s或更高的通訊傳輸量(communication throughput)，其取決於實施。可使用各種互連協定，包括(但不限於)PCIe 4.0或5.0和NVLink 2.0。然而，本發明的基本原理並不限於任何特定的通訊協定或傳輸量。

【0088】此外，在一實施例中，GPU 410~413之二或多者係在高速連結444~445之上互連，其可使用與用於高速連結440~443所使用的該些者相同或不同的協定/連結來實施。類似地，多核心處理器405~406之二或多者可在高速連結433之上連接，其可為在20GB/s、30GB/s、120GB/s或更高上運算的對稱多處理器(SMP；symmetric multi-processor)匯流排。或者，可使用相同協定/連結(例如在共同互連組構之上)完成在圖4A中繪示的各種系統組件之間的所有通訊。然而，如所提及的，本發明的基本原理並不限於任何特定類型的互連技術。

【0089】在一實施例中，各個多核心處理器405~406係分別經由記憶體互連430~431通訊地耦接至處理器記憶體401~402，且各個GPU 410~413係分別在GPU記憶體互連450~453之上通訊地耦接至GPU記憶體420~423。記憶體互連430~431及450~453可利用相同或不同的記憶體存取技術。藉由範例而非限制的方式，處理器記憶體401~402及GPU記憶體420~423可為揮發性記憶體，像是動態隨機存

取記憶體(DRAM)(包括堆疊DRAM)、圖形DDR SDRAM(GDDR)(例如，GDDR5、GDDR6)或高頻寬記憶體(HBM)及/或可為非揮發性記憶體，像是3D X點(3D XPoint)或奈米-Ram。在一實施例中，記憶體之某部分可為揮發性記憶體而另一部分可為非揮發性記憶體(例如，使用二級記憶體(2LM)階層)。

【0090】如上所述，雖然各種處理器405~406及GPU 410~413可分別實體地耦接至特定記憶體401~402、420~423，可實施統合記憶體架構，在其中相同的虛擬系統位址空間(亦稱為「有效位址(effective address)」空間)係在各種實體記憶體之所有者之間分布。例如，處理器記憶體401~402各者可包含64GB的系統記憶體位址空間，並且GPU記憶體420~423各者可包含32GB的系統記憶體位址空間(在此範例中造成總共256GB可定址的記憶體)。

【0091】圖4B闡述依據一實施例用於在多核心處理器407與圖形加速模組446之間互連的額外細節。圖形加速模組446可包括在線路卡(line card)上整合的一或多個GPU晶片，其經由高速連結440耦接至處理器407。或者，圖形加速模組446可整合在與處理器407相同的封裝或晶片上。

【0092】闡述的處理器407包括複數個核心460A~460D，各者具有轉譯旁視緩衝器461A~461D及一或多個快取462A~462D。核心可包括用於執行指令及處理資料的各種其它組件，其並未被闡述以為了避免模糊本發明的基本原理(例如，指令提取單元、分支預測單元、解碼器、執

行單元、重排序緩衝器等)。快取462A~462D可包含1級(L1)和2級(L2)快取。此外，一或多個共用快取426可被包括在快取階層中且由成組的核心460A~460D所共用。例如，處理器407之一實施例包括24個核心，各者具有它自己的L1快取、12個共用L2快取以及12個共用L3快取。在此實施例中，L2和L3快取其中一者係由兩個相鄰的核心所共用。處理器407和圖形加速器整合模組446與系統記憶體441連接，其可包括處理器記憶體401~402。

【0093】維持一致性(Coherency)以用於儲存在各種快取462A~462D、456及在經由一致性匯流排464之上核心間通訊的系統記憶體441中儲存的資料和指令。例如，各個快取可具有與其關聯的快取一致性邏輯/電路，用以反應於偵測的讀取在一致性匯流排464之上通訊至或寫入至特定快取線。在一實施中，快取窺探協定係在一致性匯流排464之上實施以用於窺探快取存取。快取窺探/一致性技術為本領域具有通常知識之該些者所周知的且於此不會詳細說明以避免模糊本發明的基本原理。

【0094】在一實施例中，代理器電路425通訊地將圖形加速模組446耦接至一致性匯流排464，其允許圖形加速模組446參加快取一致性協定，作為核心的同級(peer)。特別是，介面435在高速連結440(例如，PCIe匯流排、NVLink等)之上提供對代理器電路425的連接，且介面437將圖形加速模組446連接至連結440。

【0095】在一實施中，加速器整合電路436代表圖形

加速模組446之複數個圖形處理引擎431、432、N提供快取管理、記憶體存取、上下文管理(context management)以及中斷管理服務。圖形處理引擎431、432、N各者可包含分開的圖形處理單元(GPU)。或者，圖形處理引擎431、432、N可包含在GPU內不同類型的圖形處理引擎，像是圖形執行單元、媒體處理引擎(例如，視訊編碼器/解碼器)、取樣器及區塊影像傳輸引擎(blit engine)。換言之，圖形加速模組可為具有複數個圖形處理引擎431~432、N的GPU或圖形處理引擎431~432、N可為整合在共同封裝、線路卡或晶片上的個別GPU。

【0096】在一實施例中，加速器整合電路436包括用於進行各種記憶體管理功能的記憶體管理單元(MMU；memory management unit) 439，像是虛擬對實體記憶體轉譯(亦稱為有效對真實記憶體轉譯)和用於存取系統記憶體441的記憶體存取協定。MMU 439亦可包括轉譯旁視緩衝器(TLB)(未繪示)，以用於快取虛擬/有效對實體/真實位址轉譯。在一實施中，快取438儲存由圖形處理引擎431~432、N有效存取的命令和資料。在一實施例中，儲存在快取438和圖形記憶體433~434、N中的資料被保持與核心快取462A~462D、456以及系統記憶體411一致。如所提及的，此可經由代理器電路425來完成，其代表快取438及記憶體433~434、N參加快取一致性機制(例如，發送關於在處理器快取462A~462D、456上快取線的修改/存取的更新給快取438且從快取438接收更新)。

【0097】成組的暫存器445儲存用於由圖形處理引擎431~432、N執行的執行緒的上下文資料，並且上下文管理電路448管理執行緒上下文。例如，上下文管理電路448可進行存入和恢復運算以將在上下文切換期間各種執行緒之上下文存入及恢復(例如，其中存入第一執行緒且儲存第二執行緒，使得第二執行緒能由圖形處理引擎所執行)。例如，在上下文切換中，上下文管理電路448可儲存目前暫存器值到在記憶體中的指定區域(例如，由上下文指標所識別的)。接著，當回給上下文時，其可恢復暫存器值。在一實施例中，中斷管理電路447接收及處理從系統裝置接收的中斷。

【0098】在一實施中，來自圖形處理引擎431的虛擬/有效位址由MMU 439轉譯成在系統記憶體411中的真實/實體位址。加速器整合電路436之一實施例支援多個(例如，4、8、16)個圖形加速器模組446及/或其它加速器裝置。圖形加速器模組446可專用於在處理器407上執行的單一應用或是可共用於多個應用之間。在一實施例中，呈現有虛擬化圖形執行環境，在其中圖形處理引擎431~432、N之資源係與多個應用或虛擬機器(VM；virtual machine)共用。資源可被細分成「片(slice)」，其係基於與VM及/或應用關聯的處理要求和優先序而被分配到不同的VM及/或應用。

【0099】因此，加速器整合電路作為對用於圖形加速模組446的系統之橋接，並且提供位址轉譯和系統記憶體

快取服務。此外，加速器整合電路436可提供用於主機處理器的虛擬化設施，用以管理圖形處理引擎、中斷及記憶體管理的虛擬化。

【0100】因為圖形處理引擎431~432、N之硬體資源被明確地映射至由主機處理器407所見的真實位址，故任何主機處理器能直接使用有效位址值將該些資源定址。在一實施例中，加速器整合電路436的一功能為實體分開圖形處理引擎431~432、N，使得他們對系統就好像是獨立的單元。

【0101】如所提及的，在闡述的實施例中，一或多個圖形記憶體433~434、M係分別耦接至圖形處理引擎431~432、N之各者。圖形記憶體433~434、M儲存由圖形處理引擎431~432、N之各者處理的指令和資料。圖形記憶體433~434、M可為揮發性記憶體，像是DRAM)(包括堆疊DRAM)、GDDR記憶體(例如，GDDR5、GDDR6)或HBM及/或可為非揮發性記憶體，像是3D X點或奈米-Ram。

【0102】在一實施例中，為了降低在連結440之上的資料流量，使用偏移技術(biasing technique)以確保儲存在圖形記憶體433~434、M中的資料為由圖形處理引擎431~432、N最頻繁使用而較佳地是非由核心460A~460D使用(至少是非頻繁使用)的資料。類似地，偏移機制嘗試將由核心(且較佳地是非圖形處理引擎431~432、N)所需要的資料保留在核心及系統記憶體411之快取462A~462D、456內。

【0103】圖4C闡述加速器整合電路436被整合在處理器407內的另一實施例。在此實施例中，圖形處理引擎431~432、N直接在高速連結440之上經由介面437和介面435通訊至加速器整合電路436(又，其可利用任何形式的匯流排或介面協定)。加速器整合電路436可進行如相對圖4B所述的該些者相同的運算，但假定其緊鄰一致性匯流排462和快取462A~462D、426則潛在地於較高的傳輸量上。

【0104】一實施例支援不同程式模型，包括專屬進程程式模型(沒有圖形加速模組虛擬化)和共用程式模組(具有虛擬化)。後者可包括由加速器整合電路436控制的程式模型和由圖形加速模組446所控制的程式模型。

【0105】在專屬進程模型的一實施例中，圖形處理引擎431~432、N係專屬於單一應用或在單一作業系統下的處理。單一應用能將其它應用請求匯集到圖形引擎431~432、N，其提供在VM/分割內的虛擬化。

【0106】在專屬進程程式模型中，圖形處理引擎431~432、N能由多個VM/應用分割所共用。共用的模型需要系統超管理器(system hypervisor)將圖形處理引擎431~432、N虛擬化以允許由各個作業系統的存取。對於沒有超管理器的單一分割系統，圖形處理引擎431~432、N係由作業系統所擁有。在兩者的情形中，作業系統能將圖形處理引擎431~432、N虛擬化以提供對各個進程或應用的存取。

【0107】對於共用的程式模型，圖形加速模組446或

個別圖形處理引擎431~432、N使用進程柄程(process handle)來選擇進程元件。在一實施例中，進程元件係儲存在系統記憶體411中且使用於此所述有效位址對真實位址轉譯技術而為可定址的。當以圖形處理引擎431~432、N將其上下文登錄時(亦即，呼叫系統軟體用以將進程元件添加至進程元件連結的列表)，進程柄程可為提供給主機進程的特定實施值(mplementation-specific value)。進程柄程的較低16位元可為在進程元件連結的列表內進程元件之偏移。

【0108】圖4D闡述示範性加速器整合片490。如於此所使用的，「片」包含加速器整合電路436之處理資源的特定部分。在系統記憶體411內的應用有效位址空間482儲存進程元件483。在一實施例中，反應於來自在處理器407上執行的應用480之GPU調用481儲存進程元件483。進程元件483包含用於對應的應用480的進程狀態。包含在進程元件483中的工作描述符(WD；work descriptor)484能為由應用請求的單一工作或可包含對工作之佇列的指標。在後者的情形中，WD 484為對在應用的位址空間482中之工作請求佇列的指標。

【0109】圖形加速模組446及/或個別圖形處理引擎431~432、N能由在系統中的進程之所有或子集所共用。本發明之實施例包括用於設定進程狀態且發送WD 484至圖形加速模組446以在虛擬化環境中起始工作的基礎建設。

【0110】在一實施中，專屬進程程式模型為實施特定的(implementation-specific)。在此模型中，單一進程擁有圖形加速模組446或個別圖形處理引擎431。因為圖形加速模組446係由單一進程所擁有，在當圖形加速模組446被指定的時間處，超管理器將用於擁有分割的加速整合電路436初始化且作業系統將用於擁有進程的加速器整合電路436初始化。

【0111】在操作中，在加速器整合片490中的WD提取單元491提取下一個WD 484，其包括用以由圖形加速模組446之圖形處理引擎其中一者完成的工作之指示。來自WD 484的資料可被儲存在暫存器445中且由如所闡述的MMU 439、中斷管理電路447及/或上下文管理電路446所使用。例如，MMU 439之一實施例包括區段/頁面行走電路，以用於在OS虛擬位址空間485內存取區段/頁面表486。中段管理電路447可處理從圖形加速模組446接收的中段事件492。當進行圖形運算時，由圖形處理引擎431~432、N產生的有效位址493被轉譯成由MMU 439的真實位址。

【0112】在一實施例中，相同組的暫存器445被複製以用於各個圖形處理引擎431~432、N及/或圖形加速模組446，且可由超管理器或作業系統所初始化。這些複製的暫存器之各者可被包括在加速器整合片490中。可由超管理器初始化的示範性暫存器係繪示於表1中。

表1-超管理器初始化的暫存器

1	片控制暫存器
2	真實位址(RA)排程進程區域指標
3	權限遮罩置換暫存器
4	中斷向量表項偏移
5	中斷向量表項限制
6	狀態暫存器
7	邏輯分割ID
8	真實位址(RA)超管理器加速器利用記錄指標
9	儲存描述暫存器

【0113】可由作業系統初始化的示範性暫存器係繪示於表2中。

表2-作業系統初始化的暫存器

1	進程及執行緒識別
2	有效位址(EA)上下文存入/恢復指標
3	虛擬位址(VA)加速器利用記錄指標
4	虛擬位址(VA)存儲區段表指標
5	權限遮罩
6	工作描述符

【0114】在一實施例中，各個WD 484係特定於特別圖形加速模組446及/或圖形處理引擎431~432、N。其包含所有的資訊，圖形處理引擎431~432、N需要做其工作或其能為對記憶體位置的指標，其中應用已安裝要被完成的工作之命令佇列。

【0115】圖4E闡述針對共用模型之一實施例的額外細

節。此實施例包括超管理器真實位址空間498，在其中儲存進程元件列表499。超管理器真實位址空間498係經由將用於作業系統495的圖形加速模組引擎虛擬化的超管理器496。

【0116】共用程式模型允許來自在系統中分割之所有或子集的進程之所有或子集使用圖形加速模組446。有兩個其中圖形加速模組446係由多個進程和分割共用的程式模型：時間切片的共用和圖形指向的共用。

【0117】在此模型中，系統超管理器496擁有圖形加速模組446且使其功能對於所有作業系統495為可用的。對於用以由系統超管理器496支援虛擬化的圖形加速模組446，圖形加速模組446可遵循下列需求：1)應用的工作請求必需為自主的(亦即，狀態不需要維持於工作之間)，或圖形加速模組446必需提供上下文存入及恢復機制。2)應用的工作請求係由圖形加速模組446保證在特定時間量中完成，包括任何轉譯錯失，或圖形加速446提供用以搶先處理工作的能力。3)必需保證圖形加速模組446在當在指向共用程式模型中操作時於進程之間的公平性。

【0118】在一實施例中，對於共用的模型，需要應用480以使作業系統495進行系統呼叫，其具有圖形加速模組446類型、工作描述符(WD)、權限遮罩暫存器(AMR；authority mask register)值以及上下文存入/恢復區域指標(CSRP；context save/restore area pointer)。圖形加速模組446類型描述用於系統呼叫之目標的加速功能。圖形加速

模組446類型可為特定系統值。WD係對於圖形加速模組446明定地格式化且能以圖形加速模組446命令、對使用者界定的結構之有效位址指標、對命令之佇列的有效位址指標或用以描述要由圖形加速模組446完成的工作的任何其它資料結構的形式。在一實施例中，AMR值為用以針對目前進程使用的AMR狀態。傳遞到作業系統的值類似於應用設定AMR。若加速器整合電路436和圖形加速模組446實施並未支援使用者權限遮罩置換暫存器(UAMOR；User Authority Mask Override Register)，則在傳遞超管理器呼叫中的AMR之前，作業系統可將目前UAMOR值應用到AMR值。超管理器496可選擇地在將AMR放置到進程元件483中之前應用目前權限遮罩置換暫存器(AMOR)值。在一實施例中，CSRP為暫存器445之其中一者，其包含在用於圖形加速模組446的應用之位址空間482中區域的有效位址，用以存入且恢復上下文狀態。若沒有狀態需要被存入於工作之間或當工作被搶先時，則此指標為可選的。上下文存入/恢復區域為固定的系統記憶體。

【0119】在接收系統呼叫當下，作業系統495可驗證應用480已登錄且已給定權限來使用圖形加速模組446。接著，作業系統495呼叫超管理器496，其具有在表3中顯示的資訊。

表3-OS對超管理器呼叫參數

1	工作描述符(WD)
2	權限遮罩暫存器(AMR)值(潛在地為遮蔽的)
3	有效位址(EA)上下文存入/恢復區域指標(CSRP)
4	進程ID(PID)和可選的執行緒ID(TID)
5	虛擬位址(VA)加速器利用記錄指標(AURP)
6	儲存區段表指標(SSTP)之虛擬位址
7	邏輯中斷服務號碼(LISN)

【0120】在接收系統呼叫當下，超管理器496可驗證作業系統495已登錄且已給定權限來使用圖形加速模組446。接著，超管理器496將進程元件483放到用於對應圖形加速模組446類型的進程元件連結的列表中。進程元件可包括在表4中所繪示的資訊。

表4-進程元件資訊

1	工作描述符(WD)
2	權限遮罩暫存器(AMR)值(潛在地為遮蔽的)
3	有效位址(EA)上下文存入/恢復區域指標(CSRP)
4	進程ID(PID)和可選的執行緒ID(TID)
5	虛擬位址(VA)加速器利用記錄指標(AURP)
6	儲存區段表指標(SSTP)之虛擬位址
7	邏輯中斷服務號碼(LISN)
8	中斷向量表，取自超管理器呼叫參數。
9	狀態暫存器(SR)值
10	邏輯分割ID(LPID)
11	真實位址(RA)超管理器加速器利用記錄指標
12	儲存描述符暫存器(SDR)

【0121】在一實施例中，超管理器將複數個加速器整合片490暫存器445初始化。

【0122】如在圖4F中所闡述的，本發明之一實施例運用經由使用以存取實體處理器記憶體401~402及GPU記憶體420~423的共同虛擬記憶體位址空間而可存取的統合記憶體。在此實施中，在GPU 410~413上執行的運算利用相同的虛擬/有效記憶體位址空間來存取處理器記憶體401~402，且反之亦然，從而簡化可程式性(programmability)。在一實施例中，虛擬/有效位址空間之第一部分被分配到處理器記憶體401，第二部分被分配到第二處理器記憶體402，第三部分被分配到GPU記憶體420等。整個虛擬/有效記憶體空間(有時稱為有效位址空間)從而跨處理器記憶

體401~402及GPU記憶體420~423之各者而分布，其允許具有映射到該記憶體之虛擬位址的任何實體記憶體。

【0123】在一實施例中，在MMU 439A~439E之一或多者內的偏移/一致性管理電路494A~494E確保主機處理器(例如，405)與GPU 410~413之快取之間的快取一致性，且實施偏移技術，其指示應儲存某些類型的資料於其中的實體記憶體。在當偏移/一致性管理電路494A~494E之多個實例在圖4F中闡述的同時，偏移/一致性電路可被實施於一或多個主機處理器405之MMU內及/或於加速器整合電路436內。

【0124】一實施例允許GPU附接的記憶體420~423被映射為部分的系統記憶體，且使用共用虛擬記憶體(SVM；shared virtual memory)技術來存取，而沒有遭受與全系統快取一致性關聯的典型效能缺陷。GPU附接記憶體420~423要被存取為系統記憶體而沒有繁重的快取一致性負擔的能力提供用於GPU卸載之有利的作業環境。此安排允許主機處理器405軟體設定運算元且存取計算結果，而沒有傳統I/O DMA資料副本的負擔。這類傳統副本包含驅動呼叫、中斷以及記憶體映射I/O(MMIO；memory mapped I/O)存取，其相關於簡單記憶體存取全都是低效的。同時，存取GPU附接記憶體420~423而沒有快取一致性負擔的能力對卸載的計算之執行時間是關鍵的。例如，在具有實質串流化寫入記憶體流量的情形中，快取一致性負擔能明顯地降低由GPU 410~413所見的有效寫入頻寬。運算元

設定的效率、結果存取的效率以及GPU計算的效率全都在決定GPU卸載之效益(effectiveness)上扮演角色。

【0125】在一實施中，於GPU偏移與主機處理器偏移之間的選擇係由偏移追蹤器資料結構所驅動。可使用例如為頁面粒度結構(亦即，在記憶體頁面之粒度上進行控制)的偏移表，其包括每GPU附接記憶體頁面1或2位元。可在一或多個GPU附接記憶體420~423之被盜的記憶體範圍中實施偏移表面，無論在GPU 410~430中有或沒有偏移快取(例如，用以將頻繁/最近使用的偏移表之項目進行快取)。或者，可在GPU內維護整體偏移表。

【0126】在一實施中，與對GPU附接記憶體420~423之各個存取關聯的偏移表項係在對GPU記憶體實際存取之前被存取，造成下列運算。首先，來自在GPU偏移中發現他們的頁面的GPU 410~413的局部請求被直接轉送到對應的GPU記憶體420~423。來自在主機偏移中發現他們的頁面的GPU的局部請求被轉送到處理器405(例如，如上討論的高速連結)。在一實施例中，來自在主機處理器偏移中發現請求的頁面的處理器405的請求像正規記憶體讀取一般完成該請求。或者，關於GPU偏移的頁面的請求可被轉送到GPU 410~413。接著，若目前未正使用該頁面，則GPU可將該頁面轉移至主機處理器偏移。

【0127】頁面之偏移狀態能由基於軟體的機制、硬體協助基於軟體的機制或是針對有限組的情形純基於硬體的機制其一者所改變。

【0128】用於改變偏移狀態的一機制運用API呼叫(例如，OpenCL)，其輪流呼叫GPU的裝置驅動，其輪流發送訊息(或將命令描述符進入佇列)給GPU，引導其改變偏移狀態且對於一些轉移進行在主機中的快取排清運算(flushing operation)。快取排清運算係需要從主機處理器405偏程轉移到GPU偏移，但不需要相反的轉移。

【0129】在一實施例中，快取一致係由暫時呈現GPU偏移頁面(不可由主機處理器405快取的)維護。為了存取這些頁面，處理器405可從GPU 410請求存取，該GPU 410取決於實施可或不可准予馬上存取。因此，為了降低處理器405與GPU 410之間的通訊，有利的是，確保GPU偏移頁面為由GPU所需要的的該些者而不是主機處理器所需要的，且反之亦然。

圖形處理管線

【0130】圖5依據實施例闡述圖形處理管線500。在一實施例中，圖形處理器能實施闡述的圖形處理管線500。圖形處理器能被包括在如於此所述的平行處理子系統內，像是圖2A的平行處理器200，在一實施例中為圖1之平行處理器112的變體。各種平行處理系統能經由於此所述平行處理單元之一或多個實例(例如，圖2A之平行處理單元202)來實施圖形處理管線500。例如，著色器單元(例如，圖2D之圖形多處理器234)可組態以進行頂點處理單元504、曲面細分控制處理單元508、曲面細分評估處理單元

512、幾何處理單元516以及片段(fragment)/像素處理單元524之一或多者的功能。資料組譯器(data assembler)502、基元組譯器(primitive assembler)506、514、518、曲面細分單元510、柵格器522以及光柵運算單元526的功能亦可由在處理叢集(例如，圖3A之處理叢集214)內其它處理引擎和對應的分割單元(例如，圖2C之分割單元220A~220N)來進行。圖形處理管線500亦可使用針對一或多個功能的專屬處理單元來實施。在一實施例中，圖形處理管線500的一或多個部分能由在通用處理器(例如，CPU)內的平行處理邏輯來進行。在一實施例中，圖形處理管線500之一或多個部分能經由記憶體介面528存取晶片上記憶體(例如，如在圖2A中的平行處理記憶體222)，其可為圖2A之記憶體介面218的實例。

【0131】在一實施例中，資料組譯器502為收集用於表面及基元的頂點資料的處理單元。資料組譯器502接著將頂點資料，包括頂點屬性，輸出到頂點處理單元504。頂點處理單元504為可程式執行單元，其執行頂點著色器程式、打光及將頂點資料變換為由頂點著色器程式所明定。頂點處理單元504讀取儲存在快取、局部或系統記憶體中的資料以用於在處理頂點資料中使用，且可被程式化以將頂點資料從物件基礎座標表示變換成世界空間座標空間(world space coordinate space)或正規化裝置座標空間。

【0132】基元組譯器506之第一實例從頂點處理單元504接收頂點屬性。基元組譯器506讀取如所需儲存的頂點

屬性，且建構用於由曲面細分控制處理單元508處理的圖形基元。圖形基元包括如由各種圖形處理應用程式介面(API；application programming interface)支援的三角形、線區段、點、補丁等。

【0133】曲面細分控制處理單元508將輸入頂點視為用於幾何補丁的控制點。控制點係從來自補丁的輸入表示(例如補丁的基底)變換成合適於在由曲面細分評估處理單元512的表面評估中使用的表示。曲面細分控制處理單元508亦能計算用於幾何補丁之邊緣的曲面細分因子。曲面細分因子應用至單一邊緣且將與該邊緣關聯的細節之視點相依(view-dependent)級數量化。曲面細分單元510係組態以接收用於補丁之邊緣的曲面細分因子且用以將補丁曲面細分成多個幾何基元，像是線、三角形或四邊形基元，其被傳送到曲面細分評估處理單元512。曲面細分評估處理單元512在被細分的補丁之參數化座標上運算，用以對於與幾何基元關聯的各個頂點產生表面表示及頂點屬性。

【0134】基元組譯器514之第二實例接收來自曲面細分評估處理單元512的頂點屬性、如需要的讀取儲存頂點屬性以及建構用於由幾何處理單元516處理的幾何基元。頂點處理單元516為可程式執行單元，其執行幾何著色器程式，用以將從基元組譯器514接收的圖形基元變換為由幾何著色器程式明定。在一實施例中，幾何處理單元516被程式化以將圖形基元細分成一或多個新的圖形基原且計算使用以將新的圖形基元柵格化的參數。

【0135】在一些實施例中，幾何處理單元516能在幾何串流中添加或刪除元件。幾何處理單元516輸出明定新的圖形基元的參數和頂點到基元組譯器518。基元組譯器518從幾何處理單元516接收參數及頂點且建構用於由視埠定比(viewport scale)、剔除(cull)及剪輯單元520處理的圖形基元。幾何處理單元516讀取儲存在平行處理器記憶體或系統記憶體中的資料，以用於在處理幾何資料中使用。視埠定比、剔除及剪輯單元520進行剪輯、剔除及視埠定比且將處理的圖形基元輸出到柵格器522。

【0136】柵格器522能進行深度剔除及其它基於深度的最佳化。柵格器522亦進行在新的圖形基元上之掃描轉換用以產生片段，並且將該些片段及關聯的涵蓋資料輸出到片段/像素處理單元524。

【0137】片段/像素處理單元524為可程式執行單元，其組態以執行片段著色器程式或像素著色器程式。片段/像素處理單元524將從柵格器522接收的片段或像素變換為由片段或像素著色器程式所明定。例如，片段/像素處理單元524可被程式化以進行運算，其包括(但不限於)紋理映射、著色、混色、紋理校正及透視校正，用以生成著色的片段或像素，其被輸出至光柵運算單元526。片段/像素處理單元524可讀取儲存在平行處理器記憶體或系統記憶體中的資料，以用於在當處理片段資料時使用。片段或像素著色器程式可組態以在樣本、像素、磚或其它粒度上著色，其係取決於組態用於處理單元的取樣率。

【0138】光柵運算單元526為進行光柵運算的處理單元，包括(但不限於)模板、z測試(z test)、混色等，且將像素資料輸出為在圖形記憶體中儲存的經處理的圖形資料，圖形記憶體例如像在圖2A中的平行處理器記憶體222，及/或如在圖1中的系統記憶體104，用以在一或多個顯示裝置110上顯示，或用於由一或多個處理器102或平行處理器112進一步處理。在一些實施例中，光柵運算單元526組態以將寫入記憶體的z或色彩資料壓縮且將自記憶體讀出的z或色彩資料解壓縮。

【0139】圖6闡述依據一實施例的主控屏障及同步機制610的計算裝置600。計算裝置600代表通訊及資料處理裝置，包括(但不限於)智慧穿戴裝置、智慧電話、虛擬實境(VR)裝置、頭戴顯示器(HMD；head-mounted display)、行動電腦、物聯網(IoT)裝置、膝上型電腦、桌上型電腦、伺服器電腦等，且類似或相同於圖1之計算裝置100；據此，為了簡潔、明確及易於了解，許多上面參考圖1~5陳述的細節之後不會進一步討論或重複。

【0140】計算裝置600可更包括(沒有限制)自主機器或人工智慧代理器，像是機械代理器或機器、電子代理器或機器、虛擬代理器或機器、電子機械代理器或機器等。自主機器或人工智慧代理器的範例可包括(沒有限制)機器人、自主車輛(例如，自動駕駛汽車、自動飛行飛機、自動航行船等)、自主設備(自操作建構車輛、自操作醫療設備等)及/或類似者。遍及此文件的「計算裝置」可互換地

稱為「自主機器」或「人工智慧代理器」或簡單稱「機器人」。

【0141】可預期的是，雖然「自主車輛」和「自主駕駛」遍及此文件來參考，但實施例並不限於如此。例如，「自主車輛」並不限於自動車，但是其可包括任何數目及類型的自主機器，像是機器人、自主設備、家用自主裝置及/或類似者，且關於這類自主機器的任一或更多任務或運算可與自主驅動可交換地參考。

【0142】計算裝置600可更包括(沒有限制)大型計算系統，像是伺服器電腦、桌上型電腦等及可更包括機上盒(set-top box)(例如，基於網際網路纜線電視機上盒等)、基於全球定位系統(GPS)的裝置等。計算裝置600可包括作為通訊裝置的行動計算裝置，像是蜂巢式電話，包括智慧電話、個人數位助理(PDA)、平板電腦、膝上型電腦、電子閱讀器(e-reader)、智慧電視、電視平台、可穿戴裝置(例如，眼鏡、手錶、手環、智慧卡、首飾、服裝項目等)、媒體播放器等。例如，在一實施例中，計算裝置600可包括運用主控積體電路(「IC」)之電腦平台的行動計算裝置，像是在晶片上系統(「SoC」或「SOC」)，其在單一晶片上整合計算裝置600之各種硬體及/或軟體組件。

【0143】如所闡述的，在一實施例中，計算裝置600可包括任何數目及類型的硬體及/或軟體組件，像是(沒有限制)圖形處理單元(「GPU」或簡單稱「圖形處理器」)614、圖形驅動器(亦稱為「GPU驅動器」、「圖形驅

動邏輯」、「驅動器邏輯」、使用者模式驅動器(UMD；user-mode driver)、UMD、使用者模式驅動器框架(UMDF；user-mode driver framework)、UMDF或簡單稱「驅動器」)616、中央處理單元(「CPU」或簡單稱「應用處理器」)612、記憶體608、網路裝置、驅動器或類似者，以及輸入/輸出(I/O)來源604，像是觸控螢幕、觸控平板、觸控墊、虛擬或正規鍵盤、虛擬或正規滑鼠、埠、連接器等。計算裝置600可包括作業系統(OS)606，其作為在計算裝置600之硬體及/或實體資源與使用者之間的介面。可預期的是，圖形處理器614和應用處理器612可為圖1之處理器102中之一或多者。

【0144】要理解的是，比上述的範例更少或更多的系統對於某些實施是較佳的。因此，計算裝置600的組態取決於眾多的因子而隨實施而變化，像是價錢約束、效能需求，技術改善或其它情況。

【0145】實施例可被實施為一或多個使用母板互連的微晶片或積體電路、硬體邏輯、由記憶體裝置儲存及由微處理器執行的軟體、韌體、特定應用積體電路(ASIC)及/或場可編程閘陣列(FPGA)之任一者或組合。藉由範例的方式，術語「邏輯」、「模組」、「組件」、「引擎」及「機制」可包括軟體或硬體及/或軟體和硬體的組合。

【0146】在一實施例中，屏障和同步機制610可由計算裝置600之作業系統606所主控或促進。在另一實施例中，屏障和同步機制610可由圖形處理單元(「GPU」或簡

單稱「圖形處理器」)614或圖形處理器614之韌體或其部分所主控。例如，屏障和同步機制610可嵌入或實施為圖形處理器614之處理硬體的一部分。類似地，又在另一實施例中，儲存機制610可由中央處理單元(「CPU」或簡單稱「應用處理器」)612或其部分所主控。例如，屏障和同步機制610可嵌入或實施為應用處理器612之處理硬體的一部分。又在另一實施例中，屏障和同步機制610可由任何數目及類型的計算裝置600的組件或其部分所主控，像是屏障和同步機制610之部分可由作業系統606或其部分所主控，另一部分可由圖形處理器614或其部分所主控，另一部分可由應用處理器612或其部分所主控，同時屏障和同步機制610之一或多個部分可由作業系統606或其部分及/或任何數目及類型的計算裝置600的裝置所主控。預期的是，可運用屏障和同步機制610之一或多個部分或組件作為硬體、軟體及/或韌體。

【0147】預期的是，實施例並不限於屏障和同步機制610之任何特定實施或主控，且屏障和同步機制610及其組件的一或多者可被實施為硬體、軟體、韌體或其任何組合。

【0148】計算裝置600可主控網路介面以提供對網路的存取，網路像是LAN、寬域網路(WAN)、都會區域網路(MAN；metropolitan area network)、個人區域網路(PAN；personal area network)、藍牙、雲端網路、行動網路(例如，第三代(3G)第四代(4G)等)、內網、網際網路等。網路

介面可例如包括具有天線的(其可表示一或多個天線)無線網路介面。網路介面亦可例如包括有線的網路介面，用以經由網路纜線與遠端裝置通訊，其可例如為乙太纜線、共軸纜線、光纖纜線、串列纜線(serial cable)或平行纜線。

【0149】例如，依據於此所述的實施例，實施例可被提供為電腦程式產品，其可包括具有機器可執行指令儲存於其上的一或多個機器可讀媒體，當由一或多個機器(像是電腦、電腦之網路或其它電子裝置)執行時，可造成一或多個機器實現運算。機器可讀媒體可包括(但不限於)軟碟片、光碟片、CD-ROM(唯讀光碟)以及磁光碟、ROM、RAM、EPROM(可抹除可程式唯讀記憶體)、EEPROM(電可抹除可程式唯讀記憶體)、磁或光卡、快閃記憶體或用於儲存機器可執行指令的其它類型的媒體/機器可讀媒體。

【0150】再者，實施例可被下載為電腦程式產品，其中經由通訊連結(例如，數據機及/或網路連接)藉由體現於載波或其它傳播媒體及/或由載波或其它傳播媒體調變的一或多個資料信號的方式，程式可從遠端電腦(例如，伺服器)傳輸到請求電腦(例如，客戶端)。

【0151】遍及本文件的術語「使用者」可互換地稱為「觀眾」、「觀察者」、「人員」、「個人」、「終端使用者」及/或類似者。要注意的是，遍及本文件的術語，像是「圖形域(graphics domain)」可互換地參考以「圖形處理單元」、「圖形處理器」或簡單稱「GPU」，且類似

地「CPU域」或「主機域」可互換地參考以「電腦處理單元」、「應用處理器」或簡單稱「CPU」。

【0152】要注意的是，術語像是「節點」、「計算節點」、「伺服器」、「伺服器裝置」、「雲端電腦」、「雲端伺服器」、「雲端伺服器電腦」、「機器」、「主機機器」、「裝置」、「計算裝置」、「電腦」、「計算系統」及類似者遍及本文件可被互換地使用。更要注意的是，術語像是「應用」、「軟體應用」、「程式」、「軟體程式」、「套裝」、「軟體套裝」及類似者遍及本文件可被互換地使用。亦同樣，術語像是「工作」、「輸入」、「請求」、「訊息」及類似者遍及本文件可被互換地使用。

【0153】圖7闡述依據一實施例圖6的屏障和同步機制610。為了簡潔，參考圖1~6已討論的細節中的許多者之後不會重複或討論。在一實施例中，屏障和同步機制610可包括任何數目及類型的組件，像是(沒有限制)：偵測/觀察邏輯701；屏障邏輯703；同步邏輯705；通訊/相容性邏輯707；以及搶先和排程邏輯709。

【0154】關於屏障，傳統的技術被嚴格地限制於他們主要聚焦在基於軟體的同步，其在成本及資源方面具有其限制和低效，資源像是電力、時間、記憶體等。進一步，這類的技術並非針對全域方法提供給所有處理器，處理器像是GPU。

【0155】實施例係針對新穎技術而提供以用於運用用

於機器學習的多晶粒屏障，其中跨GPU而提供同步，像是給予硬體加速以用於跨GPU晶粒的屏障。

【0156】可預期的是，屏障係典型地使用以將在執行緒群內的執行緒同步。然而，執行屏障指令之在執行緒群中的任何執行緒停頓直到在該群中所有的執行緒到達與由他們執行屏障指令所發信相同的點。換句話說，屏障指令需要發信和同步硬體，其將他們限制到計算元件的叢集或方塊，因為任何執行緒群會需要在其對應的計算元件之方塊/叢集內被排程。由於機器學習能具有比較大數目的執行緒，此將執行緒限定到計算資源之較小叢集/方塊限制了整體處理量。

【0157】在一實施例中，在使用屏障邏輯703上，提供新穎的多晶粒屏障以用於在執行緒群中的執行緒，使得執行緒可跨較大的計算元件之組來被排程，如相對圖8A所闡述的。例如，可使用偵測/觀察邏輯701以偵測任何數目及類型的需要透過多晶粒屏障完成之排程或調度的執行緒和執行緒群。

【0158】在一實施例中，在使用屏障邏輯703上，運用多晶粒屏障以促進排程所有跨計算元件的執行緒及/或執行緒群，而不被連繫下至計算元件之特定方塊。進一步，為了確保此技術之適當的工作，屏障邏輯703促進必要的控制和同步以允許跨與多晶粒屏障關聯的所有晶粒之資料/資訊的無縫通訊，使得可維護在多極GPU(像是圖形處理器614)之間的同步和次序。

【0159】實施例更針對新穎技術提供用於如由同步邏輯705所促進的在執行緒群內且跨執行緒群的預執行緒同步(pre-thread synchronization)。在一實施例中，可使用同步邏輯705以不僅在執行緒群內維護同步，且亦藉由將籬跨該些執行緒群延伸而沒有需要需求任何屏障來跨多個執行緒群。

【0160】目前應用程式介面(API；application programming interface)允許僅在執行緒群內同步(像是藉由使用執行緒群屏障)。在一實施例中，在使用同步邏輯705上，添加新的屏障命令以用於跨執行緒群同步，使得多個執行緒群可被同步。例如，X執行緒群命令可與作為部分的引數(argument)之成列的執行緒群身份(ID)關聯，其中列於該引數中且由他們的ID識別的這些執行緒群可接著被同步。進一步，由於不同的執行緒群能在不同的串流多處理器(SM；streaming multiprocessor)中運行，GPU硬體(像是圖形處理器614之硬體)可添加交叉或DSS組構以將屏障同步跨該些SM進行通訊。此係闡述於圖8B中。

【0161】實施例更提供用於使用屏障的執行緒群同步，像是使用全域屏障，用以如由同步邏輯705促進的跨執行緒群進行同步。例如，如相對圖8C所闡述的，跨GPU(像是圖形處理器614)的資料共用可使用表面庫來進行，像是在卷積神經網路(CNN；convolutional NN)上工作的GPU可將中間NN資料存入為在雲端資料庫中的資料表面(data surface)。在相同或類似問題上工作的其它深度學

習系統可使用中間NN資料，像是自主車輛，像是並行的自主機器600，要用以具有類似的視點且能共用NN資料。

【0162】進一步，在一實施例中，在使用同步邏輯705上，此方式生成的任何表面可選擇地被壓縮以用於較快的傳輸時間，其中可使用共用的表面以用於由二或多個深度學習系統(像是自主機器600)的交叉檢查(cross-check)結果。此可造成較高的容錯能力(fault tolerance capability)，像是並行的兩部車輛可交叉檢查他們的中間NN表面足夠接近。

【0163】實施例更針對新穎技術提供用於跨SM(或DSS)或GPU排程執行緒群，其中這些執行緒群並未具有對共用局部記憶體(SLM；shared local memory)的存取。

【0164】在目前的GPU實施中，在執行緒群中所有執行緒在相同的SM(或DSS)上運行，係由於他們共用相同SLM及/或使用屏障以同步。例如，每SM實施SLM和硬體屏障，因而為了使屏障和SLM作用，在執行緒群中的所有執行緒在相同的SM調度。

【0165】在一實施例中，使用在調度中的停頓使得假定GPU接著在任何給定時間具有2個SM，SM0具有10個自由執行緒且SM1具有6個自由執行緒，其中接著新的執行緒群可準備好以16個執行緒來排程。在此情形中，目前，2個SM沒有一個可適配執行緒間隙(thread gap)，係由於他們不具有16個自由執行緒。然而，在2個SM之間，有足夠數目的自由執行緒以將新的執行緒群排程。

【0166】在一實施例中，在使用搶先和排程邏輯709上，可改變執行緒排程器政策使得編譯器及/或驅動器(像是圖6之驅動器616)可傳遞且決定執行緒群是否正使用任何屏障連同偵測由執行緒群使用的SLM尺寸。現若執行緒群不使用任何屏障或SLM，則接著可藉由搶先和排程邏輯709促進執行緒排程器以排程跨多個SM的該執行緒群。

【0167】然而，若執行緒群使用小量的SLM(其不被搶先和排程邏輯709認為對系統效能是關鍵的)，則接著可由搶先和排程邏輯709促進執行緒排程器以將該SLM空間映射到全域記憶體且後續跨多個SM排程該執行緒群。在一些實施例中，可由搶先和排程邏輯709促進編譯器以估計將SLM空間映射至全域記憶體且將該提示傳遞到執行緒排程器上是否是可接受的(像是效能方面)。這些提示可接著變為在記憶體中部分的執行緒群狀態。此參考圖8D闡述且進一步討論。

【0168】實施例更針對新穎技術提供用於基於如由搶先和排程邏輯709促進的屏障之執行緒群搶先。

【0169】可預期的是，屏障係使用以將在執行緒群內的執行緒同步。可停頓在執行屏障指令之給定的執行緒群中任何執行緒，直到在該群中所有執行緒到達如由執行該屏障指令的執行緒群發信的該點。此屏障指令可需要發信和同步硬體，其中在執行緒群中的執行緒執行屏障，其等待所有其它執行緒。在當大多數執行緒可藉由到達該屏障完成執行的同時，該進程可又被停頓直到所有其它執行緒

或執行緒群已到達該點。對於同步的需要可導致硬體資源之子最佳利用和降低的工作量之處理量。

【0170】在一實施例中，可使用搶先和排程邏輯703以允許執行緒群被搶先，同時允許其它執行緒群持續進展。例如，當第二執行緒群完成執行或進入類似屏障停頓狀態時，第一執行緒群可切換到執行。在第一實施例中，搶先和排程邏輯703可允許執行緒排程器維護搶先執行緒列表及狀態，同時其可具有關於在給定時間可搶先多少個執行緒群的最大限制，用以保持及維護對可管理尺寸的追蹤控制。此係相對於圖8E來闡述。

【0171】進一步，可使用通訊/相容性邏輯707以促進在任何數目的計算裝置600之裝置與屏障和同步機制610之各種組件之間需要的通訊及相容性。

【0172】可使用通訊/相容性邏輯707以促進在計算裝置600與任何數目及類型的其它計算裝置(像是行動計算裝置、桌上型電腦、伺服器計算裝置等)；處理裝置或組件(像是CPU、GPU等)；捕捉/感測/偵測裝置(像是包括攝像機的捕捉/感測組件、深度感測攝像機、攝像機感測器、紅綠藍(「RGB」或「rgb」)感測器、麥克風等)；顯示裝置(像是包括顯示螢幕、顯示區域、顯示投影機等的輸出組件)；使用者/情境感知(context-awareness)組件及/或識別/驗證感測器/裝置(像是生物特徵測量(biometric)感測器/偵測器、掃描器等)；資料庫730，像是記憶體或儲存裝置、資料庫及/或資料源(像是資料儲存裝置、硬碟驅動、

固態驅動、硬碟、記憶體卡或裝置、記憶體電路等)；通訊媒體725，像是一或多個通訊通道或網路(例如，雲端網路、網際網路、內網、蜂巢式網路、鄰近網路，其像是藍牙、低能耗藍牙(BLE；Bluetooth low energy)、智慧藍牙(Bluetooth Smart)、鄰近Wi-Fi(Wi-Fi proximity)、射頻識別(RFID)、近場通訊(NFC)、人體區域網路(BAN；Body Area Network)等)；無線或有線通訊及相關的協定(例如，Wi-Fi®、WiMAX、乙太等)；連通及位置管理技術；軟體應用/網站(例如，社群及/或企業網路網站等、企業應用、遊戲及其它娛樂應用等)；以及程式語言等，之間的動態通訊和相容性，同時確保與正變遷的技術、參數、協定、標準的相容性。

【0173】進一步，特定商標、字、術語、片語、名稱及/或縮寫字(acronym)，像是「刪除」、「觀察」、「執行緒」、「執行緒群」、「記憶體」、「計算資源」、「排程」、「同步」、「搶先」、「串流多處理器」、「屏障」、「多晶粒屏障」、「訓練集合」、「自主機器」、「代理器」、「機器」、「車輛」、「機器人」、「駕駛/驅動」、「CNN」、「DNN」、「NN」、「執行單元」、「EU」、「共用局部記憶體」、「SLM」、「圖形串流」、「快取」、「圖形快取」、「GPU」、「圖形處理器」、「GPU域」、「GPGPU」、「CPU」、「應用處理器」、「CPU域」、「圖形驅動器」、「工作量」、「應用」、「圖形管線」、「管線進程」、「API」、

「3D API」、「OpenGL®」、「DirectX®」、「硬體」、「軟體」、「代理器」、「圖形驅動器」、「內核模式圖形驅動器」、「使用者模式驅動器」、「使用者模式驅動器框架」、「緩衝器」、「圖形緩衝器」、「任務」、「進程」、「運算/操作」、「軟體應用」、「遊戲」等的任何使用不應被解讀為將實施例限制到將標籤載於產品上或本文件之外部文獻中的軟體或裝置。

【0174】預期的是，任何數目及類型的組件可被添加至及/或自屏障和同步機制610移除，用以促進包括添加、移除及/或增強某些特徵的各種實施例。為了簡潔、明確且易於了解屏障和同步機制610，許多的標準及/或已知的組件(像是計算裝置中的該些者)於此並未被繪示或討論。預期的是，如於此所討論的實施例並不限於任何特定技術、拓撲、系統、架構及/或標準且足夠動態的採取且適應任何未來的改變。

【0175】圖8A闡述依據一實施例用於採用及使用用於機器學習的多晶粒屏障的架構設定800。為了簡潔，參考圖1~7先前討論的細節中的許多者之後可能不會討論或重複。進一步，實施例並不限於進程及/或組件之任何特定的架構布局、框架、設定或結構，像是設定800。

【0176】如先前所討論的，傳統的技術是低效的且不合適用於機器學習，由於這些技術常導致執行緒或執行緒群的停頓直到所有執行緒或執行緒群已到達相同點，其係由執行相同屏障指令的該些執行緒或執行緒群發信。換言

之，在傳統的技術中，執行緒/執行緒群被侷限以透過計算元件之特定方塊來被調度或排程而不超過，其係對於機器學習特別有疑義的，係由於這典型地會包含大數目的執行緒。

【0177】如闡述的，在一實施例中，在使用屏障邏輯703上，多晶粒屏障被運用於自主機器，像是圖6的自主機器600，以允許在執行緒群中的執行緒跨所有計算元件來排程且並不限定於計算元件之特定叢集或群。進一步，在一實施例中，如所闡述的，屏障邏輯703提供與且在任何數目及類型的處理器之間的通訊，處理器像是GPU0 801、GPU1 803、GPU2 805、GPU3 807，連同促進在GPU 801~807之間的屏障同步使得資訊之控制和同步可皆跨處理裝置來維護，像是GPU 801~807，其與多晶粒關聯。在一實施例中，多晶粒屏障可為硬體屏障，用以給予需要屏障來跨GPU晶粒810~870工作的硬體加速。然而，實施例並不限於此。

【0178】圖8B闡述依據一實施例用於在機器學習中促進執行緒群之同步的框架820。為了簡潔，參考圖1~8A先前討論的細節中的許多者之後可能不會討論或重複。進一步，實施例並不限於進程及/或組件之任何特定的架構布局、框架、設定或結構，像是框架820。

【0179】如先前參考圖7所討論且於此闡述的，可使用具有引數833的屏障命令831以促進在多個執行緒群之間的同步，像是TG0 821、TG1 823、TG2 825以及TGN

827，其可在多個不同的SM上運行。例如，可使用圖7之同步邏輯705以促進X執行緒屏障命令，像是具有列舉執行緒群ID之引數833的屏障命令831。在一實施例中，執行屏障命令831以促進且維護資料之同步及與對應至引數833之執行緒群ID的多個執行緒群821~827之間的通訊。這樣一來，執行緒群821~827被同步，即使他們在不同的SM上運行，GPU硬體可添加交叉SM(或DSS)組構以通訊跨SM的屏障同步。

【0180】圖8C闡述依據一實施例用於在機器學習中使用表面庫促進跨GPU共用的資料的框架840。為了簡潔，參考圖1~8B先前討論的細節中的許多者之後可能不會討論或重複。進一步，實施例並不限於進程及/或組件之任何特定的架構布局、框架、設定或結構，像是框架840。

【0181】如參考圖7於此所闡述及討論的，任何中間NN資料可在通訊媒體725(像是雲端網路)之上由各個DL系統841、843、845存入為在一或多個資料庫730(像是雲端資料庫)處的表面資料。預期的是，DL系統841、843、845各者可代表自主機器，像是自主車輛，且可在彼此緊密相鄰或附近內，像是若干個自主車輛一起在相同的道路上。

【0182】在一些實施例中，由DL系統841、843、845生成的任何表面資料可被壓縮以用於在DL系統841、843、845之間較快的傳送及用於為或變為框架840之部分的其它DL系統或項目。如此一來，任何數目的DL系統841、843、845能在通訊媒體725之上彼此通訊資料/資

訊，像是交叉檢查他們的中間NN表面足夠近。

【0183】圖9A闡述依據一實施例用於在機器學習中沒有屏障或共用局部記憶體下促進執行緒群之最佳化排程的圖形處理器614。為了簡潔，參考圖1~8C先前討論的細節中的許多者之後可能不會討論或重複。

【0184】如於此闡述及先前相對圖7所討論的，圖形處理器614被繪示為具有多個SM、SMM或DSS，像是SMM0 905和SMM1 907。闡述的實施例更繪示與執行緒排程器901通訊的SMM 905和907，其能夠從由圖7之搶先和排程邏輯709促進的編譯器及/或驅動器接收關於SLM、屏障等的資訊或暗示903。

【0185】例如，在一實施例中，EU0 911A、EU1 911B、EU2 913A及EU3 913B被繪示為來自相同的執行緒群且跨過SM來排程，像是SMM 907、907，係由於此特定的執行緒群不可使用任何屏障915、917或SLM 921、923。例如，若執行緒群使用未被判定為效能關鍵的小量或SLM，則接著執行緒排程器901可簡單地將該SML空間映射至全域記憶體，允許其跨多個SM排程執行緒群，像是SMM 907、907。

【0186】圖9B闡述依據一實施例用於在機器學習中基於屏障促進執行緒群搶先的框架950。為了簡潔，參考圖1~9A先前討論的細節中的許多者之後可能不會討論或重複。

【0187】如於此闡述及先前參考圖7所討論的，若干

個執行緒群(像是TG0 951、TG1 953及TG2 955)被繪示為與如由圖7之搶先和排程邏輯709所促進的計算資源(例如，計算元件、計算方塊等)和執行緒排程器959(像是硬體執行緒排程器)通訊。

【0188】如先前提及的，例如在使用圖7之搶先和排程邏輯709上，若三個TG 951、953、955要被調度且執行並且若TG1 953不是完成執行就是進入類似屏障停頓狀態，則例如接著TG0 951可被切換到由執行緒排程器959執行，而藉由使用計算資源957不必停頓全體進程。類似地，在使用此新穎技術和計算資源之可用性上，當在執行緒群中的執行緒執行屏障時，其非必需進入到停頓中且等待所有其它執行緒到達屏障點。此係藉由使用如由圖7之搶先和排程邏輯709所促進的執行緒排程器959來完成。

機器學習概述

【0189】機器學習演算法為能基於成組的資料學習的演算法。機器學習演算法之實施例能被設計來將在資料組/資料集內的高階抽象化模型化。例如，能使用圖像辨識演算法(image recognition algorithm)來決定給定輸入屬於的幾個類別的哪一個；回歸演算法(regression algorithm)能輸出給定輸入的數值；以及能使用圖樣識別演算法以產生轉譯的文字或進行文字對口語及/或口語辨識。

【0190】示範性類型的機器學習演算法為神經網路(neural network)。有許多類型的神經網路；神經網路的簡

單類型為前饋網路(feedforward network)。前饋網路可被實施為非循環圖形，在其中節點以層來安排。典型地，前饋網路拓撲包括輸入層和由至少一隱藏層所分開的輸出層。隱藏層將由輸入層所接收的輸入變換成對於在輸出層產生輸出有用的表示(representation)。網路節點係經由邊緣完全連接到在相鄰層中的節點，但在各個層內的節點之間沒有連接。在前饋網路之輸入層的節點處接收的資料經由基於分別與連接該層邊緣之各者關聯的係數(「權重」)來計算在網路中各個連續層的節點之狀態的活化功能(activation function)而被傳播(亦即，「前饋」)到輸出層的節點。取決於正由正被執行的演算法代表的特定模型，自神經網路演算法的輸出能採取各種形式。

【0191】在機器學習演算法能被使用來模型化特定問題之前，使用訓練資料集來訓練演算法。訓練神經網路包含選擇網路拓撲、使用代表由網路模型化的問題之成組的訓練資料以及調整權重直到網路模型以對於訓練資料集的所有實例的最小誤差來進行。例如，在用於神經網路的監督學習訓練過程期間，將由反應於代表在訓練資料集中的實例的網路所生成的輸出比較於用於該實例的「正確」有標(labeled)輸出，計算代表輸出與有標輸出之間的差的誤差信號，以及調整與連接關聯的權重以將該誤差最小化為透過網路之層反向傳播的誤差信號。當針對從訓練資料集之實例產生的輸出之各者的誤差被最小化時，該網路被視為「已訓練」。

【0192】機器學習演算法之正確性能明顯地受使用來訓練演算法的資料集之品質影響。訓練過程可以為計算密集的且在傳統通用處理器上可需要相當數量的時間。據此，使用平行處理硬體來訓練許多類型的機器學習演算法。隨著在神經網路中調整係數上進行的計算自然適於平行實施，此對於最佳化訓練神經網路特別有用。具體而言，許多機器學習演算法和軟體應用已適應於利用在通用圖形處理裝置內的平行處理硬體。

【0193】圖10為機器學習軟體堆層(software stack)1000之一般化圖。機器學習應用1002能組態以使用訓練資料集來訓練神經網路或使用經訓練的深度神經網路來實施機器智慧。機器學習應用1002能包括用於神經網路及/或能被使用來在部署之前訓練神經網路的專門軟體的訓練及推理功能性。機器學習應用1002能實施任何類型的機器智慧，包括(但不限於)影像辨識、映射和局部化、自主導航、口語合成、醫療成像或語言翻譯。

【0194】能經由機器學習框架1004來致能用於機器學習應用1002的硬體加速。機器學習框架1004能提供機器學習基元(primitive)庫。機器學習基元為由機器學習演算法共同進行的基本運算。沒有機器學習框架1004，機器學習演算法的開發者會需要創建和最佳化與機器學習演算法關聯的主要計算邏輯，接著發展將計算邏輯再最佳化為新的平行處理器。相反的，機器學習應用能組態以使用由機器學習框架1004所提供的基元進行必需的計算。示範性基元

包括張量卷積(tensor convolution)、活化功能以及池化(pooling)，其為在訓練卷積神經網路(CNN；convolutional neural network)的同時進行的計算運算。機器學習框架1004亦能提供基元以實施由許多機器學習演算法進行的基本線性代數子程式，像是矩陣和向量運算。

【0195】機器學習框架1004能處理從機器學習應用1002接收的輸入資料且產生適當的輸入到計算框架1006。計算框架1006能抽取提供給GPGPU驅動器1008的下層指令(underlying instruction)用以致能機器學習框架1004來利用經由GPGPU硬體1010的硬體加速，而不需要機器學習框架1004具有GPGPU硬體1010之架構的詳盡知識。此外，計算框架1006能跨各種類型及世代的GPGPU硬體1010致能用於機器學習框架1004的硬體加速。

GPGPU機器學習加速

【0196】圖11為依據實施闡述高度平行通用圖形處理單元1100。在一實施例中，通用處理單元(GPGPU)1100能組態以在處理與訓練深度神經網路關聯的計算工作量之類型上特別有效。此外，GPGPU 1100能直接被連結到GPGPU的其它實例以創建多GPU叢集來改善用於特別深度神經網路的訓練速度。

【0197】GPGPU 1100包括主機介面1102以致能與主機處理器的連接。在一實施例中，主機介面1102為快速PCI(PCI Express)介面。然而，主機介面亦能為特定供應

商通訊介面或通訊組構。GPGPU 1100從主機處理器接收命令且使用全域排程器1104來將與該些命令關聯的執行執行緒分布到成組的計算叢集1106A~H。計算叢集1106A~H共用快取記憶體1108。快取記憶體1108能作為用於在計算叢集1106A~H之內快取記憶體的較高階的快取。

【0198】GPGPU 1100包括經由成組的記憶體控制器1112A~B與計算叢集1106A~H耦接的記憶體1114A~B。在各種實施例中，記憶體1114A~B能包括各種類型的記憶體裝置，各種類型的記憶體裝置包括動態隨機存取記憶體(DRAM；dynamic random access memory)或圖形隨機存取記憶體，像是同步圖形隨機存取記憶體(SGRAM；synchronous graphics random access memory)，同步圖形隨機存取記憶體包括圖形雙資料率(GDDR；graphics double data rate)記憶體。在一實施例中，記憶體單元224A~N亦可包括3D堆疊記憶體，包括(但不限於)高頻寬記憶體(HBM；high bandwidth memory)。

【0199】在一實施例中，各個計算叢集GPLAB06A~H包括成組的圖形多處理器，像是圖4A的圖形多處理器400。計算叢集多類型的整數及浮點邏輯單元之圖形多處理器，其能在包括合適於機器學習計算的精度之範圍上進行計算運算。例如且在實施例中，計算叢集1106A~H之各者中浮點單元之至少子集能組態以進行16位元或32位元浮點運算，同時不同子集的浮點單元能組態以進行64位元浮點運算。

【0200】GPGPU 1100之多個實例能組態以操作為計算叢集。由用於同步和資料交換的計算叢集所使用的通訊機制隨實施例而改變。在一實施例中，GPGPU 1100之多個實例在主機介面1102之上進行通訊。在一實施例中，GPGPU 1100包括將GPGPU 1100與GPU連結1100耦接的I/O中樞1108，其致能對GPGPU之其它實例的直接連接。在一實施例中，GPU連結1100係耦接至專屬GPU對GPU橋，其致能在GPGPU 1100之多個實例之間的通訊和同步。在一實施例中，GPU連結1100與高速互連耦接，用以傳送及接收資料到其它GPGPU或平行處理器。在一實施例中，GPGPU 1100之多個實例係位在分開的資料處理系統中且經由網路裝置通訊，該網路裝置經由主機介面1102而為可存取的。在一實施例中，GPU連結1100能組態以除了主機介面1102以外或對主機介面1102為替代的來致能對主機處理器的連接。

【0201】在GPGPU 1100之闡述的組態能組態以訓練神經網路的同時，一實施例提供能組態用於在高效能或低功率推理平台內之部署的GPGPU 1100的選替組態。在推理組態中，GPGPU 1100包括相對於訓練組態更少的計算叢集1106A~H。此外，與記憶體1114A~B關聯的記憶體技術可在推理與訓練組態之間相異。在一實施例中，GPGPU 1100的推理組態能支援推理特定指令。例如，推理組態能對於一或更多的8位元整數內積(dot product)指令提供支援，其在用於部署的神經網路之推理運算期間被共同的使

用。

【0202】圖12闡述依據一實施例的多GPU計算系統1200。多GPU計算系統1200能包括經由主機介面切換器1204耦接至多GPGPU 1206A~D的處理器1202。在一實施例中，主機介面切換器1204為快速PCI(PCI Express)切換器裝置，其將處理器1202耦接至快速PCI匯流排，處理器1202在其上與成組的GPGPU 1206A~D通訊。多個GPGPU 1206A~D之各者能為圖11之GPGPU 1100之實例。GPGPU 1206A~D能經由成組的高速點對點GPU對GPU連結1216來互連。高速GPU對GPU連結能經由專屬的GPU連結(像是圖11的GPU連結1110)連接到GPGPU 1206A~D之各者。P2P GPU連結1216致能在GPGPU 1206A~D之各者之間的直接通訊，而不需要在處理器1202連接到其的主機介面匯流排之上通訊。隨著GPU對GPU流量引導至P2P GPU連結，主機介面匯流排保持可用於系統記憶體存取或用以與多GPU計算系統1200之其它實例通訊(例如經由一或多個網路裝置)。在當於闡述的實施例中GPGPU 1206A~D經由主機介面切換器1204連接到處理器1202的同時，在一實施例中，處理器1202包括對於P2P GPU連接1216的直接支援且能直接連接至GPGPU 1206A~D。

機器學習神經網路實施

【0203】由於此所述的實施例提供的計算架構能組態以進行特別適於訓練及部署用於機器學習的神經網路之平

行處理的類型。神經網路能被概括為具有圖形關係的功能的網路。如在本領域所周知的，有在機器學習中使用之各種類型的神經網路實施。如先前所述，神經網路之一示範性類型為前饋網路。

【0204】第二示範性類型的神經網路為卷積神經網路(CNN)。CNN為專門前饋神經網路，以用於處理具有已知、網格狀(grid-like)拓撲的資料，像是影像資料。據此，CNN係共同使用於計算視覺和影像辨識應用，但他們亦可被使用於其它類型的圖樣辨識，像是口語和語言處理。在CNN輸入層中的節點被組織成為成組的「過濾器」(由在視網膜中發現的接收區(receptive field)所激發的特徵偵測器)，以及各組的過濾器之輸出被傳播到在網路之連續層中的節點。用於CNN的計算包括，將卷積數學運算應用到各個過濾器以生成該過濾器的輸出。卷積為由兩個函數進行的專門種類的數學運算，用以生成為兩個原始函數其中一者的修改版本的第三個函數。在卷積網路術語中，對卷積的第一函數能稱為輸入，同時第二函數能稱為卷積內核。輸出可被稱為特徵映射(feature map)。例如，對卷積層的輸入能為介定輸入影像的各種色彩成分的資料之多維度陣列。卷積內核能為參數之多維度陣列，其中參數係由用於神經網路的訓練進程來適應。

【0205】遞歸神經網路(RNN；Recurrent neural network)為前饋神經網路的家族，其包括層之間的回饋連接。RNN藉由跨不同部分的神經網路共用參數資料致能將

順序資料模型化。用於RNN的架構包括週期/循環。隨著來自RNN的輸出資料之至少部分被使用為對於處理在序列中後續的輸入的回饋，週期代表在未來時間其本身值上可變的目前值的影響。由於可變的本性(語言資料能以該可變的本性組成)，此特徵使RNN特別有用於語言處理。

【0206】下面說明的圖呈現示範性前饋、CNN以及RNN網路以及針對分別訓練及部署該些類型的網路之各者的一般過程。將了解的是，這些說明為示範性的且不限於在此說明的任何特定實施例，並且闡述的概念一般能被應用到深度神經網路與一般的機器學習技術。

【0207】上述的示範性神網路能被使用來進行深度學習。深度學習為使用神網網路的機器學習。在深度學習中使用的深度神經網路為由多個隱藏層組成的人工神經網路，其係相對於包括僅單一隱藏層的淺神經網路。較深的神經網路一般對於訓練是更計算密集的。然而，網路之額外的隱藏層致能多步驟圖樣辨別，其造成相對於淺機器學習技術的降低輸出誤差。

【0208】在深度學習中使用的深度神經網路典型地包括前端網路，用以進行耦接至代表數學模型的後端網路的特徵辨識，該數學模型能基於提供給該模型的特徵表示進行運算(例如，物件分類、口語辨識等)。深度學習致能機器學習在不需要用以對於模型進行手工製作特徵工程下來被進行。相反的，深度神經網路能基於在輸入資料內的統計結構或相關性來學習特徵。能提供經學習的特徵給能將

偵測的特徵映射到輸出的數學模型。由網路使用的數學模型係一般專門用於要被進行的特定任務，且將使用不同的模型以進行不同的任務。

【0209】一旦神經網路被結構化，學習模型能被應用到網路以訓練網路來進行特定任務。學習模型說明如何在模型內調整權重以降低網路的輸出誤差。誤差的反向傳播(backpropagation)為使用以訓練神經網路的共同方法。輸入向量被提出給用於處理的網路。使用損失函數(loss function)和誤差值將網路之輸出比較於所欲的輸出係針對在輸出層中神經元之各者來計算。接著反向傳播誤差值直到各個神經元具有大致上代表其對原始輸出的貢獻的關聯誤差值。接著網路能使用演算法從誤差學習，演算法像是隨機梯度下降演算法(stochastic gradient descent algorithm)用以更新神經網路的權重。

【0210】圖13A~B闡述示範性卷積神經網路。圖13A闡述在CNN內的各種層。如在圖13A中所繪示，使用以將影像處理模型化的示範性CNN能接收描述輸入影像的紅、綠及藍(RGB)成分的輸入1302。輸入1302能由多個卷積層(例如，卷積層1304、卷積層1306)處理。自多個卷積層的輸出可選擇地由成組的完全連接層1308來處理。在完全連接層中的神經元具有對在先前層中所有活化完全連接，如先前對於前饋網路所描述的。自完全連接層1308的輸出能被使用來從網路產生輸出結果。在完全連接層1308內的活化能使用矩陣乘法取代卷積來計算。並非所有的CNN實施

係利用完全連接層DPLA08。例如，在一些實施中，卷積層1306能產生用於CNN的輸出。

【0211】卷積層被稀疏地連接，其不同於在完全連接層1308中發現的傳統神經網路組態。將傳統神經網路層完全連接，使得每一個輸出單元與每一個輸入單元互動。然而，如所闡述的，因為一區之卷積的輸出被輸入(取代在該區中節點之各者的分別狀態值)到後續層之節點，所以卷積層被稀疏地連接。與卷積層關聯的內核進行卷積運算，其輸出被發送到下一層。在卷積層內進行的維數(dimensionality)降低為致能CNN定比來處理大影像的一個態樣。

【0212】圖13B闡述在CNN之卷積層內示範性計算階段(computation stage)。對CNN之卷積層1312的輸入能在卷積層1314之三個階段中被處理。該三個階段能包括卷積階段1316、偵測器階段1318以及池化階段1320。卷積層1314接著能輸出資料到連續卷積層。網路之最終卷積層能產生輸出特徵映射資料或提供輸入給完全連接層，例如用以產生用於對CNN輸入的分類值。

【0213】在卷積階段1316中平行進行一些卷積以生成成組的線性活化。卷積階段1316能包括仿射變換(affine transformation)，其為能被明定為線性變換加平移(translation)何變換。仿射變換包括旋轉、平移、定比以及該些變換的組合。卷積階段計算連接至在輸入中特定區域的函數之輸出(例如，神經元)，其能被判定為與神經元

關聯的局部區域。神經元計算在神經元之權重與神經元連接到其的局部輸出中的區域之間的內積。自卷積階段1316的輸出界定由卷積層1314之連續階段生成的成組的線性活化。

【0214】線性活化能由偵測器階段1318處理。在偵測器階段1318中，各個線性活化係由非線性活化函數處理。非線性活化函數在沒有影響卷積層之接收區下增加整體網路之非線性性質。可使用幾個類型的非線性活化函數。一個特別類型為線性整流單元(ReLU；rectified linear unit)，其使用界定為$f(x)=max(0,x)$的活化函數，使得活化在0處被定限(thresholded)。

【0215】池化階段1320使用池化功能，其以附近輸出之摘要統計(summary statistic)取代卷積層1306之輸出。能使用池化功能以將平移不變性(translation invariance)引導至神經網路，使得對輸出的小平移不會改變池化的輸出。對局部平移的不變性在輸入資料中特徵之出現比特徵之精準位置更重要的情境中可以是有用的。在池化階段1320期間能使用各種類型的池化功能，包括最大池化、平均池化以及l2範數池化(l2-norm pooling)。此外，一些CNN實施不包括池化階段。相反的，這類實施替代及相對於先前卷積階段具有增加的跨步的額外卷積階段。

【0216】來自卷積層1314的輸出接著能由下一層1322處理。下一層1322能為額外的卷積層或完全連接層1308其中之一。例如，圖13A的第一卷積層1304能輸出到第二卷

積層1306，同時第二卷積層能輸出到完全連接層1308之第一層。

【0217】圖14闡述示範性遞歸神經網路1400。在遞歸神經網路(RNN)中，網路之先前狀態影響網路之目前狀態的輸出。RNN能使用各種函數以各種方式被建立。RNN之使用一般圍繞著使用數學模型來基於先前輸入之序列預測未來。例如，可使用RNN以進行統計語言模型化來預測給定先前字之序列的即將來臨的字。闡述的RNN 1400能被描述為具有輸入層1402，其接收輸入向量、用以實施遞歸功能的隱藏層1404、用以致能先前狀態之「記憶」的回饋機制1405以及用以輸出結果的輸出層1406。RNN 1400基於時間步驟來運算。在給定時間步驟的RNN之狀態係基於經由回饋機制1405的先前時間步驟而受影響。對於給定的時間步驟，隱藏層1404之狀態係由先前狀態和在目前時間步驟的輸入所界定。在第一時間步驟的初始輸入(x_1)能由隱藏層1404處理。第二輸入(x_2)能使用在處理初始輸入(x_1)期間決定的狀態資訊由隱藏層1404處理。給定的狀態能被計算為$s_t=f(Ux_t+Ws_{t-1})$，其中U和W為參數矩陣。函數f一般為非線性，像是雙曲線正切函數(Tanh)或整流器函數$f(x)=max(0,x)$之變體。然而，在隱藏層1404中使用的特定數學函數能取決於RNN 1400之特定實施細節來變化。

【0218】除了上述基本CNN及RNN網路，可致能在該些網路上的變異。一範例RNN變體為長短期記憶(LSTM；long short term memory) RNN。LSTM RNN係能夠學習長

期相依，其可需要用於處理語言的較長序列。在CNN上的變體為卷積深度信念網路(belief network)，其具有類似於CNN的結構且以類似於深度信念網路的方式來訓練。深度信念網路(DBN；deep belief network)為生成神經網路(generative neural network)，其係由隨機(隨機的(random))變數的多個層所組成。DBN能使用貪婪無監督學習(greedy unsupervised learning)來層對層訓練。接著藉由決定用於神經網路的最佳初始組的權重能使用DBN之學習權重以提供預訓練神經網路。

【0219】圖15闡述深度神經網路的訓練及部署。一旦給定網路已針對任務被結構化，則神經網路係使用訓練資料集1502來訓練。各種訓練框架1504已被發展來致能訓練進程的硬體加速。例如，圖10之機器學習框架1004可組態為訓練框架1004。訓練框架1004能勾連到未訓練的神經網路1506且致能未訓練的神經網來使用於此所述的平行處理資源來訓練以產生訓練的神經網1508。

【0220】為了開始訓練過程，可隨機地或藉由使用深度信念網路預訓練挑選初始權重。接著以監督的或非監督的方式進行訓練周期。

【0221】監督的學習為在其中訓練被進行為居間運算的學習方法，像是當訓練資料集1502包括與用於輸入的所欲輸出搭配的輸入時，或其中訓練資料集包括具有已知輸出的輸入且神經網路之輸出被手動的分等。網路處理輸入且將結果的輸出相對成組預測或所欲的輸出比較。接著透

過系統將誤差反向傳播。訓練框架1504能進行調整以調整控制未訓練的神經網路1506的權重。訓練框架1504能提供工具以監控未訓練的神經網路1506有多佳地正朝向合適於基於已知輸入資料產生正確答案的模型收斂。訓練過程隨著網路的權重被調整以改良由神經網路產生的輸出而重複地發生。訓練過程能持續直到神經網路到達與訓練神經網1508關聯的統計所欲正確性。訓練的神經網路1508接著能被部署以實施任何數目的機器學習運算。

【0222】未監督的學習為在其中網路嘗試使用未標籤的資料訓練其本身的學習方法。因此，對於未監督的學習，訓練資料集1502將包括沒有任何關聯的輸出資料的輸入資料。未訓練的神經網路1506能在未標籤的輸入內學習分組且能決定個別輸入如何關於整體資料集。能使用未監督的訓練以產生自組織的映射，其為能夠進行在降低資料之維數有用的運算之訓練的神經網路1507的類型。亦能使用未監督的訓練來進行異常偵測，其允許在偏離資料之正常圖樣的輸入資料集中資料點的識別。

【0223】亦可運用在監督和未監督訓練上的變異。半監督的學習為在其中於訓練資料集1502中包括相同分布之標籤的及未標籤的資料的混合的技術。遞增學習為監督學習的變體，在其中持續使用輸入資料來進一步訓練模型。遞增學習致能訓練的神經網路1508以在不忘記於初始訓練期間的網路內灌輸的知識下適應新的資料1512。

【0224】無論監督的或未監督的，用於特別深度神經

網路的訓練過程可能對於單一計算節點太過計算密集。取代使用單一計算節點的是，能使用計算節點之分布網路來加速訓練過程。

【0225】圖16為闡述分布式學習的方塊圖。分布式學習為使用多個分布式計算節點以進行監督的或未監督的神經網路之訓練的訓練模型。分布的計算節點各者能包括一或多個主機處理器和通用處理節點之一或多者，像是在圖1100中高度平行通用圖形處理單元1100。如所闡述的，分布式學習能進行模型平行性(model parallelism)1602、資料平行性(data parallelism)1604或模型與資料平行性1604的組合。

【0226】在模型平行性1602中，在分布式系統中不同的計算節點能針對單一網路之不同部分進行訓練計算。例如，神經網路之各個層能由分布式系統之不同處理節點來訓練。模型平行性之益處包括用以定比到特別大的模型的能力。分拆與神經網路之不同層關聯的計算致能非常大神經網路之訓練，在其中所有層之權重不會適配到單一計算節點的記憶體中。在一些實例中，模型平行性在進行大型神經網路之非監督的訓練上能特別的有用。

【0227】在資料平行性1604中，分布式網路之不同節點具有完整的模型之實例，且各個節點接收資料之不同的部分。接著結合來自不同節點的結果。在當對資料平行性不同的方法是可能的同時，資料平行訓練方法皆需要結合練果及同步各個節點之間的模型參數的技術。用以結合資

料的示範性方法包括基於資料平行性平均及更新的參數。參數平均在訓練資料之子集上訓練各個節點且將全域參數(例如，權重、偏移)設定到來自各個節點的參數之平均。參數平均使用維護參數資料的中央參數伺服器。基於資料平行性的更新類似於參數平均，排除了取代將參數從節點傳輸到參數伺服器而將更新傳輸到模型。此外，能以去中心化的方式進行基於資料平行性的更新，其中更新被壓縮且在節點之間傳輸。

【0228】例如在各個計算節點包括多個GPU的分布式系統中，能實施結合的模型和資料平行性1606。各個節點能具有模型的完整實例，以各個節點內分開的GPU被使用來訓練模型的不同部分。

【0229】分布式訓練已增加相對於在單一機器上的訓練之負擔。然而，於此所述的平行處理器及GPGPU各者能實施各種技術來降低分布式訓練的負擔，包括用以致能高頻寬GPU對GPU資料傳輸及加速遠端資料同步的技術。

示範性機器學習應用

【0230】能應用機器學習來解決各種科技問題，包括(但不限於)電腦視覺、自主駕駛及導航、口語辨識以及語言處理。電腦視覺已傳統地為用於機器學習應用的大多數主動研究領域其中一者。電腦視覺之應用範圍從重製人類視覺能力(像是辨識人臉)到創建新的視覺能力之類別。例如，電腦視覺應用能組態以從在視訊中可見的物體中誘發

的振動來辨識聲波。平行處理器加速機器學習致能電腦視覺應用以使用比先前可實行的明顯更大的訓練資料集來訓練且致能推理系統以使用低功率平行處理器來部署。

【0231】平行處理器加速機器學習具有自主駕駛應用，包括巷及道路標誌辨識、障礙物避免、導航及駕駛控制。能使用加速機器學習技術以基於界定對於特定訓練輸入的適當反應的資料集來訓練駕駛模型。於此所述的平行處理器能致能使用於自主駕駛解決的日增的複雜神經網路之快速的訓練且致能益合適於整合到自主車輛中的行動平台中低功率推理處理器的部署。

【0232】平行處理器加速深度網路已致能對自動口語辨識(ASR；automatic speech recognition)的機器學習方法。ASR包括計算大多數可攜語言序列(其給定輸入聲音序列)的功能的創建。使用深度神經網路的加速機器學習已致能先前使用於ASR的隱藏馬可夫模型(HMM；hidden Markov model)和高斯混合模型(GMM；Gaussian mixture model)之取代。

【0233】平行處理器加速機器學習亦能被使用來加速自然語言處理。自動學習程序能利用統計推理演算法來生成對錯誤或不熟悉的輸入是健全的模型。示範性自然語言處理器應用包括在人類語言之間自動機器翻譯。

【0234】使用於機器學習的平行處理平台能被分成訓練平台和部署平台。訓練平台一般為高度平行且包括對加速多GPU單一節點訓練和多節點、多GPU訓練的最佳化。

合適於訓練的示範性平行處理器包括圖1100之高度平行通用圖形處理單元1100和圖1200之多GPU計算系統1200。相反的，部署機器學習平台一般包括適於在產品中使用(像是攝像機、自主機器人以及自主車輛)的較低功率平行處理器。

【0235】圖17闡述適於使用訓練的模型進行推論的示範性推理晶片上系統(SOC)1700。SOC 1700能整合處理組件，包括媒體處理器1702、視覺處理器1704、GPGPU 1706以及多核心處理器1708。SOC 1700能額外的包括晶片上記憶體1705，其能致能共用晶片上資料池，該共用晶片上資料池可由該處理組件之各者存取的。處理組件能針對低功率運算而被最佳化，用以致能對各種機器學習平台的部署，包括自主車輛和自主機器人。例如，能使用SOC 1700之一個實施作為用於自主車輛的主控制系統的一部分。在當SOC 1700係組態用於在自主車輛中使用下，SOC係設計且組態用於符合部署管轄之有關的功能安全標準。

【0236】在運算期間，媒體處理器1702和視覺處理器1704能和諧地工作以加速電腦視覺運算。媒體處理器1702能致能多個高解析(例如，4K、8K)視訊串流的低延遲解碼。解碼的視訊串流能被寫入到在晶片上記憶體1705中的緩衝器。視覺處理器1704接著能剖析解碼的視訊且在準備使用訓練的影像辨識模型處理訊框中在解碼的視訊之訊框上進行初步處理運算。例如，視覺處理器1704能加速用於在高解析視訊資料上進行影像辨識的CNN的卷積運算，同

時後端模型計算係由GPGPU 1706所進行。

【0237】多核心處理器1708能包括控制邏輯，用以協助由媒體處理器1702和視覺處理器1704所進行的資料傳輸及共用記憶體運算的定序和同步。多核心處理器1708亦能作用為應用處理器，用以執行能利用GPGPU 1706之推理計算能力的軟體應用。例如，導航和駕駛邏輯的至少部分能以在多核心處理器1708上執行的軟體來實施。這類軟體能直接發出計算工作量到GPGPU 1706或計算工作量能被發出給多核心處理器1708，其能卸載GPGPU 1706之該些運算的至少部分。

【0238】GPGPU 1706能包括計算叢集，像是在高度平行通用圖形處理單元1100內的計算叢集1106A~1106H之低功率組態。在GPGPU 1706內的計算叢集能支援被具體最佳化以在訓練的神經網路上進行推理計算的指令。例如，GPGPU 1706能支援用以進行低精準度計算的指令，像是8位元和4位元整數向量運算。

系統概觀II

【0239】 圖18為依據實施例的處理系統1800之方塊圖。在各種實施例中，系統1800包括一或多個處理器1802和一或多個圖形處理器1808，並且可為單一處理器桌上型電腦系統、多處理器工作站系統或具有大數量的處理器1802或處理器核心1807的伺服器系統。在一實施例中，系統1800為包含在晶片上系統(SoC；system-on-a-chip)積體

電路內的處理平台，以用於在行動、手持或嵌入式裝置中使用。

【0240】系統1800之實施例能包括(或被包括在其內)基於伺服器的遊戲平台、包括遊戲及媒體控制台的遊戲機(game console)、行動遊戲機(mobile gaming console)、手持遊戲機或線上遊戲機。在一些實施例中，系統1800為行動電話、智慧電話、平板計算裝置或行動網際網路裝置。資料處理系統1800亦能包括(與其耦接或整合在其之內)可穿戴裝置，諸如智慧手錶可穿戴裝置、智慧眼戴式裝置、擴增實境(augmented reality)裝置或虛擬實境裝置。在一些實施例中，資料處理系統1800為電視或機上盒裝置，其具有一或多個處理器1802以及由一或多個圖形處理器1808產生的圖形介面。

【0241】在一些實施例中，該一或多個處理器1802各者包括一或多個處理器核心1807，用以處理指令，其當執行時，進行用於系統及使用者軟體的運算。在一些實施例中，一或多個處理器核心1807係組態以處理特定指令集1809。在一些實施例中，指令集1809可促進複雜指令集計算(CISC；Complex Instruction Set Computing)、精簡指令集計算(RISC；Reduced Instruction Set Computing)或經由超長指令字(VLIW；Very Long Instruction Word)計算。多處理器核心1807各者可處理不同的指令集1809，其可包括用以促進其它指令集之仿真(emulation)的指令。處理器核心1807亦可包括其它處理裝置，像是數位信號處理器

(DSP；Digital Signal Processor)。

【0242】在一些實施例中，處理器1802包括快取記憶體1804。取決於架構，處理器1802能具有單一內部快取或多級的內部快取。在一些實施例中，快取記憶體係共用於處理器1802之各種組件之間。在一些實施例中，處理器1802亦使用外部快取(例如，第3級(L3))快取或最後級快取(LLC；Last Level Cache))(未繪示)，其可共用於使用已知快取一致性(cache coherency)技術的處理器核心1807之間。暫存器檔案1806係額外地被包括在處理器1802內，其可包括用於儲存不同類型資料的不同類型的暫存器(例如，整數暫存器、浮點暫存器、狀態暫存器以及指令指標暫存器)。一些暫存器可為通用暫存器，同時其它暫存器可特定於處理器1802之設計。

【0243】在一些實施例中，處理器1802係耦接至處理器匯流排1810用以傳送通訊信號，通訊信號像是位址、資料或處理器1802與在系統1800中其它組件之間的控制信號。在一實施例中，系統1800使用示範性「中樞」系統架構，包括記憶體控制器中樞1816和輸入輸出(IO)控制器中樞1830。記憶體控制器中樞1816促進記憶體裝置與系統1800之其它組件之間的通訊，同時I/O控制器中樞(ICH；I/O Controller Hub)1830提供經由局部I/O匯流排對I/O裝置的連接。在一實施例中，記憶體控制器中樞1816之邏輯係整合在處理器內。

【0244】記憶體裝置1820可以為動態隨機存取記憶體

(DRAM)裝置、靜態隨機存取記憶體(SRAM)裝置、快閃記憶體裝置、相變記憶體裝置或具有合適效能以作為處理記憶體的一些其它記憶體裝置。在一實施例中，記憶體裝置1820能操作為用於系統1800的系統記憶體，用以儲存資料1822及指令1821，其用於當一或多個處理器1802執行應用或進程時使用。記憶體控制器中樞1816亦與可選外部圖形處理器1812耦接，其可與在處理器1802中一或多個圖形處理器1808通訊，以進行圖形和媒體運算。

【0245】在一些實施例中，ICH 1830使周邊能經由高速I/O匯流排連接到記憶體裝置1820和處理器1802。I/O周邊包括(但不限於)音訊控制器1846、韌體介面1828、無線收發器1826(例如，WiFi、藍牙)、資料儲存裝置1824(例如，硬碟驅動、快閃記憶體等)以及用於將舊有(例如，個人系統2(PS/2))裝置耦接至系統的舊有I/O控制器1840。一或多個通用串列匯流排(USB)控制器1842連接輸入裝置，像是鍵盤及滑鼠1844組合。網路控制器1834亦可耦接至ICH 1830。在一些實施例中，高效能網路控制器(未繪示)耦接至處理器匯流排1810。將理解的是，繪示的系統1800為示範性的且非限制的，同時亦可使用不同組態的其它類型資料處理系統。例如，I/O控制器中樞1830可在一或多個處理器1802內整合，或記憶體控制器中樞1816和I/O控制器中樞1830可整合到獨立外部圖形處理器，像是外部圖形處理器1812。

【0246】圖19為具有一或多個處理器核心1902A~

1902N、積體記憶體控制器1914以及積體圖形處理器1908之處理器1900之實施例的方塊圖。於此具有與任其它圖之元件相同參考號碼(或名稱)的圖19之該些元件能以類似於在此其它處說明之該者的任何方式來操作或作用，但不限於這樣。處理器1900能包括多達且包括由虛線框代表的額外核心1902N的額外的核心。處理器核心1902A~1902N之各者包括一或多個內部快取單元1904A~1904N。在一些實施例中，各個處理器核心亦具有對一或多個共用快取單元1906的存取。

【0247】內部快取單元1904A~1904N和共用快取單元1906代表在處理器1900內的快取記憶體分層(cache memory hierarchy)。快取記憶體分層可包括在各個處理器核心內的至少一級指令及資料快取和一或多級的共用中級快取，像是2級(L2)、3級(L3)、4級(L4)或其它級的快取，其中在外部記憶體被分類為LLC之前的最高級的快取。在一些實施例中，快取一致性邏輯維持各種快取單元1906及1904A~1904N之間的一致性。

【0248】在一些實施例中，處理器1900亦可包括成組的一或多個匯流排控制器單元1916及系統代理核心1910。一或多個匯流排控制單元1916管理成組的周邊匯流排，像是一或多個周邊組件互連匯流排(例如，PCI、快速PCI(PCI Express))。系統代理核心1910提供用於各種處理器組件的管理功能特性(management functionality)。在一些實施例中，系統代理核心1910包括一或多個積體記憶體控

制器1914，用以管理對各種外部記憶體裝置(未繪示)的存取。

【0249】在一些實施例中，處理器核心1902A~1902N之一或多者包括對於同時多執行緒(simultaneous multi-threading)的支援。在這類實施例中，系統代理核心1910包括用於在多緒處理(multi-threaded)期間協調及操作核心1902A~1902N的組件。系統代理核心1910可額外地包括電力控制單元(PCU；power control unit)，其包括用以調節處理器核心1902A~1902N和圖形處理器1908之電源狀態的邏輯和組件。

【0250】在一些實施例中，處理器1900額外地包括圖形處理器1908，用以執行圖形處理運算。在一些實施例中，圖形處理器1908與成組的共用快取單元1906和系統代理核心1910耦接，其包括一或多個積體記憶體控制器1914。在一些實施例中，顯示控制器1911與圖形處理器1908耦接，用以驅動圖形處理器輸出到一或多個耦接的顯示器。在一些實施例中，顯示控制器1911可為經由至少一互連與圖形處理器耦接之分開的模組，或可整合在圖形處理器1908或系統代理核心1910內。

【0251】在一些實施例中，使用基於環狀的互連單元1912來耦接處理器1900之內部組件。然而，可使用替代的互連單元，像是點對點互連、切換互連(switched interconnect)或其它技術，包括在本領域中周知的技術。在一些實施例中，圖形處理器1908經由I/O連結1913與環

狀互連1912耦接。

【0252】示範性I/O連結1913代表多個各式各樣的I/O互連之至少一者，包括封裝上I/O互連(on package I/O interconnect)，其促進各種處理器組件和高效能嵌入式記憶體模組1918之間的通訊，像是eDRAM模組。在一些實施例，處理器核心1902~1902N之各者和圖形處理器1908使用嵌入式記憶體模組1918作為共用最後級快取。

【0253】在一些實施例中，處理器核心1902A~1902N為執行相同指令集架構的同質核心。在另一實施例中，處理器核心1902A~1902N按照指令集架構(ISA；instruction set architecture)為異質的，其中處理器核心1902A~N之一或多者執行第一指令集，同時其它核心之至少一者執行第一指令集或不同指令集之子集。在一實施例中，處理器核心1902A~1902N按照微架構為異質的，其中具有相對較高功率消耗的一或多個核心與具有較低功率消耗的一或多個功率核心耦接。此外，處理器1900能在一或多個晶片上實施或實施為除了其它組件以外具有闡述的組件的SoC積體電路。

【0254】圖20為可為圖形處理器2000之方塊圖，其可為獨立圖形處理單元或為與複數個處理核心整合的圖形處理器。在一些實施例中，圖形處理器經由記憶體映射的I/O介面來對在圖形處理器上的暫存器通訊且具有放置到處理器記憶體中的命令。在一些實施例中，圖形處理器2000包括用以存取記憶體的記憶體介面2014。記憶體介面

2014能為對本地記憶體、一或多個內部快取、一或多個共用外部快取及/或系統記憶體的介面。

【0255】在一些實施例中，圖形處理器2000亦包括顯示控制器2002，用以將顯示輸出資料驅動至顯示裝置2020。顯示控制器2002包括用於一或多個重疊平面的硬體以用於顯示以及包括多層的視訊或使用者介面元件的成分。在一些實施例中，圖形處理器2000包括視訊編解碼器引擎2006，用以從或在一或多個媒體編碼格式之間將媒體編碼、解碼或轉碼(transcode)，媒體編碼格式包括(但不限於)像是MPEG-2的動態圖碼專家群(MPEG；Moving Picture Experts Group)格式、像是H.264/MPEG-4 AVC的先進視訊編碼(AVC；Advanced Video Coding)格式以及像是JPEG的動畫&電視工程社群(SMPTE；Society of Motion Picture & Television Engineers)421M/VC-1及聯合圖形專家群(JPEG；Joint Photographic Experts Group)格式和動態JPEG(MJPEG；Motion JPEG)格式。

【0256】在一些實施例中，圖形處理器2000包括區塊影像傳輸(BLIT；block image transfer)引擎2004，用以進行二維(2D)柵格器(rasterizer)運算，例如包括位元邊界區域傳輸(bit-boundary block transfer)。然而，在一實施例中，使用圖形處理引擎(GEP；graphics processing engine)2010之一或多個組件來進行2D圖形運算。在一些實施例中，圖形處理引擎2010為用於進行圖形運算的計算引擎，包括三維(3D)圖形運算及媒體運算。

【0257】在一些實施例中，GPE 2010包括用於進行3D運算的3D管線2012，像是使用作用在3D基本形狀(primitive shape)(例如，矩形、三角形等)上的處理函數來算繪三維影像及場景。3D管線2012包括在元件內進行各種任務及/或產出(spawn)執行的執行緒給3D/媒體子系統2015的可程式及固定功能元件。當能使用3D管線2012以進行媒體運算時，GPE 2010之實施例亦包括媒體管線2016，其具體使用來進行媒體運算，像是視訊後處理(video post-processing)和影像增強。

【0258】在一些實施例中，媒體管線2016包括固定功能或可程式邏輯單元，用以進行一或多個專門的媒體運算，像是視訊解碼加速、視訊解交錯(de-interlacing)以及取代或代表視訊編解碼引擎2006的視訊編碼加速。在一些實施例中，媒體管線2016額外地包括執行緒產出單元，用以產出用於在3D/媒體子系統2015上執行的執行緒。所產出的執行緒對於在被包括在3D/媒體子系統2015中之一或多個圖形執行單元上的媒體運算來進行計算。

【0259】在一些實施例中，3D/媒體子系統2015包括用於執行由3D管線2012及媒體管線2016所產出的執行緒之邏輯。在一實施例中，管線發送執行緒執行請求給3D/媒體子系統2015，其包括執行緒調度邏輯，以用於仲裁及調度各種請求給可用的執行緒執行資源。執行資源包括圖形執行單元之陣列，用以處理3D及媒體執行緒。在一些實施例中，3D/媒體子系統2015包括用於執行緒指令及資料的

一或多個內部快取。在一些實施例中，子系統亦包括共用記憶體，其包括暫存器及可定址記憶體，用以在執行緒之間共用資料且用以儲存輸出資料。

3D/媒體處理

【0260】圖21為依據一些實施例的圖形處理器之圖形處理引擎2110的方塊圖。在一實施例中，圖形處理引擎(GPE)2110為在圖20所繪示的GPE 2010之一版本。於此具有與任其它圖之元件相同參考號碼(或名稱)的圖21之元件能以類似於在此其它處說明之該者的任何方式來操作或作用，但不限於這樣。例如，闡述了圖20的3D管線2012和媒體管線2016。媒體管線2016在GPE 2110的一些實施例中是可選的而可能不會明確地包括在GPE 2110內。例如，且在至少一實施例中，分開的媒體及/或影像處理器係耦接至GPE 2110。

【0261】在一些實施例中，GPE 2110與命令串流器(command streamer)2103耦接或包括命令串流器2103，其提供命令串流給3D管線2012及/或媒體管線2016。在一些實施例中，命令串流器2103係與記憶體耦接，其可以為系統記憶體或是內部快取記憶體和共用快取記憶體之一或多者。在一些實施例中，命令串流器2103從記憶體接收命令且將命令發送至3D管線2012及/或媒體管線2016。命令為從環狀緩衝器取出的指示(directive)，環狀緩衝器儲存用於3D管線2012及媒體管線2016的命令。在一實施例中，環

狀緩衝器可以額外地包括批次命令(batch command)緩衝器，其儲存多個命令之批次。用於3D管線2012的命令亦能包括對儲存在記憶體中之資料的參考，像是(但不限於)用於3D管線2012的頂點及幾何資料及/或用於媒體管線2016的影像資料及記憶體物件。3D管線2012及媒體管線2016藉由經由在分別管線內的邏輯進行運算或藉由調度一或多個執行執行緒給圖形核心陣列2114來處理命令。

【0262】在各種實施例中，3D管線2012能藉由處理指令及調度執行執行緒給圖形核心陣列2114來執行一或多個著色器程式，著色器程式像是頂點著色器、幾何著色器、像素著色器、片段著色器、計算著色器或其它著色器程式。圖形核心陣列2114提供執行資源之統合方塊。在圖形核心陣列2114內多用途執行邏輯(例如，執行單元)包括對於各種3D API著色器語言的支援且能執行與多個著色器關聯的多個同時執行執行緒。

【0263】在一些實施例中，圖形核心陣列2114亦包括執行邏輯以進行媒體功能，像是視訊及/或影像處理。在一實施例中，執行單元額外地包括通用邏輯，其可程式化以除了圖形處理運算以外，進行平行通用計算運算。通用邏輯能並行地或連同在圖18之處理器核心1807或如在圖19中的核心1902A~1902N內的通用邏輯來進行處理運算。

【0264】由在圖形核心陣列2114上的執行緒執行產生的輸出資料能將資料輸出給在統合回傳緩衝器(URB；unified return buffer)2118中的記憶體。URB 2118能儲存

用於多個執行緒的資料。在一些實施例中，可使用URB 2118在圖形核心陣列2114上執行的不同執行緒之間發送資料。在一些實施例中，URB 2118可額外地被使用於在圖形核心陣列與在共用功能邏輯2120內之固定功能邏輯上的的執行緒之間的同步。

【0265】在一些實施例中，圖形核心陣列2114為可擴充的，使得陣列包括可變數目的圖形核心，各者基於GPE 2110之目標功率和效能等級具有可變數目的執行單元。在一實施例中，執行資源係動態可擴充的，使得執行資源能如所需要的被致能或禁能。

【0266】圖形核心陣列2114與共用功能邏輯2120耦接，該共用功能邏輯包括在圖形核心陣列中的圖形核心之間共用的多個資源。在共用功能邏輯2120內的共用功能為提供專門補充功能性給圖形核心陣列2114的硬體邏輯單元。在各種實施例中，共用功能邏輯2120包括(但不限於)取樣器2121、數學2122和執行緒間通訊(ITC；inter-thread communication)2123邏輯。此外，一些實施例在共用功能邏輯2120內實施一或多個快取2125。在當對於給定專門功能的需求不足以用於包括在圖形核心陣列2114內下實施共用功能。相反的，該專門功能的例示被實施為在共用功能邏輯2120中的單獨實體且在圖形核心陣列2114內的執行資源之間共用。在圖形核心陣列2114之間共用且被包括在圖形核心陣列2114內的精準的功能組係在實施例之間變化。

【0267】圖22為圖形處理器2200之另一實施例的方塊

圖。於此具有與任其它圖之元件相同參考號碼(或名稱)的圖22之元件能以類似於在此其它處說明之該者的任何方式來操作或作用，但不限於這樣。

【0268】在一些實施例中，圖形處理器2200包括環狀互連2202、管線前端2204、媒體引擎2237以及圖形核心2280A~2280N。在一些實施例中，環狀互連2202將圖形處理器耦接至其它處理單元，包括其它圖形處理器或一或多個通用處理器核心。在一些實施例中，圖形處理器為在多核心處理系統內整合的許多處理器之一者。

【0269】在一些實施例中，圖形處理器2200經由環狀互連2202接收命令之批次。到來的命令係由在管線前端2204中的命令串流器2203來解譯。在一些實施例中，圖形處理器2200包括可擴充執行邏輯，用以經由圖形核心2280A~2280N進行3D幾何處理及媒體處理。對於3D幾何處理命令，命令串流器2203供應命令給幾何管線2236。對於至少一些媒體處理命令，命令串流器2203供應命令給視訊前端2234，其與媒體引擎2237耦接。在一些實施例中，媒體引擎2237包括視訊品質引擎(VQE；Video Quality Engine)2230，以用於視訊及影像後處理，且包括多格式編碼/解碼(MFX；multi-format encode/decode)2233引擎，用以提供硬體加速的媒體資料編碼及解碼。在一些實施例中，幾何管線2236和媒體引擎2237各者產生執行執行緒，以用於由至少一圖形核心2280A提供的執行緒執行資源。

【0270】在一些實施例中，圖形處理器2200包括可擴

充執行緒執行資源，其採用模組化核心2280A~2280N(有時稱為核心片層(core slice))，各者具有多個子核心2250A~2250N、2260A~2260N(有時稱為核心子片層(core sub-slice))。在一些實施例中，圖形處理器2200可以具有任何數目的圖形核心2280A到2280N。在一些實施例中，圖形處理器2200包括圖形核心2280A，其具有至少第一子核心2250A及第二核心子核心2260A。在其它實施例中，圖形處理器為具有單一子核心(例如，2250A)的低功率處理器。在一些實施例中，圖形處理器2200包括多個圖形核心2280A~2280N，各者包括一組第一子核心2250A~2250N和一組第二子核心2260A~2260N。在該組第一子核心2250A~2250N中各個子核心包括至少第一組執行單元2252A~2252N和媒體/紋理取樣器2254A~2254N。在該組第二子核心2260A~2260N中各個子核心包括至少第二組執行單元2262A~2262N和取樣器2264A~2264N。在一些實施例中，各個子核心2250A~2250N、2260A~2260N共用一組共用資源2270A~2270N。在一些實施例中，共用資源包括共用快取記憶體和像素運算邏輯。其它共用資源亦被包括在圖形處理器之各種實施例中。

執行邏輯

【0271】圖23闡述包括在GPE之一些實施例中運用的處理元件之陣列的執行緒執行邏輯2300。於此具有與任其它圖之元件相同參考號碼(或名稱)的圖23之元件能以類似

於在此其它處說明之該者的任何方式來操作或作用，但不限於這樣。

【0272】在一些實施例中，執行緒執行邏輯2300包括像素著色器2302、執行緒調度器2304、指令快取2306、包括複數個執行單元2308A~2308N的可擴充執行單元陣列、取樣器2310、資料快取2312以及資料埠2314。在一些實施例中，所包括的組件係經由連結到組件之各者的互連組構(interconnect fabric)來進行互連。在一些實施例中，執行緒執行邏輯2300包括透過一或多個指令快取2306對記憶體的一或多個連接(像是系統記憶體或快取記憶體)、資料埠2314、取樣器2310以及執行單元陣列2308A~2308N。在一些實施例中，各個執行單元(例如，2308A)為能夠執行多個同時執行緒及對於各執行緒並行處理多個資料元件的個別向量處理器。在一些實施例中，執行單元陣列2308A~2308N包括任何數目個別執行單元。

【0273】在一些實施例中，主要使用執行單元陣列2308A~2308N以執行「著色器」程式。在一些實施例中，在陣列2308A~2308N中的執行單元執行包括用於許多標準3D圖形著色器指令的原生支援的指令集，使得來自圖形庫(例如，直接3D及OpenGL)的著色器程式以最小的轉譯來執行。執行單元支援頂點(vertex)及幾何處理(例如，頂點程式、幾何程式、頂點著色器)、像素處理(例如，像素著色器、片段著色器(fragment shader))以及通用處理(例如，計算及媒體著色器)。

【0274】在執行單元陣列2308A~2308N中的各個執行單元在資料元件之陣列上運算。資料元件之數目為「執行尺寸」，或是用於指令的通道之數目。執行通道為用於資料元件存取、遮蔽及在指令內流程控制之執行的邏輯單元。通道之數目可無關於用於特定圖形處理器的實體算術邏輯單元(ALU；Arithmetic Logic Unit)或浮點單元(FPU；Floating Point Unit)之數目。在一些實施例中，執行單元2308A~2308N支援整數及浮點資料類型。

【0275】執行單元指令集包括單指令多資料(SIMD；single instruction multiple data)指令或單指令多執行緒(SIMT；single instruction multiple thread)指令。各種資料元件在暫存器中能被儲存為緊縮資料類型(packed data type)並且執行單元將基於元件之資料尺寸來處理各種元件。舉例來說，當在256位元寬的向量上運算時，向量之256位元儲存在暫存器中並且執行單元在向量上運算為4個分開的64位元緊縮資料元件(四倍字(QW；Quad-Word)尺寸資料元件)、8個分開的32位元緊縮資料元件(雙倍字(DW；Double Word)尺寸資料元件)、16個分開的16位元緊縮資料元件(字(W)尺寸資料元件)或是32個分開的8位元資料元件(位元組(B；byte)尺寸資料元件)。然而，不同的向量寬度和暫存器尺寸是可能的。

【0276】一或多個內部指令快取(例如，2306)被包括在執行緒執行邏輯2300中，用以對用於執行單元的執行緒指令進行快取。在一些實施例中，一或多個資料快取(例

如，2312)被包括以在執行緒執行期間對執行緒資料進行快取。在一些實施例中，取樣器2310被包括以提供用於3D運算的紋理取樣以及用於媒體運算的媒體取樣。在一些實施例中，取樣器2310包括專門的紋理或媒體取樣功能特性，用以在提供取樣的資料給執行單元之前於取樣過程期間處理紋理或媒體資料。

【0277】在執行期間，圖形及媒體管線經由執行緒產出及調度邏輯對執行緒執行邏輯2300發送執行緒發起請求。在一些實施例中，執行緒執行邏輯2300包括本地執行緒調度器2304，其仲裁來自圖形及媒體管線的執行緒發起請求，並且在一或多個執行單元2308A~2308N上體現請求的執行緒。舉例來說，幾何管線(例如，圖22之2236)調度頂點處理、曲面細分(tessellation)或幾何處理執行緒給執行緒執行邏輯2300(圖23)。在一些實施例中，執行緒調度器2304亦能處理來自執行著色器程式的運行時刻執行緒產出請求(runtime thread spawning request)。

【0278】一旦幾何物件之群組已被處理且被柵格化成像素資料，則呼用像素著色器2302以進一步計算輸出資訊並造成結果被寫入到輸出表面(例如，色彩緩衝器、深度緩衝器、模板(stencil)緩衝器等)。在一些實施例中，像素著色器2302計算各種頂點屬性之值，其係用以跨柵格化的物件來被內插。在一些實施例中，像素著色器2302接著執行應用程式介面(API；application programming interface)-供應的像素著色器程式。為了執行像素著色器程式，像素

著色器2302經由執行緒調度器2304調度執行緒給執行單元(例如，2308A)。在一些實施例中，像素著色器2302使用在取樣器2310中的紋理取樣邏輯來在記憶體中儲存的紋理圖(texture map)中存取紋理資料。在紋理資料和輸入幾何資料上的算術運算計算用於各個幾何片段的像素色彩資料或從進一步處理丟棄一或多個像素。

【0279】在一些實施例中，資料埠2314提供記憶體存取機制以為了執行緒執行邏輯2300輸出處理的資料給記憶體以用於在圖形處理器輸出管線上處理。在一些實施例中，資料埠2314包括或耦接到一或多個快取記憶體(例如，資料快取2312)以經由資料埠對用於記憶體存取的資料進行快取。

【0280】圖24為依據一些實施例闡述圖形處理器指令格式2400的方塊圖。在一或多個實施例中，圖形處理器執行單元支援具有在多個格式上之指令的指令集。實線框闡述一般被包括在執行單元指令中的組件，同時虛線包括可選或僅被包括在指令之子集中的組件。在一些實施例中，說明及闡述的指令格式2400為巨集指令(macro-instruction)，因為他們為供應給執行單元的指令，其相對於一旦處理指令時從指令解碼造成的微運算(micro-operation)。

【0281】在一些實施例中，圖形處理器執行單元原生地支援在128位元指令格式2410中的指令。64位元壓縮指令格式2430可用於基於選定指令的一些指令、指令選項以及眾多的運算元。原生128位元指令格式2410提供對所有

指令選項的存取，同時一些選項和運算被限制在64位元指令格式2430中。在64位元指令格式2430中可用的原生指令隨實施例變化。在一些實施例中，使用在索引欄位2413中之一組索引值部分地壓縮指令。執行單元硬體基於索引值參考一組壓縮表且使用壓縮表輸出以在128位元指令格式2410中重建原生指令。

【0282】對於各個格式，指令運算碼2412界定執行單元要用以進行的運算。執行單元並行地執行跨各個運算元之多個資料元件的各個指令。例如，反應於相加指令，執行單元進行跨代表紋理元件或圖片元件之各個色彩通道的同時相加運算。藉由預設，執行單元進行跨運算元之所有資料通道的各個指令。在一些實施例中，指令控制欄位2414在某些執行選項之上致能控制，像是通道選擇(例如，預測)和資料通道次序(例如，拌和(swizzle))。對於128位元指令2410，執行尺寸(exec-size)欄位2416限定將被並行地執行的資料通道的數目。在一些實施例中，執行尺寸欄位2416並非可用於在64位元壓縮指令格式2430中使用。

【0283】一些執行單元指令具有至多三個運算元，其包括兩個來源運算元src0 2420、src1 2422和一個目的地2418。在一些實施例中，執行單元支援雙目的地指令，其中隱含了目的地之其中之一。資料操控指令(manipulation instruction)能具有第三來源運算元(例如，SRC2 2424)，其中指令運算碼2412決定來源運算元之數目。指令的最後

來源運算源可以為以指令傳遞的立即(例如，硬編碼的(hard-coded))值。

【0284】在一些實施例中，128位元指令格式2410包括存取/位址模式資訊2426，其例如明定直接暫存器定址模式或間接暫存器定址模式是否被使用。當使用直接暫存器定址模式時，一或多個運算元之暫存器位址係直接地由在指令2410中的位元所提供。

【0285】在一些實施例中，128位元指令格式2410包括存取/位址模式欄位2426，其明定用於指令的位址模式及/或存取模式。在一實施例中，存取模式用以界定對於指令的資料存取對齊(data access alignment)。一些實施例支援包括16位元組對齊存取模式和1位元組對齊存取模式的存取模式，其中存取模式之位元組對齊決定指令運算元之存取對齊。例如，當在第一模式中時，指令2410可對來源及目的地運算元使用位元組對齊定址，且當在第二模式中時，指令2410可對所有來源及目的地運算元使用16位元組對齊的定址。

【0286】在一實施例中，存取/位址模式欄位2426之位址模式選項決定指令是否要用以使用直接或間接定址。當使用直接暫存器定址模式時，在指令2410中的位元直接提供一或多個運算元之暫存器位址。當使用直接暫存器定址模式時，一或多個運算元之暫存器位址可基於位址暫存器值和在指令中位址立即欄位來計算。

【0287】在一些實施例中，指令係基於運算碼2412位

元欄位來分群組用以簡化運算碼解碼2440。對於8位元運算碼，位元4、5及6允許執行單元決定運算碼的類型。此顯示的精確運算碼分群組僅為範例。在一些實施例中，移動和邏輯運算碼群組2442包括資料移動及邏輯指令(例如，移動(mov)、計算(cmp))。在一些實施例中，移動和邏輯群組2442共用5個最高有效位元(MSB；most significant bit)，其中移動(mov)指令係以0000xxxxb之形式並且邏輯指令係以0001xxxxb之形式。流程控制指令群組2444(例如，呼叫、跳躍(jmp))包括以0010xxxxb之形式(例如，0x20)的指令。雜項指令群組2446包括指令之混合，其包括以0011xxxxb之形式(例如，0x30)的同步指令(例如，等待、發送)。平行數學指令群組2448包括以0100xxxxb(例如，0x40)之形式的分量方式(component-wise)算術指令(例如，加、乘(mul))。平行數學群組2448跨資料通道並行地進行算術運算。向量數學群組2450包括以0101xxxxb(例如，0x50)之形式的算術指令(例如，dp4)。向量數學群組進行像是在向量運算元上的內積(dot product)計算。

圖形管線

【0288】圖25為圖形處理器2500之另一實施例的方塊圖。於此具有與任其它圖之元件相同參考號碼(或名稱)的圖25之元件能以類似於在此其它處說明之該者的任何方式來操作或作用，但不限於這樣。

【0289】在一些實施例中，圖形處理器2500包括圖形管線2520、媒體管線2530、顯示引擎2540、執行緒執行邏輯2550以及算繪輸出管線2570。在一些實施例中，圖形處理器2500為在包括一或多個通用處理核心的多核心處理系統內的圖形處理器。圖形處理器藉由到一或多個控制暫存器(未繪示)的暫存器寫入或經由發出到圖形處理器2500(其經由環形互連2502)的命令來控制。在一些實施例中，環形互連2502將圖形處理器2500耦接到其它處理組件，像是其它圖形處理器或通用處理器。來自環形互連2502的命令係由命令串流2503來解譯，其將指令供給到圖形管線2520或媒體管線2530之個別組件。

【0290】在一些實施例中，命令串流2503引導頂點抓取器(vertex fetcher)2505之運算，其從記憶體讀取頂點資料且執行由命令串流器2503提供的頂點處理命令。在一些實施例中，頂點抓取器2505提供頂點資料給頂點著色器2507，其對各個頂點進行座標空間轉換及照明運算(lighting operation)。在一些實施例中，頂點抓取器2505和頂點著色器2507藉由將執行執行緒經由執行緒分派器2531分派給執行單元2552A、2552B來執行頂點處理指令。

【0291】在一些實施例中，執行單元2552A、2552B為具有用於進行圖形及媒體運算之指令集的向量處理器之陣列。在一些實施例中，執行單元2552A、2552B具有附加的L1快取2551，其特定用於各個陣列或陣列間的共用。

快取可以組態為資料快取、指令快取或分割成包含在不同分割中的資料及指令的單一快取。

【0292】在一些實施例中，圖形管線2520包括曲面細分組件，用以進行3D物件之硬體加速曲面細分。在一些實施例中，可程式外殼著色器(hull shader)2511組態該曲面細分運算。可程式域著色器2517提供曲面細分輸出之後端評估。曲面細分器2513在外殼著色器2511之方向上運算並且包含專用邏輯以基於提供為對圖形管線2520之輸入的粗略幾何模型產生一組詳細的幾何物件。在一些實施例中，若未使用曲面細分，則可以繞過曲面細分組件2511、2513、2517。

【0293】在一些實施例中，完整幾何物件能由經由分派到執行單元2552A、2552B的一或多個執行緒的幾何著色器2519處理或是直接行進到剪取器(clipper)2529。在一些實施例中，幾何著色器在整個幾何物件上運算，而不是如在圖形管線之先前階段中的頂點或頂點之補丁。若禁能曲面細分，則幾何著色器2519從頂點著色器2507接收輸入。在一些實施例中，幾何著色器2519可藉由幾何著色器程式而可程式化，用以若禁能曲面細分單元則進行幾何曲面細分。

【0294】在柵格化之前，剪取器2529處理頂點資料。剪取器2529可為固定功能剪取器或具有剪取及幾何著色器功能的可程式剪取器。在一些實施例中，在算繪輸出管線2570中的柵格器及深度測試組件2573調度像素著色器以將

幾何物件轉換成他們的每像素表示。在一些實施例中，像素著色器邏輯被包括在執行緒執行邏輯2550中。在一些實施例中，應用能繞過柵格化且經由流出單元2523存取未柵格化的頂點資料。

【0295】圖形處理器2500具有互連匯流排、互連組構或允許資料及訊息在處理器之主要組件之間通過的一些其它互連機制。在一些處理器中，執行單元2552A、2552B及關聯的快取2551、紋理及媒體取樣器2554以及紋理/取樣器快取2558經由資料埠2556互連以進行記憶體存取且與處理器之算繪輸出管線組件通訊。在一些實施例中，取樣器2554、快取2551、2558以及執行單元2552A、2552B各者具有分開的記憶體存取路徑。

【0296】在一些實施例中，算繪輸出管線2570包含柵格器及深度測試組件2573，其將基於頂點的物件(vertex-based object)轉換成關聯的基於像素的表示(pixel-based representation)。在一些實施例中，算繪輸出管線2570包括視窗器/遮蔽器單元，用以進行固定功能的三角及線柵格化。關聯的算繪快取2578和深度快取2579亦在一些實施例中是可利用的。像素運算組件2577在資料上進行基於像素的運算，雖然在一些實例中，與2D運算關聯的像素運算(例如，以混色(blending)來傳輸位元區塊影像)係藉由2D引擎2541來進行，或是使用重疊顯示平面藉由顯示控制器2543來在顯示時間取代。在一些實施例中，共用L3快取2575對所有圖形組件是可用的，其允許不使用主系統記憶

體下共用資料。

【0297】在一些實施例中，圖形處理器媒體管線2530包括媒體引擎2537和視訊前端2534。在一些實施例中，視訊前端2534從命令串流器2503接收管線命令。在一些實施例中，媒體管線2530包括分開的命令串流器。在一些實施例中，在發送命令到媒體引擎2537之前，視訊前端2534處理媒體命令。在一些實施例中，媒體引擎2537包括執行緒產出功能特性，用以產出用於經由執行緒調度器2531調度到執行緒執行邏輯2550的執行緒。

【0298】在一些實施例中，圖形處理器2500包括顯示引擎2540。在一些實施例中，顯示引擎2540在處理器2500外部且經由環形互連2502或一些其它互連匯流排或組構來與圖形處理器耦接。在一些實施例中，顯示引擎2540包括2D引擎2541和顯示控制器2543。在一些實施例中，顯示引擎2540包含特別目的邏輯，其能夠獨立地運算3D管線。在一些實施例中，顯示控制器2543與顯示裝置(未繪示)耦接，其可為系統積體顯示裝置，如在膝上型電腦中的或經由顯示裝置連接器附接的外部顯示裝置。

【0299】在一些實施例中，圖形管線2520和媒體管線2530可組態以基於多個圖形及媒體程式介面來進行運算，並且非特定於任何一個應用程式介面(API)。在一些實施例中，用於圖形處理器的驅動器軟體將特定於特別圖形或媒體庫的API呼叫轉譯成能由圖形處理器處理的命令。在一些實施例中，對於來自科納斯組織(Khronos Group)的開

放圖形庫(OpenGL；Open Graphics Library)及開放計算語言(OpenCL；Open Computing Language)、來自微軟公司的直接3D庫(Direct3D library)提供支援，或可對OpenGL及D3D兩者提供支援。亦可對於開放來源電腦視覺庫(OpenCV；Open Source Computer Vision Library)提供支援。若可從未來API之管線對圖形處理器之管線作成映射，則亦會支援具有相容3D管線的未來API。

圖形管線程式化

【0300】圖26A為依據一些實施例闡述圖形處理器指令格式2600的方塊圖。圖26B為依據實施例闡述圖形處理器命令序列2610的方塊圖。在圖26A中的實線框闡述一般被包括在圖形命令中的組件，同時虛線包括可選或僅被包括在圖形命令之子集中的組件。圖26A之示範性圖形處理器命令格式2600包括資料欄位，用以識別命令之目標客戶端2602、命令運算碼(opcode)2604以及用於命令的相關資料2606。子運算碼2605及命令尺寸2608亦包括在一些命令中。

【0301】在一些實施例中，客戶端2602明定處理命令資料的圖形裝置之客戶端單元。在一些實施例中，圖形處理器命令剖析器檢驗各個命令的客戶端欄位，用以調適命令之未來處理以及路由命令資料到適當的客戶端單元。在一些實施例中，圖形處理器客戶端單元包括記憶體介面單元、算繪單元、2D單元、3D單元以及媒體單元。各個客

戶端單元具有處理命令的相應處理管線。一旦由客戶端單元接收命令時，客戶端單元讀取運算碼2604，且若出現，子運算碼2605用以決定要進行的運算。客戶端單元使用在資料欄位2606中的資訊來進行命令。對於一些命令，預期明確的命令尺寸2608以明定命令的尺寸。在一些實施例中，命令剖析器基於命令運算碼自動地決定命令之至少一些的尺寸。在一些實施例中，命令係經由多個雙倍字來對齊。

【0302】在圖26B中的流程圖繪示示範性圖形處理器命令序列2610。在一些實施例中，採用圖形處理器之實施例的資料處理系統之軟體或韌體使用繪示以安裝、執行及終止成組的圖形運算的命令序列之版本。僅為了範例的目的，繪示及說明樣本命令序列，同時實施例並不限於這些特定命令或不限於此命令序列。再者，命令可發出為在命令序列中命令之批次，使得圖形處理器將在至少部分同時發生中處理命令之序列。

【0303】在一些實施例中，圖形處理器命令序列2610可以管線清洗(pipeline flush)命令2612開始來引起任何主動圖形管線完成對於管線同時擱置。在一些實施例中，3D管線2622和媒體管線2624並非同時地運算。進行管線清洗以引起主動圖形管線完成任何擱置命令。反應於管線清洗，用於圖形處理器的命令剖析器將暫停命令處理，直到主動繪圖引擎完成擱置運算且使相關讀取快取無效。可選地，可以對記憶體清洗標記為「髒(dirty)」之在算繪快取

中的資料。在一些實施例中，對於管線同步或在將圖形處理器放置到低功率狀態中之前，能使用管線清洗命令2612。

【0304】在一些實施例中，當命令序列需要圖形處理器明確地在管線之間切換時，使用管線選擇命令2613。在一些實施例中，在發出管線命令之前於執行環境(上下文)(execution context)內僅需要一次管線選擇命令2613，除非環境係用以對於兩者的管線皆發出命令。在一些實施例中，立即在經由管線選擇命令2613的管線切換之前需要管線清洗命令2612。

【0305】在一些實施例中，管線控制命令2614組態用於運算的圖形管線且使用以程式化3D管線2622和媒體管線2624。在一些實施例中，管線控制命令2614組態用於主動管線的管線狀態。在一實施例中，管線控制命令2614係使用於管線同步且用以在處理命令之批次之前於主動管線內從一或多個快取記憶體清除資料。

【0306】在一些實施例中，使用用於回傳緩衝器狀態2616的命令以組態成組的回傳緩衝器以用於分別的管線來寫入資料。一些管線運算需要一或多個回傳緩衝器之分配、選擇或組態，於處理期間運算將立即資料寫到該一或多個回傳緩衝器中。在一些實施例中，圖形處理器亦使用一或多個回傳緩衝器用以儲存輸出資料且用以進行跨執行緒通訊。在一些實施例中，組態回傳緩衝器狀態2616包括選定尺寸及回傳緩衝器之數量以使用於成組的管線運算。

【0307】在命令序列中剩餘的命令係基於用於運算的主動管線而不同。基於管線決定2620，命令序列係對以3D管線狀態2630開始的3D管線2622或以媒體管線狀態2640開始的媒體管線2624量身定制。

【0308】用於3D管線狀態2630的命令包括用於頂點緩衝器狀態、頂點元件狀態、恆定色彩狀態、深度緩衝器狀態以及用以在處理3D基原(primitive)命令之前被組態的其它狀態變數的3D狀態設定命令。這些命令之值係至少部分基於使用中的特別3D API來決定。在一些實施例中，3D管線狀態2630命令亦能選擇性地禁能或繞過某些管線元件，若將不會使用該些元件的話。

【0309】在一些實施例中，使用3D基原2632命令以提交用以由3D管線處理的3D基原。經由3D基原2632命令傳遞到圖形處理器的命令及關聯的參數被轉發到在圖形管線中的頂點抓取功能。頂點抓取功能使用3D基原2632命令資料以產生頂點資料結構。頂點資料結構係儲存在一或多個回傳緩衝器中。在一些實施例中，使用3D基原2632命令以經由頂點著色器在3D基原上進行頂點運算。為了處理頂點著色器，3D管線2622調度著色器執行執行緒給圖形處理器執行單元。

【0310】在一些實施例中，3D管線2622係經由執行2634命令或事件來觸發。在一些實施例中，暫存器寫入觸發命令執行。在一些實施例中，經由在命令序列中的「去(go)」或「踢(kick)」命令來觸發。在一實施例中，使用

管線同步來觸發命令執行以清洗通過圖形管線的命令序列。3D管線將進行用於3D基原的幾何處理。一旦完成運算，造成的幾何物件被柵格化且像素引擎上色造成的像素。亦可包括用以控制像素著色及像素後端運算之額外的命令以用於該些運算。

【0311】在一些實施例中，當進行媒體運算時，圖形處理器命令序列2610跟隨媒體管線2624。一般而言，對於媒體管線2624程式化的特定使用及方式取決於要用以進行的媒體或計算運算。在媒體解碼期間可對媒體管線卸載特定媒體解碼運算。在一些實施例中，亦能繞過媒體管線並且能使用由一或多個通用處理核心所提供的資源整體或部分地進行媒體解碼。在一實施例中，媒體管線亦包括用於通用圖形處理器單元(GPGPU；general-purpose graphics processor unit)運算的元件，其中使用圖形處理器以使用沒有明確關於圖形基原之算繪的計算著色器程式進行SIMD向量運算。

【0312】在一些實施例中，媒體管線2624係如3D管線2622的類似方式來組態。在媒體物件命令2642之前，用以組態媒體管線狀態2640的成組的命令被調度或放置到命令佇列中。在一些實施例中，用於媒體管線狀態2640的命令包括用以組態將被使用以處理媒體物件之媒體管線元件的資料。此包括用以在媒體管線內組態視訊解碼及視訊編碼邏輯的資料，像是編碼或解碼格式。在一些實施例中，用於媒體管線狀態的命令2640亦支援使用對包含狀態設定之

批次的「間接」狀態元件之一或多個指標。

【0313】在一些實施例中，媒體物件命令2642將指標供應給媒體物件以用於藉由媒體管線來處理。媒體物件包括記憶體緩衝器，其包含用以被處理的視訊資料。在一些實施例中，在發出媒體物件命令2642之前，所有媒體管線狀態必需是有效的。一旦組態管線狀態且媒體物件命令2642排隊時，媒體管線2624係經由執行命令2644或等效執行事件(例如，暫存器寫入)來觸發。可接著藉由被3D管線2622或媒體管線2624提供的運算後處理來自媒體管線2624輸出。在一些實施例中，GPGPU運算係在與媒體運算類似的方式來組態及執行。

圖形軟體架構

【0314】圖27依據一些實施例闡述用於資料處理系統2700的示範性圖形軟體架構。在一些實施例中，軟體架構包括3D圖形應用2710、作業系統2720以及至少一處理器2730。在一些實施例中，處理器2730包括圖形處理器2732以及一或多個通用處理器核心2734。圖形應用2710和作業系統2720各者在資料處理系統之系統記憶體2750中執行。

【0315】在一些實施例中，3D圖形應用2710包含一或多個著色器程式，其包括著色器指令2712。著色器語言指令可在高階著色器語言中，像是高階著色器語言(HLSL；High Level Shader Language)或OpenGL著色器語言(GLSL；OpenGL Shader Language)。應用亦包括在合適於

藉由通用處理器核心2734執行的機器語言中可執行的指令2714。應用亦包括由頂點資料界定的圖形物件2716。

【0316】在一些實施例中，作業系統2720為來自微軟公司的微軟視窗(Microsoft® Windows®)作業系統、專有的UNIX類作業系統或使用林納斯(Linux)內核之變體的開放來源UNIX類作業系統。作業系統2720能支援API 2722，像是Direct3D API或OpenGL API。當Direct3D API在使用時，作業系統2720使用前端著色器編譯器2724用以將在HLSL中的任何著色器指令2712編譯成低階著色器語言。編譯可為即時化(JIT；just-in-time)編譯或是應用能進行著色器預編譯。在一些實施例中，在3D圖形應用2710之編譯期間，高階著色器被編譯成低階著色器。

【0317】在一些實施例中，使用者模式圖形驅動器2726包含後端著色器編譯器2727用以將著色器指令2712轉換成硬體特定表示。當OpenGL API在使用時，在GLSL高階語言中的著色器指令2712被傳遞到用於編譯的使用者模式圖形驅動器2726。在一些實施例中，使用者模式圖形驅動器2726使用作業系統內核模式功能2728用以與內核模式圖形驅動器2729通訊。在一些實施例中，內核模式圖形驅動器2729與圖形處理器2732通訊以調度命令及指令。

IP核心實施

【0318】至少一實施例之一或多個態樣可藉由儲存在機器可讀媒體上之代表性的碼來實施，其代表及/或界定

在像是處理器的積體電路內的邏輯。舉例來說，機器可讀媒體可包括代表在處理器內各種邏輯的指令。當由機器讀取時，指令可引起機器製造用以進行於此說明的技術的邏輯。這類代表(已知為「IP核心」)為用於積體電路之邏輯的可再用單元，其可被儲存在有形、機器可讀媒體上作為描述積體電路之結構的硬體模型。可供應硬體模型給各種客戶或生產設施(manufacturing facility)，其將硬體模型載入到生產積體電路的製造機器上。可製造積體電路使得電路進行與此說明的實施例之任一者關聯的運算。

【0319】圖28為闡述依據實施例可被使用來生產用以進行運算的積體電路之IP核心發展系統2800的方塊圖。可使用IP核心發展系統2800以產生模組化、可再使用設計，其能被納入到較大設計中或使用以建構完整積體電路(例如，SOC積體電路)。設計設施2830能以高階程式語言(例如，C/C++)產生IP核心設計之軟體模擬2810。能使用軟體模擬2810以使用模擬模型2812來設計、測試及驗證IP核心之行為。模擬模型2812可包括功能、行為及/或時間模擬。接著能從模擬模型2812創建或合成暫存器轉移位準(RTL；register transfer level)設計2815。RTL設計2815為積體電路之行為的抽取，其塑造硬體暫存器之間數位信號的流程，包括與使用塑造的數位信號進行的關聯邏輯。除了RTL設計2815以外，亦可創建、設計或合成在邏輯位準(logic level)或電晶體位準(transistor level)上的低階設計。因此，初始設計及模擬的特定細節可能變化。

【0320】RTL設計2815或等效更可藉由設計設施合成為硬體模型2820，其可在硬體描述語言(HDL；hardware description language)上或實體設計資料的某種其它表示。可更模擬或測試HDL以驗證IP核心設計。能使用非揮發性記憶體2840(例如，硬碟、快閃記憶體或任何非揮發性儲存媒體)儲存IP核心設計以用於遞送至第三方製造設施2865。或者，可在有線連接2850或無線連接2860之上傳送IP核心設計(例如經由網際網路)。製造設施2865接著可製造至少部分基於IP核心設計的積體電路。製造的積體電路可以組態以依據於此說明的至少一實施例進行運算。

示範性晶片上系統積體電路

【0321】圖29~31依據於此所述的各種實施例闡可使用一或多個IP核心來製造的示範性積體電路和關聯的圖形處理器。除了所闡述者以外，可包括其它邏輯和電路，其包括額外的圖形處理器/核心、周邊介面控制器或通用處理器核心。

【0322】圖29為依據實施例闡述可使用一或多個IP核心來製造的晶片上系統積體電路2900的方塊圖。示範性積體電路2900包括一或多個應用處理器2905(例如，CPU)、至少一圖形處理器2910，並且可額外包括影像處理器2915及/或視訊處理器2920，其之任一者可為來自相同或多個不同設計設施的模組化IP核心。積體電路2900包括周邊或匯流排邏輯，其包括USB控制器2925、UART控制器

2930、SPI/SDIO控制器2935以及I^2S/I^2C控制器2940。此外，積體電路能包括顯示裝置2945，其耦接至一或多個高解析度多媒體介面(HDMI；high-definition multimedia interface)控制器2950以及行動工業處理器介面(MIPI；mobile industry processor interface)顯示介面2955。可由快閃記憶體子系統2960提供儲存，其包括快閃記憶體和快閃記憶體控制器。可經由記憶體控制器2965提供記憶體介面以用於對SDRAM或SRAM記憶體裝置存取。一些積體電路額外地包括嵌入式安全引擎2970。

【0323】圖30為依據實施例闡述可使用一或多個IP核心來製造的晶片上系統積體電路之示範性圖形處理器3010的方塊圖。圖形處理器3010能為圖29之圖形處理器2910的變體。圖形處理器3010包括頂點處理器3005和一或多個片段處理器3015A~3015N(例如，3015A、3015B、3015C、3015D到3015N-1以及3015N)。圖形處理器3010能經由分開的邏輯執行不同的著色器程式，使得最佳化頂點處理器3005以執行用於頂點著色器程式的運算，同時一或多個片段處理器3015A~3015N執行用於片段或像素著色器程式的片段(例如，像素)著色運算。頂點處理器3005進行3D圖形管線的頂點處理階段且產生基原及頂點資料。片段處理器3015A~3015N使用由頂點處理器3005產生的基原和頂點資料以生成在顯示裝置上顯示的訊框緩衝(framebuffer)。在一實施例中，片段處理器3015A~3015N被最佳化以執行如用於在OpenGL API中提供的片段著色器程式，其可被使

用以進行如用於在直接3D API提供的像素著色器程式之類似的運算。

【0324】圖形處理器3010額外地包括一或多個記憶體管理單元(MMU；memory management unit)3020A~3020B、快取3025A~3025B以及電路互連3030A~3030B。一或多個MMU 3020A~3020B提供用於對於圖形處理器3010的虛擬對實體位址映射，包括對於頂點處理器3005及/或片段處理器3015A~3015N，除了儲存在一或多個快取3025A~3025B中的頂點或影像/紋理資料以外，其可參考儲存在記憶體中的頂點或影像/紋理資料。在一實施例中，一或多個MMU 3020A~3020B可與在系統內的其它MMU同步，包括與圖29之一或多個應用處理器2905、影像處理器2915及/或視訊處理器2920關聯的一或多個MMU，使得各個處理器2905~2920能參共用或統合虛擬記憶體系統。依據實施例，一或多個電路互連3030A~3030B致能圖形處理器3010以經由SoC之內部匯流排或經由直接連接其一者與在SoC內的其它IP核心介接。

【0325】圖31為依據實施例闡述可使用一或多個IP核心來製造的晶片上系統積體電路之額外示範性圖形處理器3110的方塊圖。圖形處理器3110能為圖29之圖形處理器2910的變體。圖形處理器3110包括圖30之積體電路300的一或多個MMU 3020A~3020B、快取3025A~3025B以及電路互連3030A~3030B。

【0326】圖形處理器3110包括一或多個著色器核心

3115A~3115N(例如，3115A、3115B、3115C、3115D、3115E、3115F到3015N-1及3015N)，其提供用於統合的著色器核心架構，在其中單核或類型或核心能執行所有類型的可程式化著色器碼，包括著色器程式碼，用以實施頂點著色器、片段著色器及/或計算著色器。切確出現的著色器核心之數目能在實施例及實施之間變化。此外，圖形處理器3110包括核間任務管理器3105，其作充當執行緒調度器，用以調度執行執行緒給一或多個著色器核心3115A~3115N。圖形處理器3110額外包括鋪磚單元3118，用以加速用於基於磚之算繪的鋪磚運算，在其中用於場景的算繪運算在影像空間中被細分。能使用基於磚的算繪以利用在場景內局部的空間一致性或用以最佳化內部快取的使用。

【0327】參考至「一實施例」、「實施例」、「範例實施例」、「各種實施例」等指示所述的實施例可包括特別特徵、結構或特性，但非每一個實施例必然包括該特別特徵、結構或特性。進一步，一些實施例可具有一些、所有或沒有針對其它實施例所說明的特徵。

【0328】在前述說明書中，已參考其特定範例實施例來說明實施例。然而，將顯而易見的是，在不悖離如在所附的申請專利範圍中提出的實施例之較寬廣的精神及範圍下可對其作成各種修飾和改變。據此，本說明書及圖式係以闡述性的含義而非限制性的含義來看待。

【0329】在下列說明及申請專利範圍中，可使用術語「耦接」連同其衍生。「耦接」係使用以指示兩個或多個

元件彼此協作或互動，但是他們可或不可具有在他們之間的中介實體或電組件。

【0330】如在申請專利範圍中所使用的，除非另以明定，用以描述共同元件的序數形容詞「第一」、「第二」、「第三」等的使用僅指示相似元件之不同實例正被參照，而不打算暗示所述元件必需以給定的序列，在時間上、空間上、排序上或以任何其它方式其一者。

【0331】下列款項及/或範例屬於進一步實施例或範例。可在一或多個實施例中任何處使用在該些範例中的特異性。不同實施例或範例的各種特徵可與包括的一些特徵和排除以配合各種不同的應用的其它者多方面地結合。範例可包括標的，像是方法、用於進行該方法之動作的機構、包括可執行指令的至少一機器可讀媒體，當由機器施行時，引起機器進行用於依據於此所述的實施例及範例促進混合通訊的方法或設備的動作。

【0332】一些實施例屬於範例1，其包括一種設備，用以在自主機器機促進用於機器學習的屏障和同步，該設備包含：偵測/觀察邏輯，用以偵測關於與一或多個處理裝置關聯的機器學習的執行緒群；以及屏障邏輯，用以促進跨多個晶粒的該執行緒群之屏障同步，使得在執行緒群中各個執行緒係跨與多個晶粒關聯的成組計算元件而被排程，其中各個晶粒代表該一或多個處理裝置中的處理裝置，該處理裝置包括圖形處理器。

【0333】範例2包括範例1之標的，更包含同步邏輯，

用以基於對應於該執行緒群的執行緒群識別(ID)使用屏障命令來促進該執行緒群的同步，其中執行該屏障命令用以當由插入到該屏障命令作為部分的引數的該執行緒群ID所識別時將該執行緒群同步。

【0334】範例3包括範例1~2的標的，其中使用一或多個組構交叉跨多個圖形處理器、跨多個串流多處理器來同步該執行緒群。

【0335】範例4包括範例1~3的標的，其中該同步邏輯更用以透過在通訊網路之上的資料庫促進共用在複數個自主機器之間的中間神經網路，其中該中間神經網路資料由該複數個自主機器中之一或多個被儲存在資料庫處且被壓縮以用於較快的傳送，其中該資料庫包括雲端資料庫且該通訊網路包括雲端網路。

【0336】範例5包括範例1~4的標的，更包含搶先和排程邏輯，用以藉由將該執行緒群之一或多者映射到全域記憶體而跨多個圖形處理器、跨多個SM來排程該執行緒群之一或多者，其中該執行緒群之一或多者未被指定屏障或共用局部記憶體(SLM)。

【0337】範例6包括範例1~5的標的，其中該搶先和排程邏輯更用以藉由切換至執行以防止該一或多個執行緒群進入停頓狀態來搶先該一或多個執行緒群，其中此搶先步驟係基於由執行緒排程器維護的執行緒群列表和狀態，其中該執行緒群列表用以指示允許在周期時間期間被搶先的執行緒群的數目。

【0338】範例7包括範例1~6的標的，其中該圖形處理器係與在共同半導體封裝上的應用處理器共定位。

【0339】一些實施例屬於範例8，其包括一種方法，用以在自主機器機促進用於機器學習的屏障和同步，該方法包含：偵測關於與一或多個處理裝置關聯的機器學習的執行緒群；以及促進跨多個晶粒的該執行緒群之屏障同步，使得在執行緒群中的各個執行緒跨與該多個晶粒關聯的成組計算元件而被排程，其中各個晶粒代表該一或多個處理裝置中的處理裝置，該處理裝置包括圖形處理器。

【0340】範例9包括範例8之標的，更包含基於對應於該執行緒群的執行緒群識別(ID)使用屏障命令來促進該執行緒群的同步，其中執行該屏障命令用以當由插入到該屏障命令作為部分的引數的該執行緒群ID所識別時將該執行緒群同步。

【0341】範例10包括範例8~9的標的，其中使用一或多個組構交叉跨多個圖形處理器、跨多個串流多處理器來(SM)同步該執行緒群。

【0342】範例11包括範例8~10的標的，更包含透過在通訊網路之上的資料庫促進共用在複數個自主機器之間的中間神經網路，其中該中間神經網路資料由該複數個自主機器中之一或多個被儲存在資料庫處且被壓縮以用於較快的傳送，其中該資料庫包括雲端資料庫且該通訊網路包括雲端網路。

【0343】範例12包括範例8~11的標的，更包含藉由將

該執行緒群之一或多者映射到全域記憶體而跨多個圖形處理器、跨多個SM來排程該執行緒群之一或多者，其中該執行緒群之一或多者未被指定屏障或共用局部記憶體(SLM)。

【0344】範例13包括範例8~12的標的，更包含藉由切換至執行以防止該一或多個執行緒群進入停頓狀態來搶先該一或多個執行緒群，其中此搶先步驟係基於由執行緒排程器維護的執行緒群列表和狀態，其中該執行緒群列表用以指示允許在周期時間期間被搶先的執行緒群的數目。

【0345】範例14包括範例8~13的標的，其中該圖形處理器係與在共同半導體封裝上的應用處理器共定位。

【0346】一些實施例屬於範例15，其包括一種圖形處理系統，包含計算裝置，具有耦接至處理器的記憶體，該處理器用以：偵測關於與一或多個處理置關聯的機器學習的執行緒群；以及跨多個晶粒促進該執行緒群的屏障同步，使得在執行緒群中各個執行緒跨與該多個晶粒關聯的成組計算元件而被排程，其中各個晶粒代表該一或多個處理裝置中的處理裝置，該處理裝置包括圖形處理器。

【0347】範例16包括範例15之標的，其中該處理器更用以基於對應於該執行緒群的執行緒群識別(ID)使用屏障命令來促進該執行緒群的同步，其中執行該屏障命令用以當由插入到該屏障命令作為部分的引數的該執行緒群ID所識別時將該執行緒群同步。

【0348】範例17包括範例15~16的標的，其中使用一

或多個組構交叉跨多個圖形處理器、跨多個串流多處理器(SM)來同步該執行緒群。

【0349】範例18包括範例15~17的標的，其中該運算更包含透過在通訊網路之上的資料庫促進共用在複數個自主機器之間的中間神經網路，其中該中間神經網路資料由該複數個自主機器中之一或多個被儲存在資料庫處且被壓縮以用於較快的傳送，其中該資料庫包括雲端資料庫且該通訊網路包括雲端網路。

【0350】範例19包括範例15~18的標的，其中該運算更包含藉由將該執行緒群之一或多者映射到全域記憶體而跨多個圖形處理器、跨多個SM來排程該執行緒群之一或多者，其中該執行緒群之一或多者未被指定屏障或共用局部記憶體(SLM)。

【0351】範例20包括範例15~19的標的，其中該運算更包含藉由切換至執行以防止該一或多個執行緒群進入停頓狀態來搶先該一或多個執行緒群，其中此搶先步驟係基於由執行緒排程器維護的執行緒群列表和狀態。

【0352】範例21包括範例15~20的標的，其中該執行緒群列表用以指示允許在周期時間期間被搶先的執行緒群之數目，其中該圖形處理器係與在共用半導體封裝上的應用處理器共定位。

【0353】範例22包括至少一種非暫態或有形的機器可讀媒體，包含複數個指令，當在計算裝置上執行時，用以實施或進行如在申請專利範圍或範例8~14之任一者中所請

求的方法。

【0354】範例23包括至少一種機器可讀媒體，包含複數個指令，當在計算裝置上執行時，用以實施或進行如在申請專利範圍或範例8~14之任一者中所請求的方法。

【0355】範例24包括一種系統，包含用以實施或進行如在申請專利範圍或範例8~14之任一者中所請求的方法的機制。

【0356】範例25包括一種設備，包含用於實施或進行如在申請專利範圍或範例8~14之任一者中所請求的方法的機構。

【0357】範例26包括一種計算裝置，安排以實施或進行如在申請專利範圍或範例8~14之任一者中所請求的方法。

【0358】範例27包括一種通訊裝置，安排以實施或進行如在申請專利範圍或範例8~14之任一者中所請求的方法。

【0359】範例28包括至少一種機器可讀媒體，包含複數個指令，當在計算裝置上執行時，用以實施或進行如在先前任何申請專利範圍中請求的方法或實現如在先前任何申請專利範圍中請求的設備。

【0360】範例29包括至少一種非暫態或有形的機器可讀媒體，包含複數個指令，當在計算裝置上執行時，用以實施或進行如在先前任何申請專利範圍中請求的方法或實現如在先前任何申請專利範圍中請求的設備。

【0361】範例30包括一種系統，包含用以實施或進行如在先前任何申請專利範圍中請求的方法或實現如在先前任何申請專利範圍中請求的設備的機制。

【0362】範例31包括一種設備，包含用於用以進行如在先前任何申請專利範圍中請求的方法的機構。

【0363】範例32包括一種計算裝置，安排以實施或進行如在先前任何申請專利範圍中請求的方法或實現如在先前任何申請專利範圍中請求的設備。

【0364】範例33包括一種通訊裝置，安排以實施或進行如在先前任何申請專利範圍中請求的方法或實現如在先前任何申請專利範圍中請求的設備。

【0365】圖式和先前說明給定實施例的範例。本領域具有通常知識的該些者將了解的是，描述的元件之一或多者將良好地被結合成單功能元件。或者，某些元件可被分成多個功能元件。來自一實施例的元件可被加至另一實施例。例如，於此說明的過程之次序可被改變但不限於於此說明的方式。再者，任何流程圖的動作不需要以所繪示的次序來實施；且不必然需要進行所有的動作。也同樣的，非相依於其它動作的該些動作可與其餘的動作平行進行。實施例的範圍絕非由這些特定範例限制。無論在說明書中有無明白地給定，眾多的變化，像是在結構、維度及材料之使用上的差異是可能的。實施例的範圍係至少如由下列申請專利範圍所給定般的寬。

【符號說明】

【0366】

100:計算系統

101:處理子系統

102:處理器

104:系統記憶體

105:記憶體中樞

106:通訊連結

107:輸入/輸出中樞

108:輸入裝置

110A/110B:顯示裝置

111:輸入/輸出子系統

112:平行處理器

113:通訊連結

114:系統儲存單元

116:I/O切換器

118:網路配接器

119:無線網路配接器

120:外加裝置

202(A-N):平行處理單元

204:輸入/輸出單元

206:主機介面

208:前端

210:排程器

212:處理叢集陣列

214A-N:叢集

216:記憶體交叉

218:記憶體介面

220A-N:分割單元

222:平行處理器記憶體

224A-N:記憶體單元

221:2級快取

225:框緩衝器介面

226:光柵運算單元

232:管線管理器

234:圖形多處理器

236:紋理單元

240:資料交叉

242:預光柵運算單元

245:記憶體管理單元

252:指令快取

254:指令單元

256:位址映射單元

258:暫存器檔案

262:通用圖形處理單元核心

266:載入/儲存單元

268:記憶體和快取互連

270:共用記憶體

272:快取記憶體

325:圖形多處理器

327:互連組構

330:共同指令快取

332A-B:指令單元

334A-B:暫存器檔案

336A-B:通用圖形處理單元核心

337A-B:通用圖形處理單元核心

338A-B:通用圖形處理單元核心

340A-B:載入/儲存單元

342:快取記憶體

344A-B:紋理單元

346:共用記憶體

350:圖形多處理器

352:互連組構

354:指令快取

356A-D:執行資源

358A-B:快取記憶體

360A-D:紋理單元

362:共用記憶體

401:處理器記憶體

402:處理器記憶體

405:多核心處理器

406:多核心處理器

407:多核心處理器

410:圖形處理單元

411:圖形處理單元

412:圖形處理單元

413:圖形處理單元

420:圖形處理單元記憶體

421:圖形處理單元記憶體

422:圖形處理單元記憶體

423:圖形處理單元記憶體

425:代理器電路

426:快取

430:記憶體互連

431:圖形處理引擎

432:圖形處理引擎

433:圖形記憶體

434:圖形記憶體

435:介面

436:加速器整合電路

437:介面

438:快取

439(A-E):記憶體管理單元

440:連結

441:系統記憶體

442:高速連結

443:高速連結

444:高速連結

445:高速連結

446:圖形加速器模組

447:中斷管理電路

448:上下文管理

450:上下文管理電路

451:記憶體互連

452:記憶體互連

453:記憶體互連

460A-D:核心

461A-D:轉譯旁視緩衝器

462A-D:快取

464:一致性匯流排

456:核心快取

480:應用

481:圖形處理單元調用

482:位址空間

483:進程元件

484:工作描述符

485:作業系統虛擬位址空間

486:區段/頁面表

490:加速器整合片

491:提取單元

492:中斷事件

493:有效位址

494A-E:偏移/一致性管理電路

495:作業系統

496:系統超管理器

498:超管理器真實位址空間

499:進程元件列表

500:圖形處理管線

502:資料組譯器

504:頂點處理單元

506:基元組譯器

508:曲面細分控制處理單元

510:曲面細分單元

512:曲面細分評估處理單元

514:基元組譯器

516:幾何處理單元

518:基元組譯器

520:視埠定比、剔除及剪輯單元

522:柵格器

524:片段/像素處理單元

526:光柵運算單元

528:記憶體介面

600:計算裝置

604:輸入/輸出來源

606:作業系統

608:記憶體

610:屏障和同步機制

612:應用處理器

614:圖形處理器

616:驅動器

701:偵測/觀察邏輯

703:屏障邏輯

705:同步邏輯

707:通訊/相容性邏輯

709:搶先和排程邏輯

725:通訊媒體

730:資料庫

801:圖形處理單元

803:圖形處理單元

805:圖形處理單元

807:圖形處理單元

820:框架

821:執行緒群

823:執行緒群

825:執行緒群

827:執行緒群

831:屏障命令

833:引數

841:系統

843:系統

845:系統

901:執行緒排程器

903:資訊或暗示

905:串流多處理器

907:串流多處理器

911A-B:執行單元

913A-B:執行單元

915:屏障

917:屏障

921:共用局部記憶體

923:共用局部記憶體

925:快取

927:快取

950:框架

951:執行緒群

953:執行緒群

955:執行緒群

957:計算資源

959:執行緒排程器

1000:機器學習軟體堆層

1002:機器學習應用

1004:機器學習框架

1006:計算框架

1008:通用圖形處理單元驅動器

1010:通用圖形處理單元硬體

1100:高度平行通用圖形處理單元

1102:主機介面

1104:全域排程器

1106A-H:計算叢集

1108:輸入/輸出中樞

1110:通用圖形處理單元連結

1112A-B:記憶體控制器

1114A-B:記憶體

1202:處理器

1204:主機介面切換器

1206A-D:通用圖形處理單元

1216:高速點對點GPU對CPU連結

1302:輸入

1304:卷積層

1306:卷積層

1308:完全連接層

1312:卷積層

1314:卷積層

1316:卷積階段

1318:偵測器階段

1320:池化階段

1322:層

1400:遞歸神經網路

1402:輸入層

1404:隱藏層

1405:回饋機制

1406:輸出層

1502:訓練資料集

1504:訓練框架

1506:未訓練的神經網路

1507/1508:訓練的神經網路

1512:資料

1514:結果

1602:模型平行性

1604:資料平行性

1606:結合的模型和資料平行性

1702:媒體處理器

1704:視覺處理器

1705:晶片上記憶體

1706:通用圖形處理單元

1708:多核心處理器

1800:處理系統

1802:處理器

1804:快取記憶體

1806:暫存器檔案

1807:處理核心

1808:圖形處理器

1809:指令集

1810:處理器匯流排

1812:外部圖形處理器

1816:記憶體控制器中樞

1820:記憶體裝置

1821:指令

1822:資料

1830:輸入/輸出控制器中樞

1834:網路控制器

1840:舊有的輸入/輸出控制器

1842:通用串列匯流排控制器

1844:鍵盤和滑鼠

1846:音訊控制器

1900:處理器

1904A-N:內部快取單元

1906:共用快取單元

1908:圖形處理器

1910:系統代理核心

1911:顯示控制器

1912:基於環狀的互連單元

1913:輸入/輸出連結

1914:積體記憶體控制器

1916:匯流排控制器單元

1918:記憶體模組

2000:圖形處理器

2002:顯示控制器

2004:區塊影像轉移引擎

2006:視訊編解碼器引擎

2010:圖形處理引擎

2012:三維管線

2014:記憶體介面

2015:三維/媒體子系統

2016:媒體管線

2020:顯示裝置

2103:命令串流器

2110:圖形處理引擎

2112:三維管線

2114:記憶體介面

2116:媒體管線

2118:統合回傳緩衝器

2120:共用功能邏輯

2121:取樣器

2122:數學邏輯

2123:執行緒間通訊邏輯

2125:快取

2200:圖形處理器

2202:環狀互連

2203:命令串流器

2204:管線前端

2230:視訊品質引擎

2233:多格式編碼/解碼引擎

2234:視訊前端

2236:幾何管線

2237:媒體引擎

2252A-N:執行單元

2254A-N:媒體/紋理取樣器

2260A-N:第二子核心

2262A-N:執行單元

2264A-N:取樣器

2270A-N:共用資源

2280A-N:圖形核心

1450A:子核心

1460A:子核心

550N:子核心

560N:子核心

2300:執行緒執行邏輯

2302:像素著色器

2304:執行緒調度器

2306:指令快取

2308A-N:執行單元

2310:取樣器

2312:資料快取

2314:資料埠

2400:圖形處理器指令格式

2412:指令運算碼

2413:索引欄位

2414:指令控制欄位

2416:執行尺寸欄位

2418:目的地

2420:來源運算元

2422:來源運算元

2424:來源運算元

2426:存取/位址模式資訊

2430:64位元壓縮指令格式

2440:運算碼解碼

2442:移動和邏輯群組

2444:流程控制指令群組

2446:雜項指令群組

2448:平行數學指令群組

2450:向量數學群組

2500:圖形處理器

2502:環狀互連

2503:令命串流器

2505:頂點提取器

2507:頂點著色器

2511:可程式外殼著色器

2513:曲面細分器

2517:可程式域著色器

2519:幾何著色器

2520:圖形管線

2523:流出單元

2529:剪取器

2530:圖形處理器媒體管線

2534:視訊前端

2537:媒體引擎

2540:顯示引擎

2541:二維引擎

2543:顯示控制器

2551:1級快取

2552A-B:執行單元

2556:資料埠

2558:快取

2570:算繪輸出管線

2573:深度測器組件

2575:3級快取

2577:像素運算組件

2578:算繪快取

2579:深度快取

1731:執行緒調度器

1754:取樣器

1778:算繪快取

2600:圖形處理器命令格式

2602:目標客戶端

2604:命令運算碼

2605:子運算碼

2606:資料

2608:命令尺寸

2700:資料處理系統

2710:三維圖形應用

2712:著色器指令

2714:可執行指令

2716:圖形物件

2720:作業系統

2722:圖形可程式介面

2724:前端著色器編譯器

2726:使用者模式圖形驅動器

2727:後端著色器編譯器

2728:作業系統內核模式功能

2729:內核模式圖形驅動器

2730:處理器

2732:圖形處理器

2734:通用處理器核心

2750:系統記憶體

2800:智財核心發展系統

2810:軟體模擬

2812:模擬模型

2815:暫存器轉移位準設計

2820:硬體模型

2830:設計設施

2840:非揮發性記憶體

2850:有線連接

2860:無線連接

2865:製造設施

2900:晶片積體電路

2905:應用處理器

2910:圖形處理器

2915:影像處理器

2920:視訊處理器

2925:通用序列匯流排處理器

2930:通用非同步接收器傳輸器控制器

2935:序列周邊介面/安全數位輸入輸出控制器

2940:積體電路間聲音/積體電路間控制器

2945:顯示裝置

2950:高傳真多媒體介面控制器

2955:行動工業處理器介面顯示介面

2960:快閃記憶體子系統

2965:記憶體控制器

2970:嵌入式安全引擎

3005:頂點處理器

3010:圖形處理器

3015A-N:片段處理器

3020A-B:記憶體管理單元

3025A-B:快取

3030A-B:電路互連

3105:核心間任務管理器

3110:圖形處理器

3115A-N:著色器核心

3118:鋪磚單元

3120A-B:記憶體管理單元

3125A-B:快取

3130A-B:互連

【發明申請專利範圍】

【請求項1】一種用於機器學習的設備，包含：

執行緒排程器，用以跨代表多個圖形處理器的多個晶粒來排程執行緒群，該多個圖形處理器包括分開的成組的處理器資源，其中該執行緒排程器係用以排程用於在第一晶粒上第一圖形處理器之第一組處理資源及在第二晶粒上第二圖形處理器之第二組處理資源上執行的第一執行緒群，該第二晶粒不同於該第一晶粒；以及

硬體屏障邏輯，組態以偵測該執行緒群關於機器學習運算，並且在該偵測之後，經由多晶粒屏障跨該多個晶粒促進在該執行緒群內執行緒之屏障同步，其中該多晶粒屏障包括由在該多個晶粒上該多個圖形處理器執行的屏障指令，用以同步在跨該多個晶粒而排程的該執行緒群內該執行緒的執行，其中反應於該第一執行緒群的執行緒已進入由執行該多個晶粒屏障之該屏障指令引起的屏障停頓狀態的判定，該執行緒排程器係用來以被排程且擱置執行同時在可執行狀態的第二執行緒群之執行緒取代在該屏障停頓狀態中的該第一執行緒群之該執行緒，其中該執行緒排程器係用來以該第二執行緒群之第一執行緒取代該第一執行緒群之第一執行緒而不引起該第一執行緒群之該第一執行緒在該屏障停頓狀態中等待。

【請求項2】如請求項1所述的設備，其中該硬體屏障邏輯係用以使用具有對應於該執行緒群的執行緒群識別(ID)的屏障命令來促進該屏障同步，其中各個執行緒群被

指定執行緒群ID。

【請求項3】如請求項2所述的設備，其中使用一或更多組構交叉及該多個圖形處理器之多晶粒屏障硬體來跨該多個圖形處理器同步該執行緒群。

【請求項4】如請求項1所述的設備，其中為了致能該執行緒群之一或更多者跨該多個圖形處理器來被排程，該執行緒排程器係用以將該執行緒群之一或更多者的共用局部記憶體空間映射至對該多個圖形處理器是全域的記憶體空間。

【請求項5】如請求項1所述的設備，其中該執行緒排程器係用以維護跨該多個晶粒排程的成列的執行緒群以及在該列執行緒群內執行緒之狀態，基於該列執行緒群及在該列執行緒群內該執行緒之該狀態來搶先該第一執行緒群之該執行緒，並且維護成列的搶先的執行緒群及用於在該列搶先的執行緒群內搶先的執行緒的狀態。

【請求項6】如請求項5所述的設備，其中基於該列執行緒群搶先該第一執行緒群之該執行緒包括：在多個晶粒上搶先執行緒。

【請求項7】一種用於機器學習的方法，包含：

經由執行緒排程器跨代表多個圖形處理器的多個晶粒來排程執行緒群，該多個圖形處理器包括分開的成組的處理器資源，其中該執行緒排程器係用以排程用於在第一晶粒上第一圖形處理器之第一組處理資源及在第二晶粒上第二圖形處理器之第二組處理資源上執行的第一執行緒群，

該第二晶粒不同於該第一晶粒；

經由硬體屏障邏輯來偵測該執行緒群關於機器學習運算；以及

在該偵測之後，經由多晶粒屏障跨該多個晶粒促進在該執行緒群內執行緒之屏障同步，其中該多晶粒屏障包括由在該多個晶粒上該多個圖形處理器執行的屏障指令，用以同步在跨該多個圖形處理器而排程的該執行緒群內該執行緒的執行，其中反應於該第一執行緒群的執行緒已進入由執行該多個晶粒屏障之該屏障指令引起的屏障停頓狀態的判定，以被排程且擱置執行同時在可執行狀態的第二執行緒群之執行緒取代在該屏障停頓狀態中的該第一執行緒群之該執行緒，其中該取代包括：以該第二執行緒群之第一執行緒取代該第一執行緒群之第一執行緒而不引起該第一執行緒群之該第一執行緒在該屏障停頓狀態中等待。

【請求項8】如請求項7所述的方法，更包含：使用具有對應於該執行緒群的執行緒群識別(ID)的屏障命令來促進同步，其中各個執行緒群被指定執行緒群ID。

【請求項9】如請求項8所述的方法，其中使用一或更多組構交叉及該多個圖形處理器之多晶粒屏障硬體來跨該多個圖形處理器同步該執行緒群。

【請求項10】如請求項7所述的方法，更包含：藉由將該執行緒群之一或更多者的共用局部記憶體空間映射至對該多個圖形處理器是全域的記憶體空間，致能跨該多個圖形處理器該執行緒群之一或更多者的該排程。

【請求項11】如請求項7所述的方法，更包含：

由該執行緒排程器維護跨該多個晶粒排程的成列的執行緒群以及在該列執行緒群內執行緒的狀態；

由該執行緒排程器基於該列執行緒群及在該列執行緒群內該執行緒之該狀態來搶先在該第一執行緒群中的該執行緒；以及

由該執行緒排程器維護成列的搶先的執行緒群以及用於在該列搶先的執行緒群內搶先的執行緒的狀態。

【請求項12】如請求項11所述的方法，更包含：基於跨該多個晶粒排程的該列執行緒群來跨該多個晶粒搶先該執行緒。

【請求項13】一種非暫態機器可讀媒體，包含指令，其當由計算裝置執行時，引起該計算裝置進行運算，包含：

經由執行緒排程器跨代表多個圖形處理器的多個晶粒來排程執行緒群，該多個圖形處理器包括分開的成組的處理器資源，其中該執行緒排程器係用以排程用於在第一晶粒上第一圖形處理器之第一組處理資源及在第二晶粒上第二圖形處理器之第二組處理資源上執行的第一執行緒群，該第二晶粒不同於該第一晶粒；

經由硬體屏障邏輯來偵測該執行緒群關於機器學習運算；以及

在該偵測之後，經由多晶粒屏障跨該多個晶粒促進該執行緒群之屏障同步，其中該多晶粒屏障包括由在該多個

晶粒上該多個圖形處理器執行的屏障指令，用以同步在跨該多個圖形處理器而排程的該執行緒群內該執行緒的執行，其中反應於該第一執行緒群的執行緒已進入由執行該多個晶粒屏障之該屏障指令引起的屏障停頓狀態的判定，以被排程且擱置執行同時在可執行狀態的第二執行緒群之執行緒取代在該屏障停頓狀態中的該第一執行緒群之該執行緒，其中該取代包括：以該第二執行緒群之第一執行緒取代該第一執行緒群之第一執行緒而不引起該第一執行緒群之該第一執行緒在該屏障停頓狀態中等待。

【請求項14】如請求項13所述的非暫態機器可讀媒體，其中該運算更包含：使用具有對應於該執行緒群的執行緒群識別(ID)的屏障命令來促進同步，其中各個執行緒群被指定執行緒群ID。

【請求項15】如請求項14所述的非暫態機器可讀媒體，其中使用一或更多組構交叉及該多個圖形處理器之多晶粒屏障硬體來跨該多個圖形處理器同步該執行緒群。

【請求項16】如請求項13所述的非暫態機器可讀媒體，其中該運算更包含：藉由將該執行緒群之一或更多者的共用局部記憶體空間映射至對該多個圖形處理器是全域的記憶體空間，致能跨該多個圖形處理器該執行緒群之一或更多者的該排程。

【請求項17】如請求項13所述的非暫態機器可讀媒體，更包含：

由該執行緒排程器維護跨該多個晶粒排程的成列的執

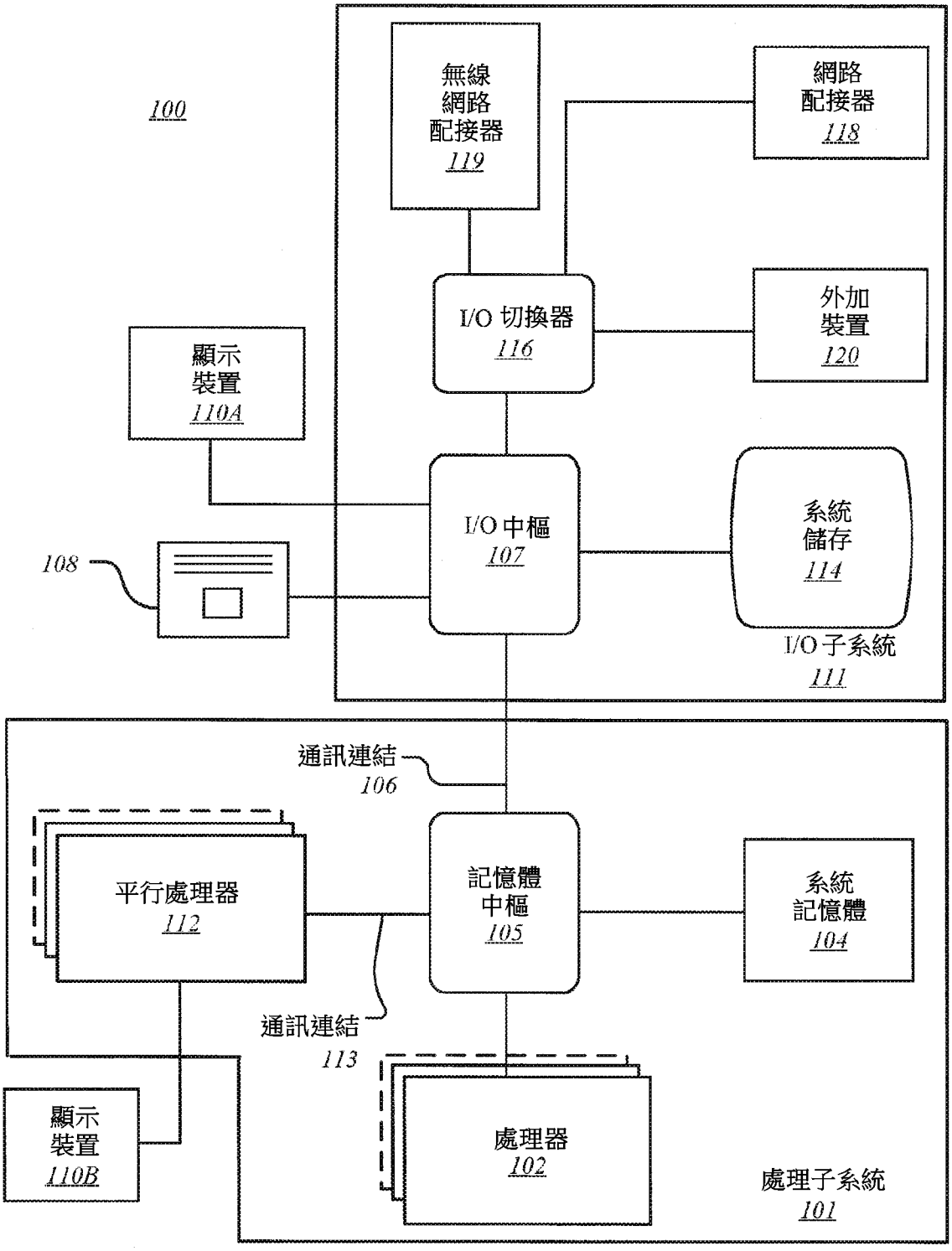
行緒群以及在該列執行緒群內執行緒的狀態；

由該執行緒排程器基於該列執行緒群及在該列執行緒群內該執行緒之該狀態來搶先在該第一執行緒群中的該執行緒；以及

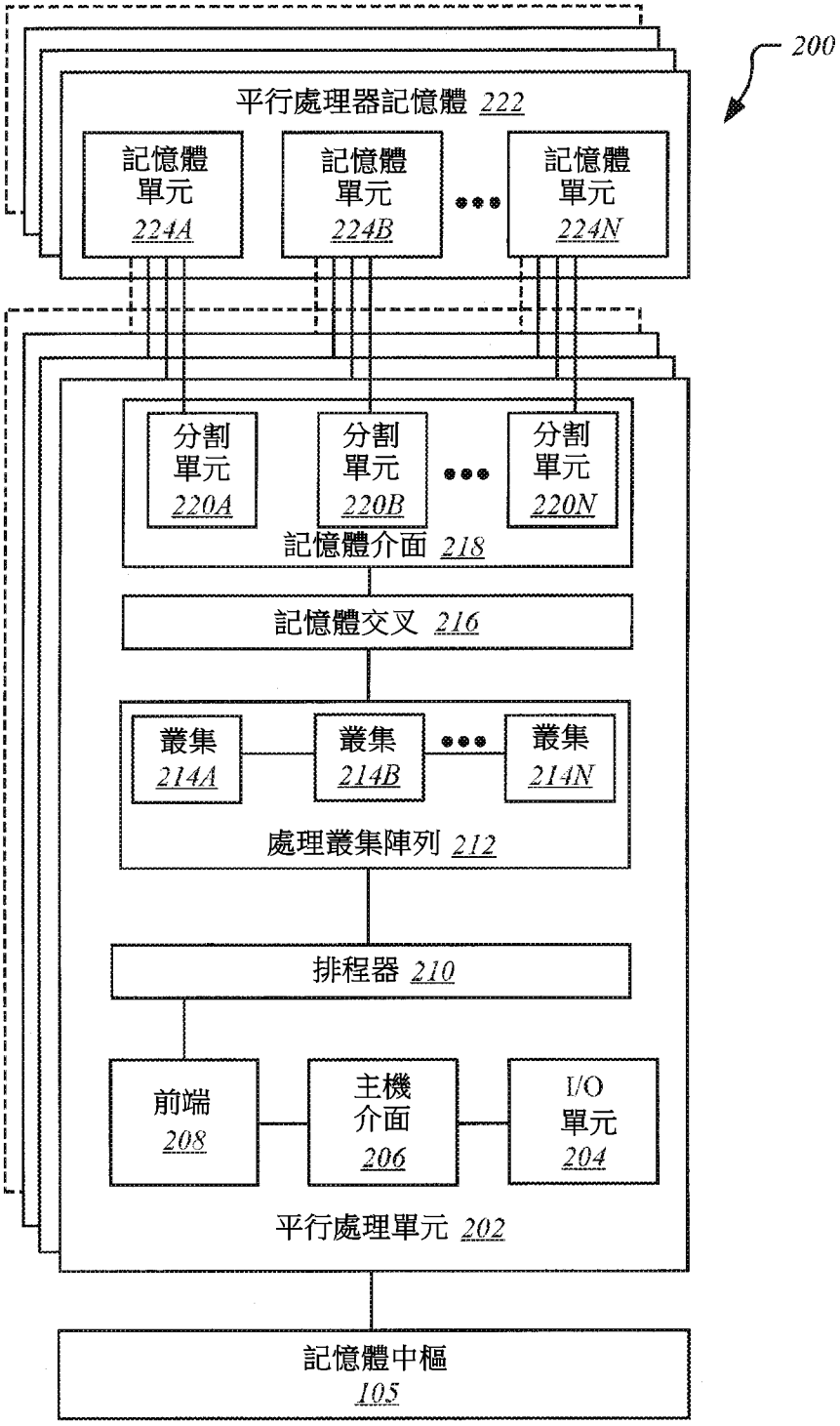
由該執行緒排程器維護成列的搶先的執行緒群以及用於在該列搶先的執行緒群內搶先的執行緒的狀態。

【請求項18】如請求項17所述的非暫態機器可讀媒體，該運算更包含：基於跨該多個晶粒排程的該列執行緒群來跨該多個晶粒搶先該執行緒。

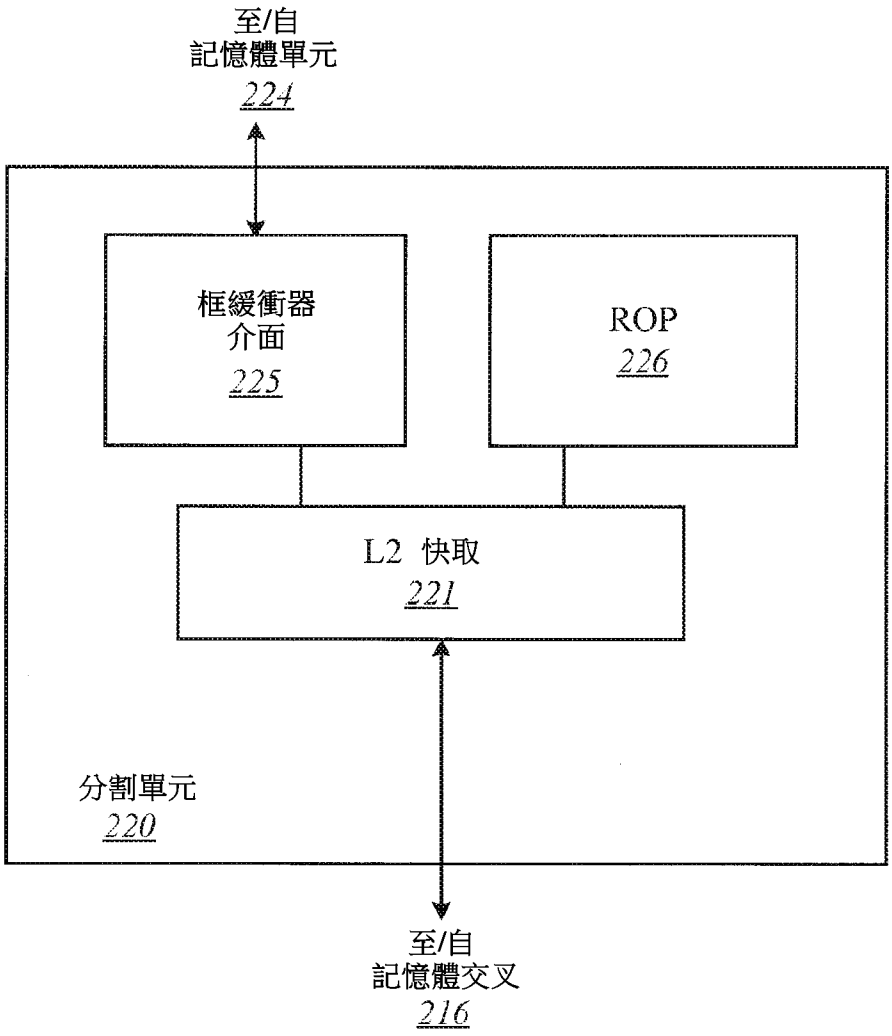
【發明圖式】


100
無線
網路
配接器
119
網路
配接器
118
I/O 切換器
116
外加
裝置
120
顯示
裝置
110A
I/O 中樞
107
系統
儲存
114
108
I/O 子系統
111
通訊連結
106
平行處理器
112
記憶體
中樞
105
系統
記憶體
104
通訊連結
113
顯示
裝置
110B
處理器
102
處理子系統
101

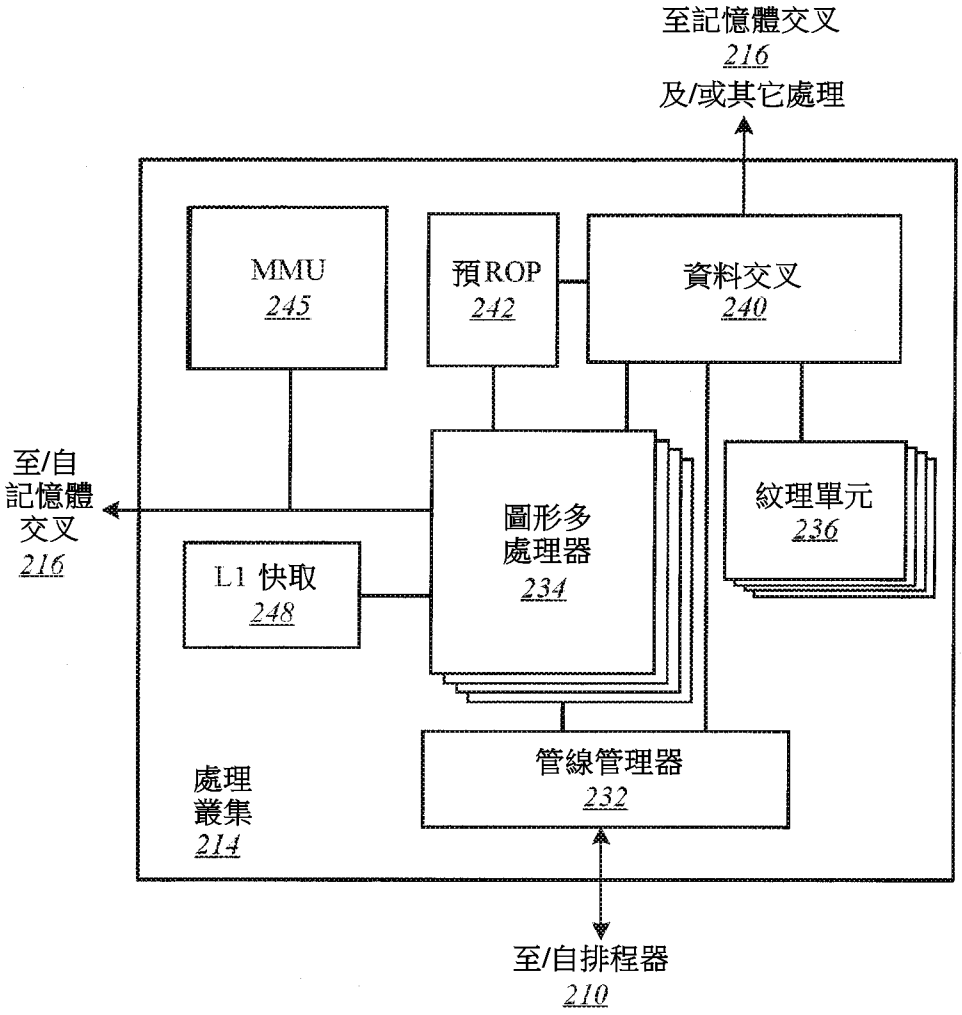
【圖 1】


200
平行處理器記憶體 222
記憶體單元 224A
記憶體單元 224B
記憶體單元 224N
分割單元 220A
分割單元 220B
分割單元 220N
記憶體介面 218
記憶體交叉 216
叢集 214A
叢集 214B
叢集 214N
處理叢集陣列 212
排程器 210
前端 208
主機介面 206
I/O 單元 204
平行處理單元 202
記憶體中樞 105

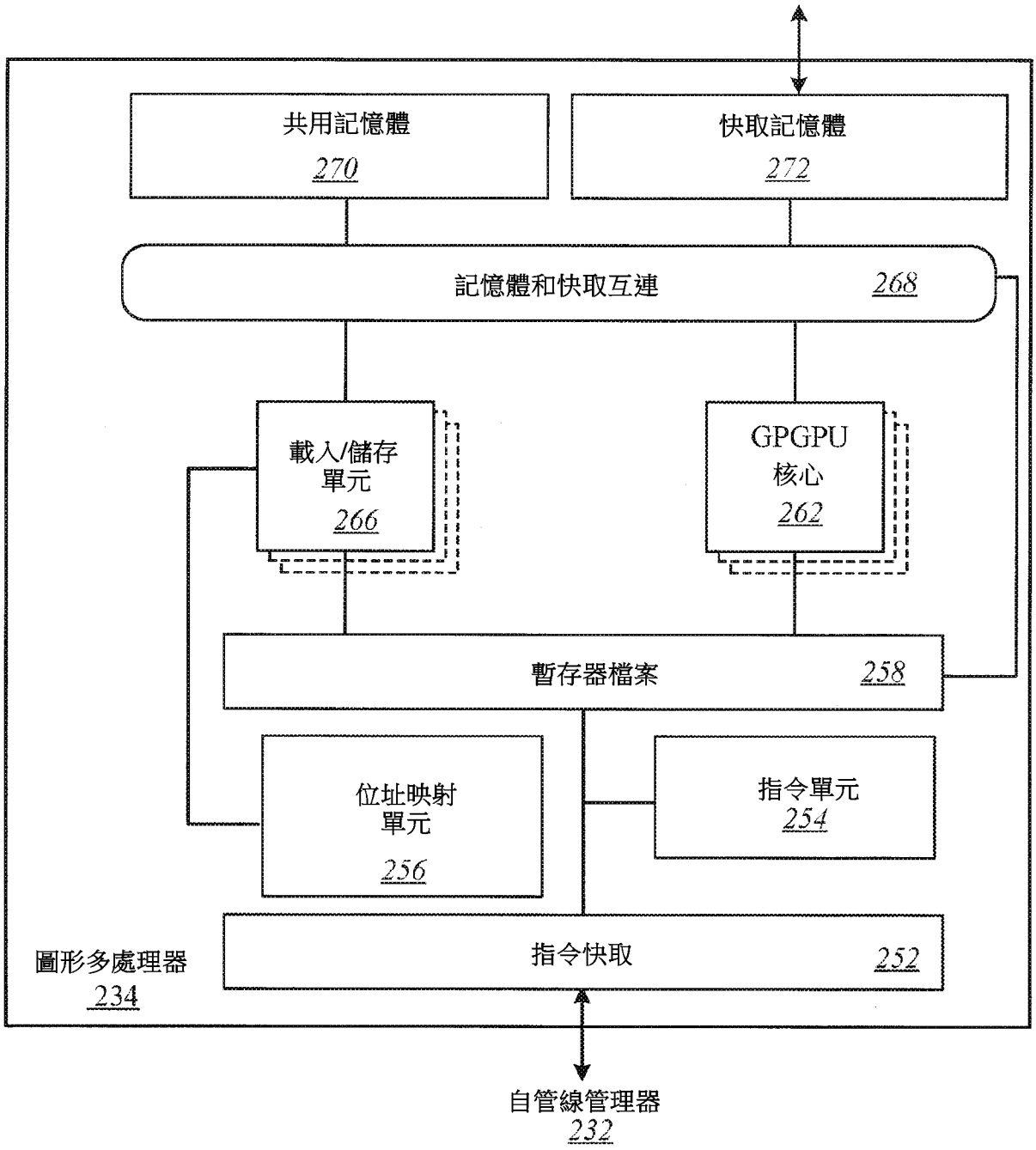
【圖 2A】



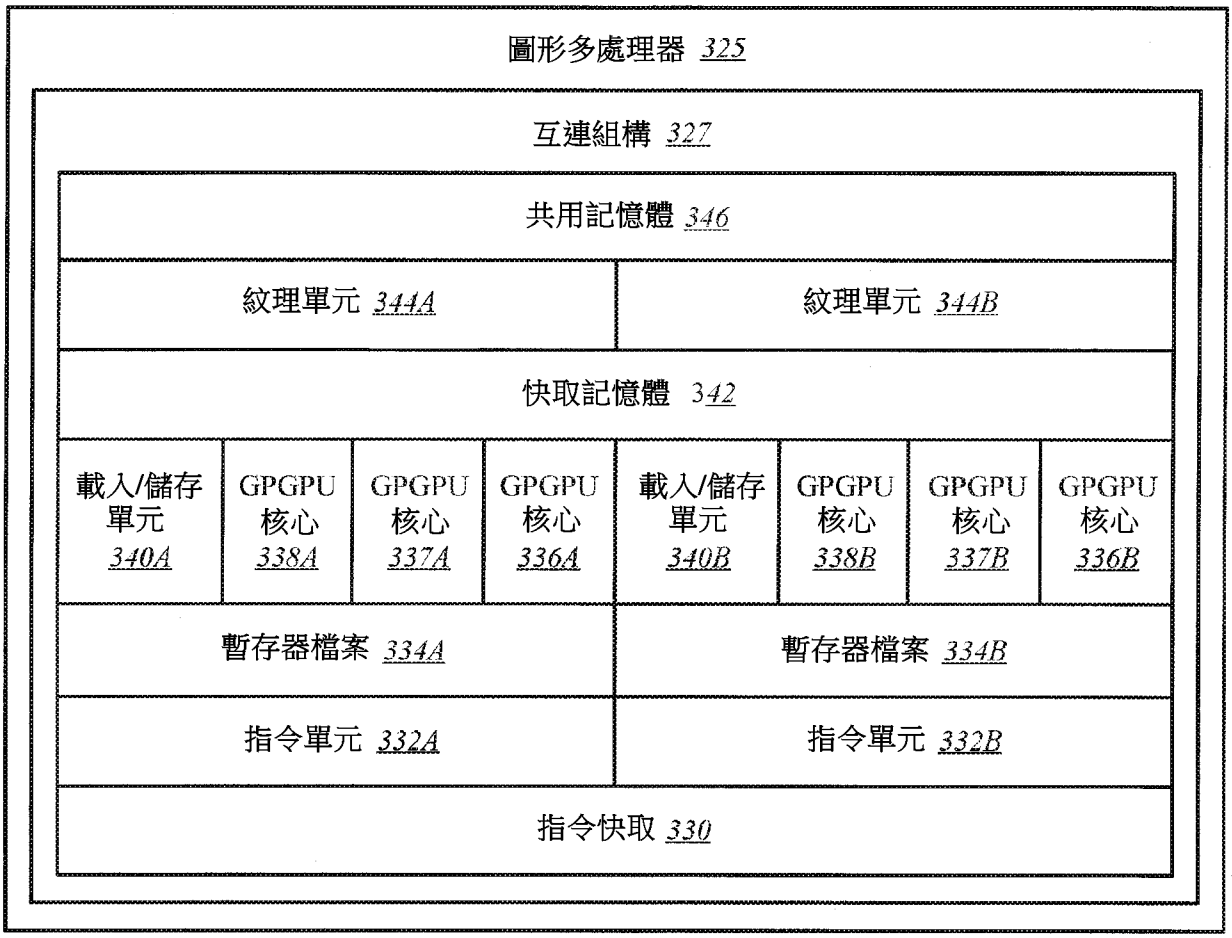
【圖 2B】


至記憶體交叉
216
及/或其它處理
MMU
245
預ROP
242
資料交叉
240
至/自
記憶體
交叉
216
圖形多
處理器
234
紋理單元
236
L1 快取
248
管線管理器
232
處理
叢集
214
至/自排程器
210

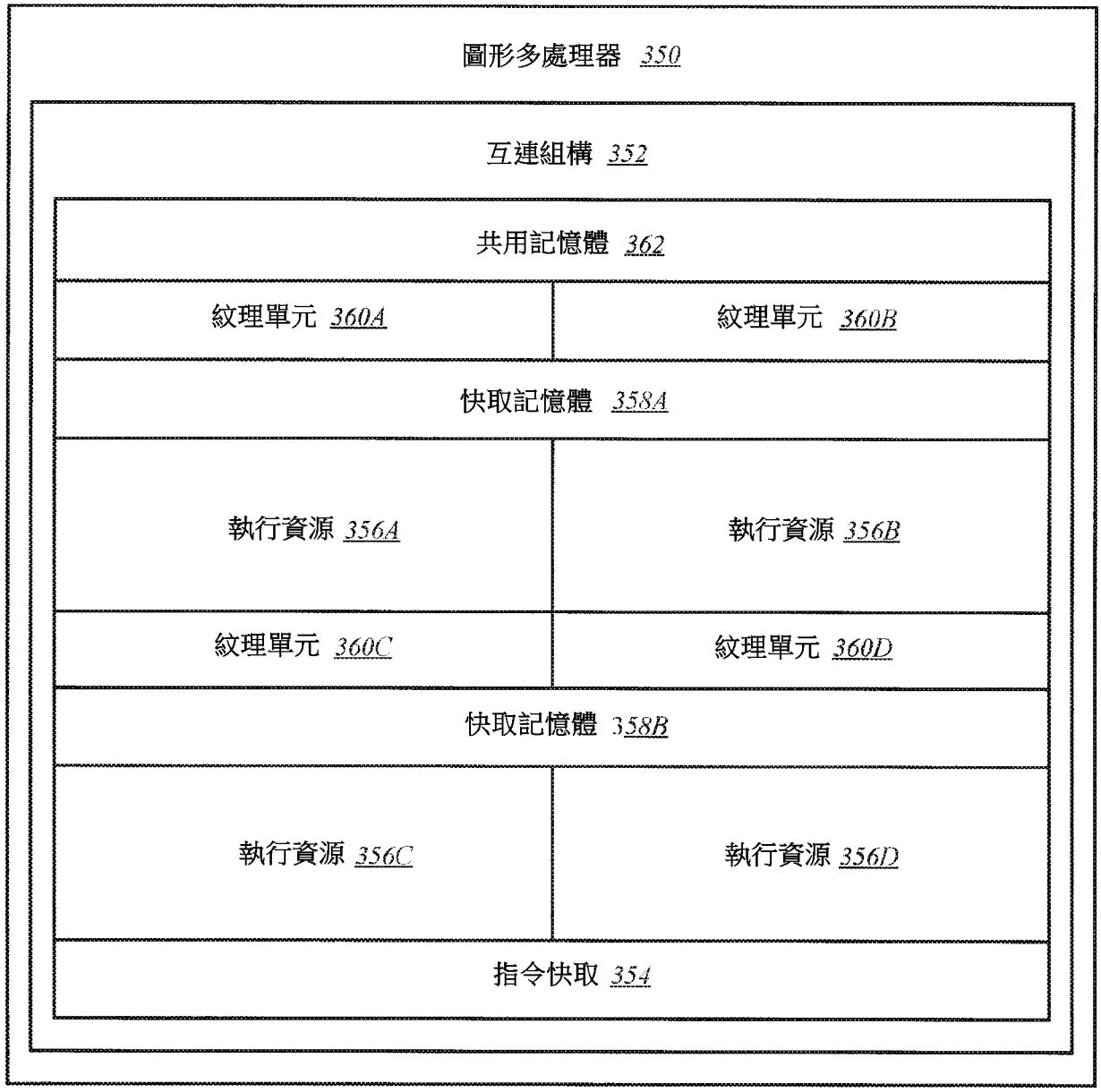
【圖 2C】


共用記憶體
270
快取記憶體
272
記憶體和快取互連
268
載入/儲存
單元
266
GPGPU
核心
262
暫存器檔案
258
位址映射
單元
256
指令單元
254
指令快取
252
圖形多處理器
234
自管線管理器
232

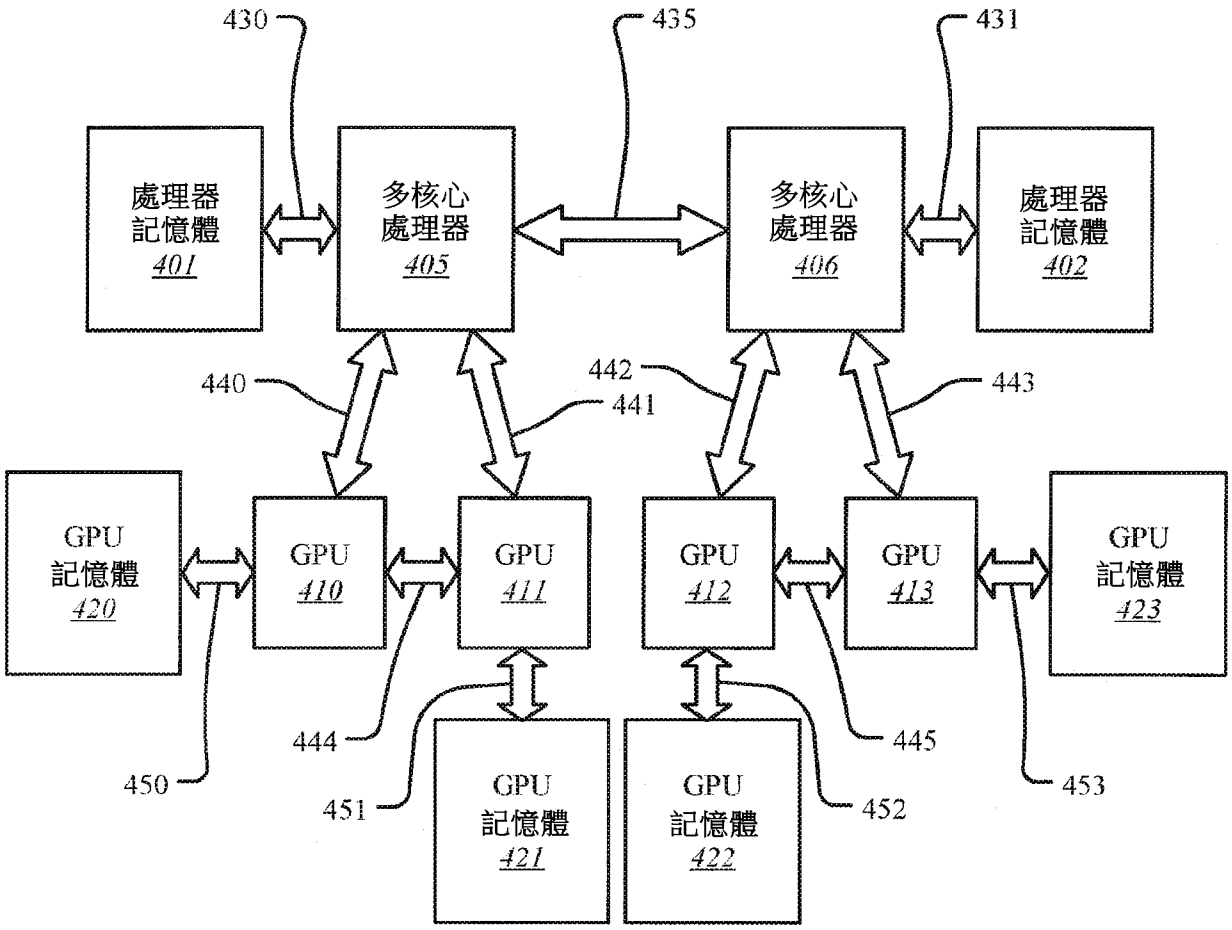
【圖 2D】


圖形多處理器 325
互連組構 327
共用記憶體 346
紋理單元 344A
紋理單元 344B
快取記憶體 342
載入/儲存單元 340A
GPGPU 核心 338A
GPGPU 核心 337A
GPGPU 核心 336A
載入/儲存單元 340B
GPGPU 核心 338B
GPGPU 核心 337B
GPGPU 核心 336B
暫存器檔案 334A
暫存器檔案 334B
指令單元 332A
指令單元 332B
指令快取 330

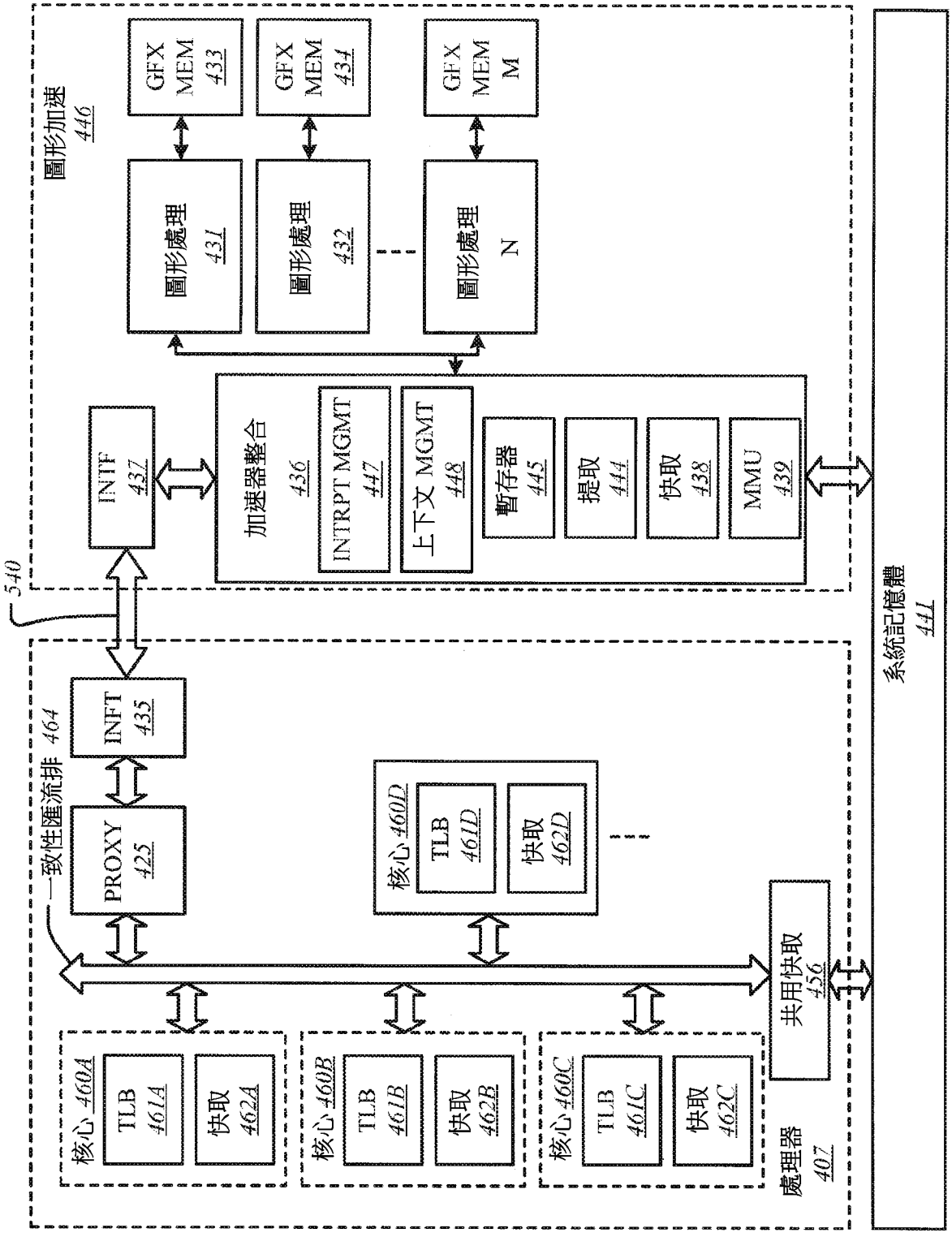
【圖 3A】


圖形多處理器 350
互連組構 352
共用記憶體 362
紋理單元 360A
紋理單元 360B
快取記憶體 358A
執行資源 356A
執行資源 356B
紋理單元 360C
紋理單元 360D
快取記憶體 358B
執行資源 356C
執行資源 356D
指令快取 354

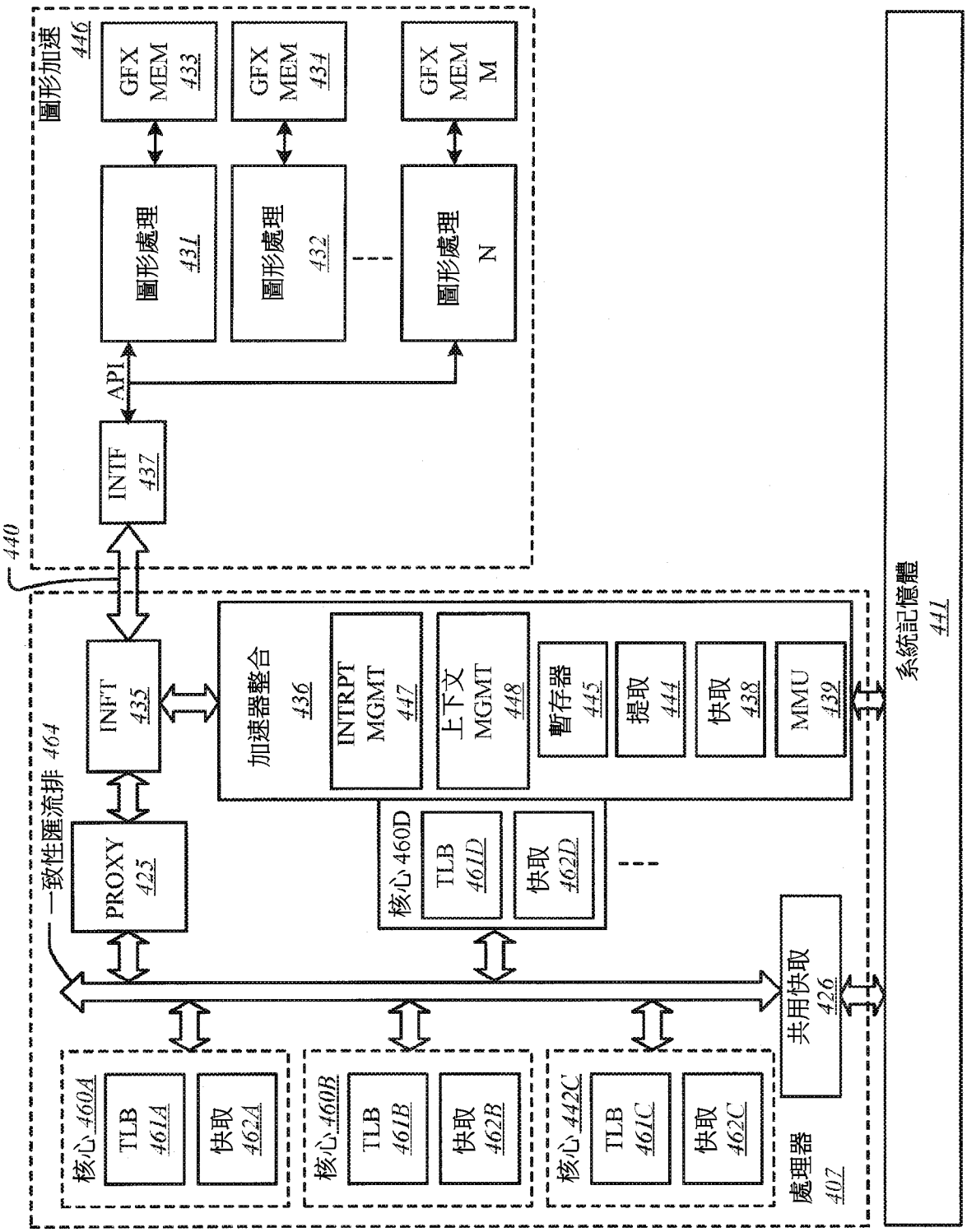
【圖 3B】


430
435
431
處理器
記憶體
401
多核心
處理器
405
多核心
處理器
406
處理器
記憶體
402
440
442
441
443
GPU
記憶體
420
GPU
410
GPU
411
GPU
412
GPU
413
GPU
記憶體
423
444
445
450
451
452
453
GPU
記憶體
421
GPU
記憶體
422

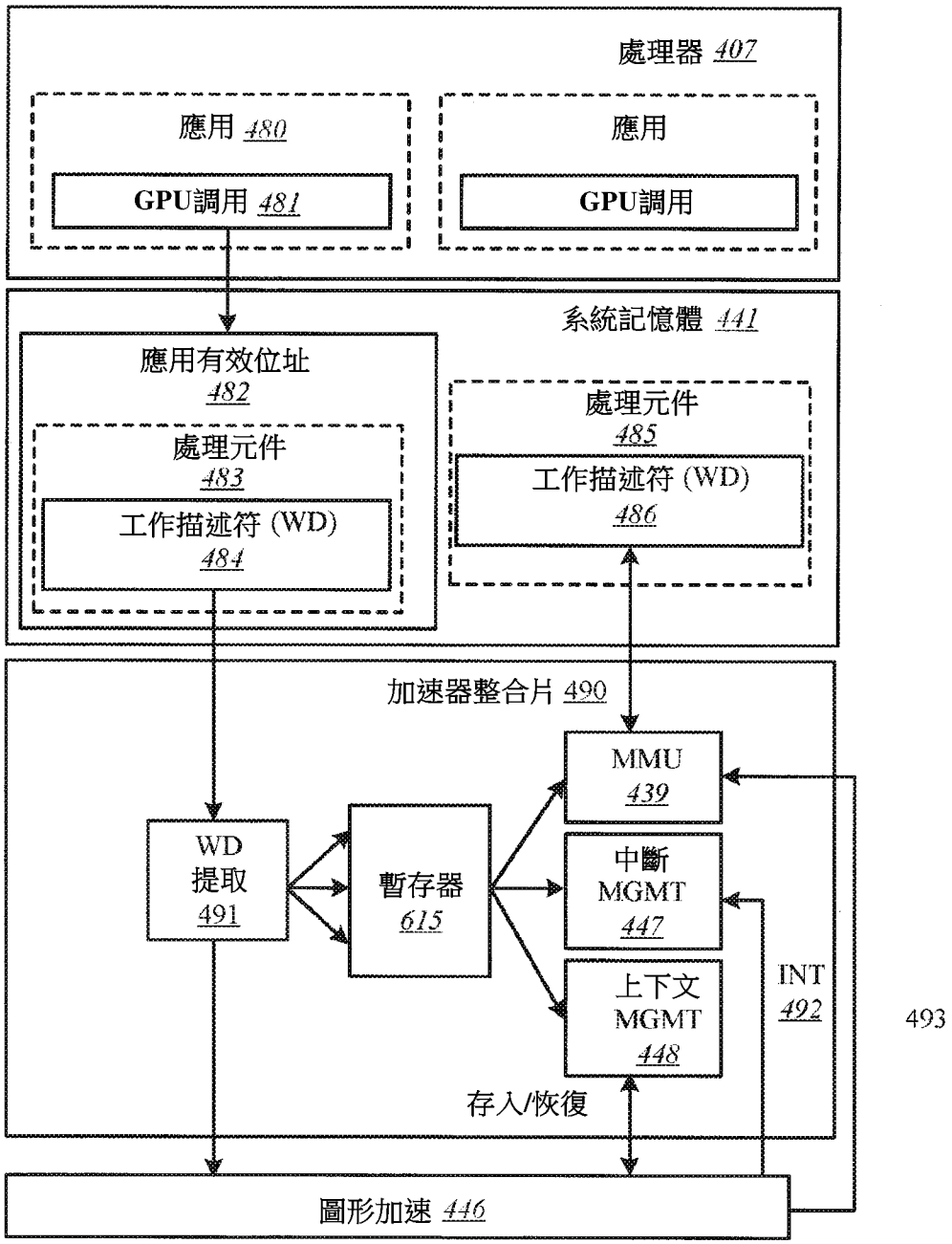
【圖 4A】


圖形加速
446
GFX MEM 433
GFX MEM 434
GFX MEM M
圖形處理 431
圖形處理 432
圖形處理 N
INTF 437
加速器整合 436
INTRPT MGMT 447
上下文 MGMT 448
暫存器 445
提取 444
快取 438
MMU 439
540
一致性匯流排 464
INFT 435
PROXY 425
核心 460D
TLB 461D
快取 462D
核心 460A
TLB 461A
快取 462A
核心 460B
TLB 461B
快取 462B
核心 460C
TLB 461C
快取 462C
共用快取 456
處理器 407
系統記憶體 441

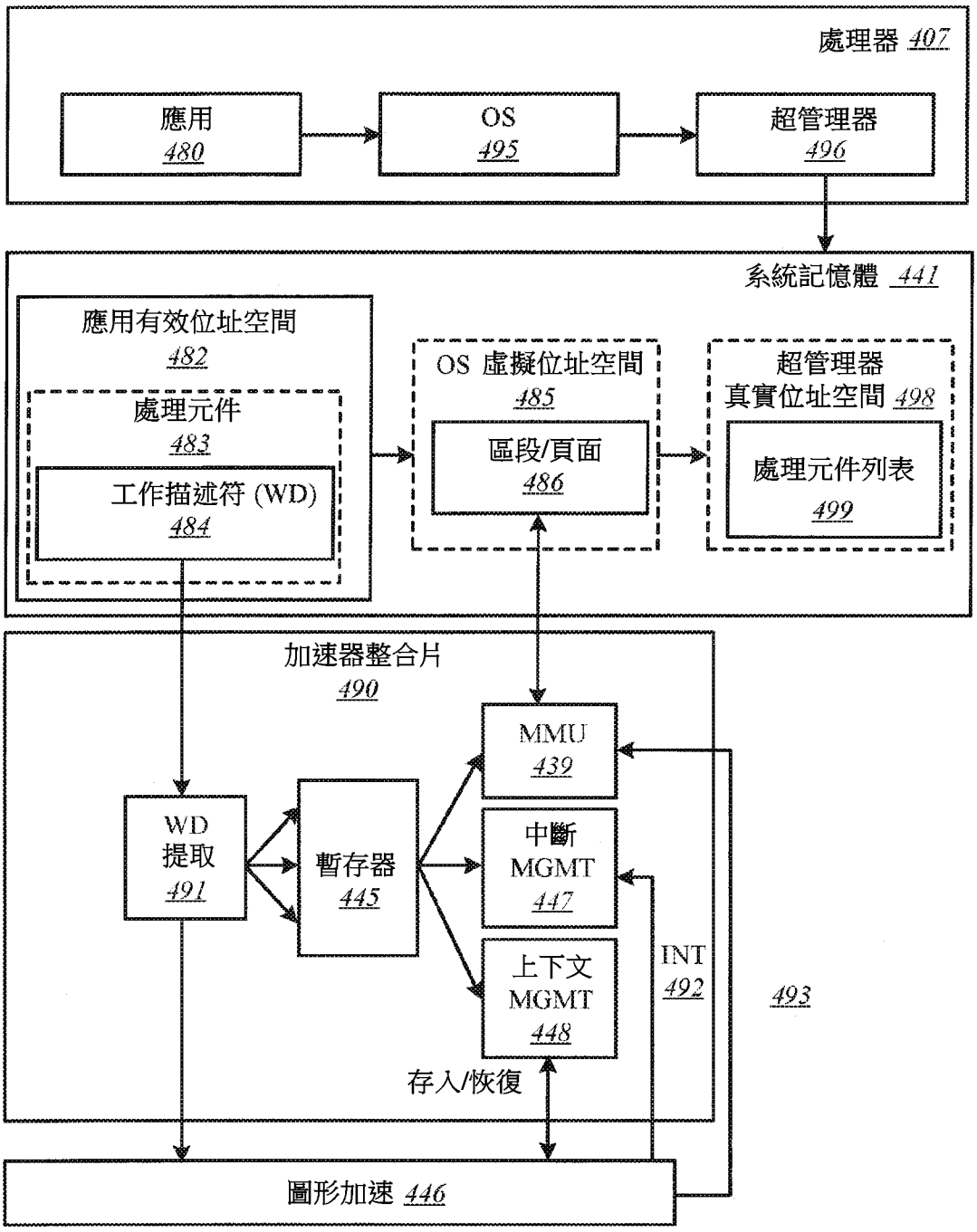
【圖 4B】


圖形加速 446
GFX MEM 433
GFX MEM 434
GFX MEM M
圖形處理 431
圖形處理 432
圖形處理 N
API
INTF 437
440
一致性匯流排 464
INFT 435
PROXY 425
加速器整合 436
INTRPT MGMT 447
上下文 MGMT 448
暫存器 445
提取 444
快取 438
MMU 439
核心 460D
TLB 461D
快取 462D
共用快取 426
核心 460A
TLB 461A
快取 462A
核心 460B
TLB 461B
快取 462B
核心 442C
TLB 461C
快取 462C
處理器 407
系統記憶體 441

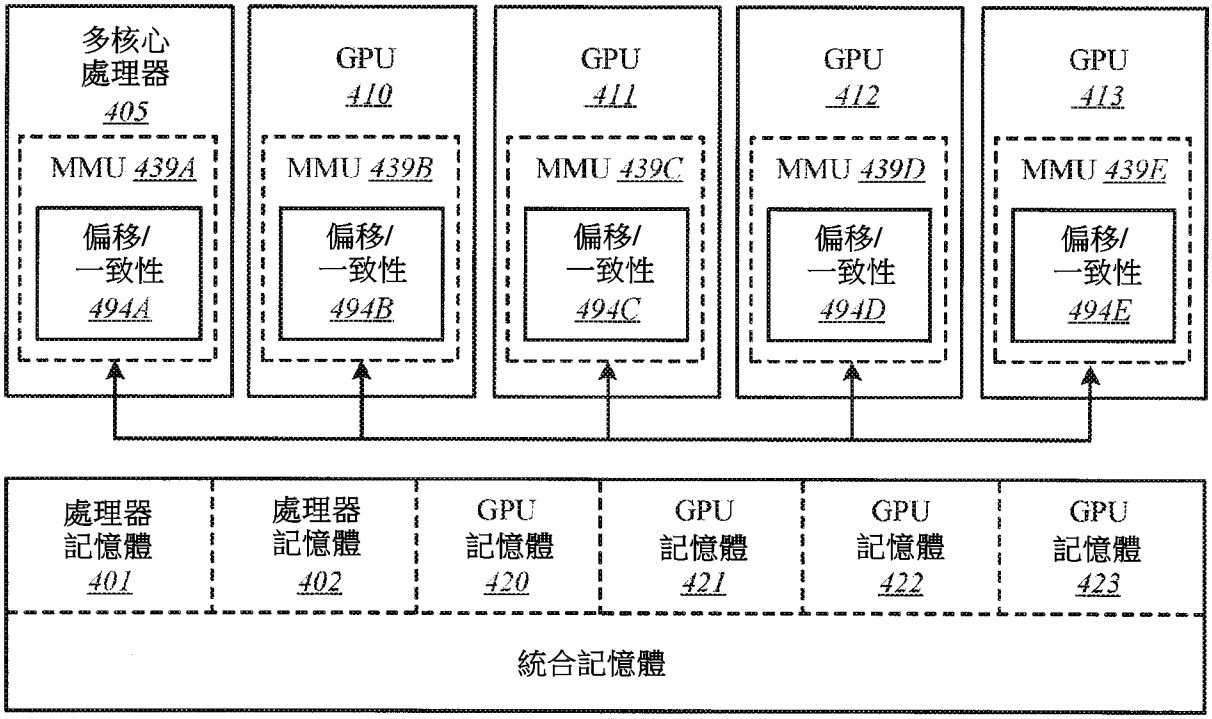
【圖 4C】


處理器 407
應用 480
GPU調用 481
應用
GPU調用
系統記憶體 441
應用有效位址
482
處理元件
483
工作描述符 (WD)
484
處理元件
485
工作描述符 (WD)
486
加速器整合片 490
MMU
439
WD
提取
491
暫存器
615
中斷
MGMT
447
上下文
MGMT
448
INT
492
493
存入/恢復
圖形加速 446

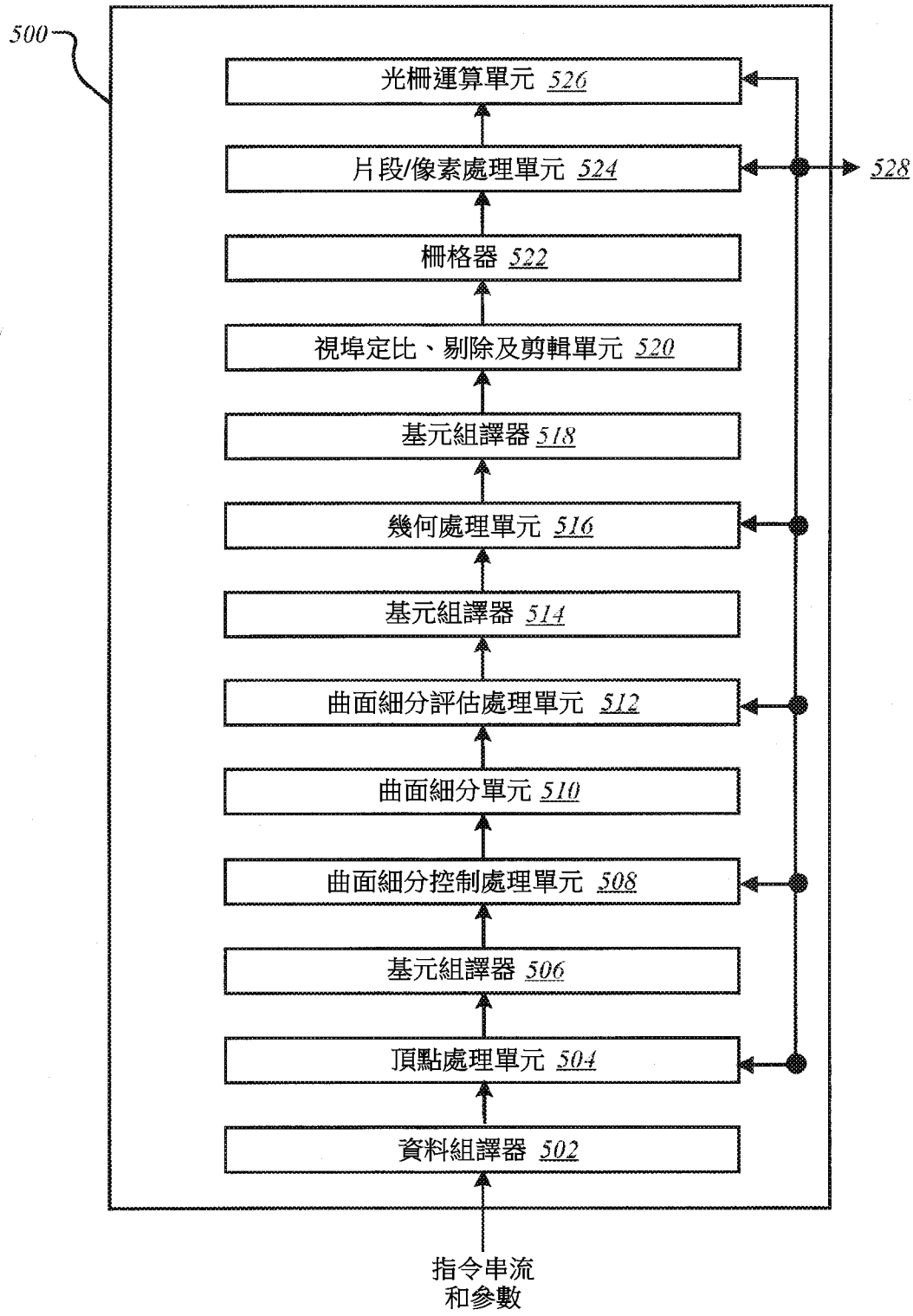
【圖 4D】


處理器 407
應用
480
OS
495
超管理器
496
系統記憶體 441
應用有效位址空間
482
處理元件
483
工作描述符 (WD)
484
OS 虛擬位址空間
485
區段/頁面
486
超管理器
真實位址空間 498
處理元件列表
499
加速器整合片
490
MMU
439
WD
提取
491
暫存器
445
中斷
MGMT
447
上下文
MGMT
448
INT
492
493
存入/恢復
圖形加速 446

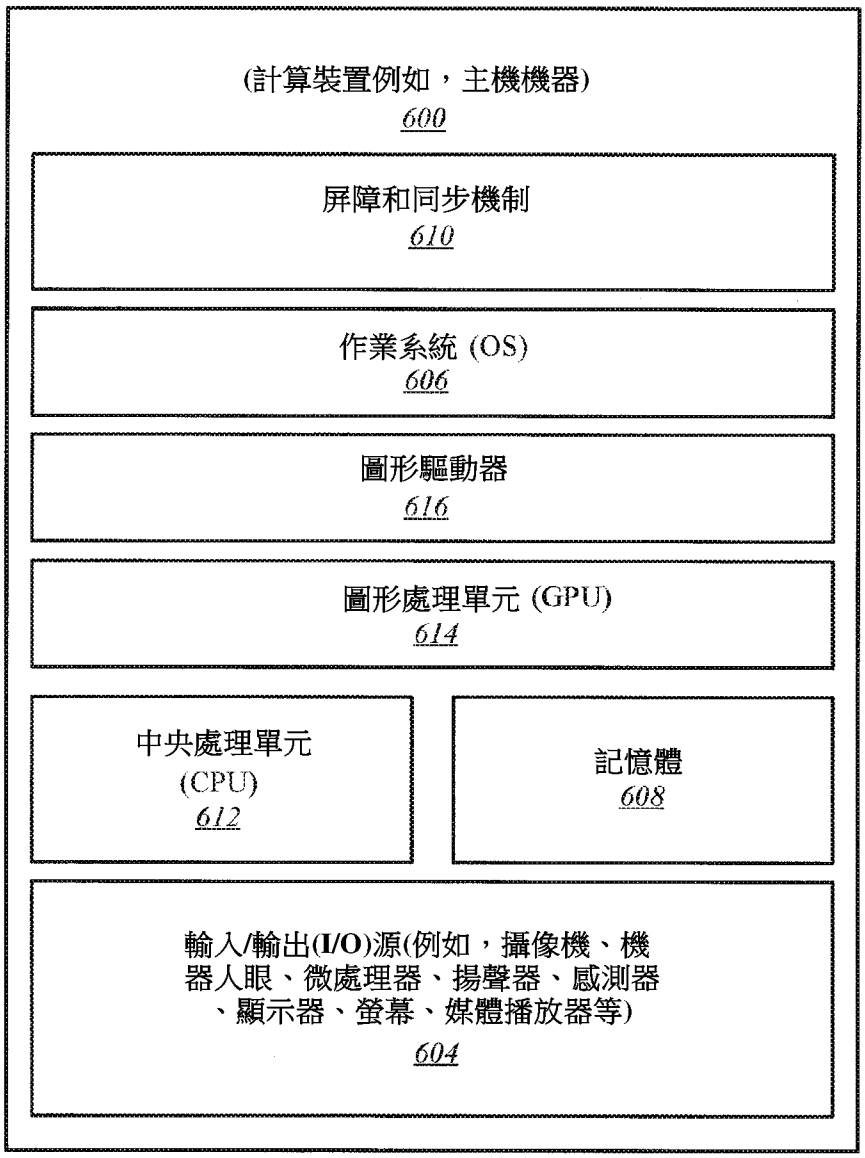
【圖 4E】


多核心
處理器
405
MMU 439A
偏移/
一致性
494A
GPU
410
MMU 439B
偏移/
一致性
494B
GPU
411
MMU 439C
偏移/
一致性
494C
GPU
412
MMU 439D
偏移/
一致性
494D
GPU
413
MMU 439E
偏移/
一致性
494E
處理器
記憶體
401
處理器
記憶體
402
GPU
記憶體
420
GPU
記憶體
421
GPU
記憶體
422
GPU
記憶體
423
統合記憶體

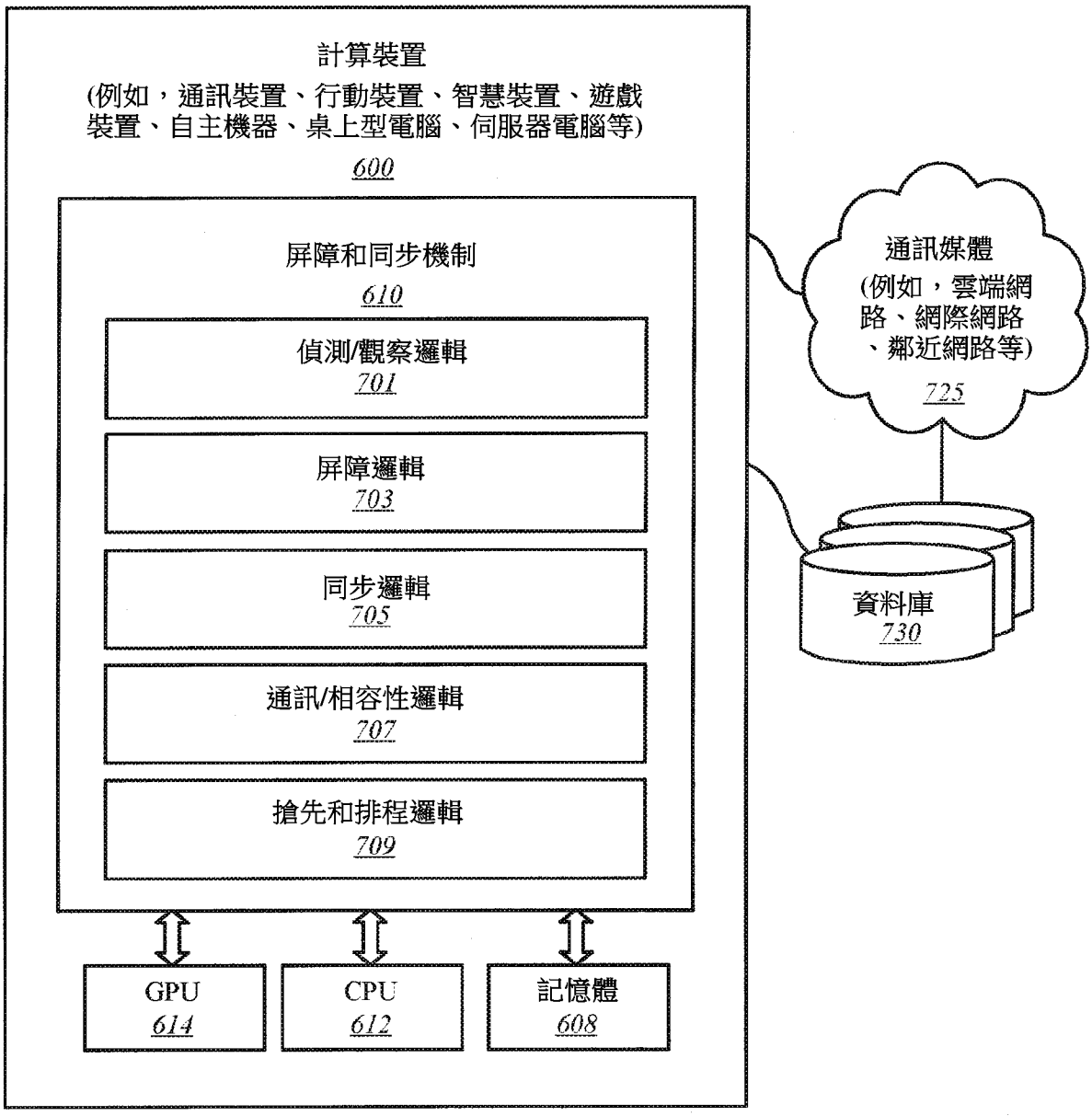
【圖 4F】


500
光柵運算單元 526
片段/像素處理單元 524
528
柵格器 522
視埠定比、剔除及剪輯單元 520
基元組譯器 518
幾何處理單元 516
基元組譯器 514
曲面細分評估處理單元 512
曲面細分單元 510
曲面細分控制處理單元 508
基元組譯器 506
頂點處理單元 504
資料組譯器 502
指令串流
和參數

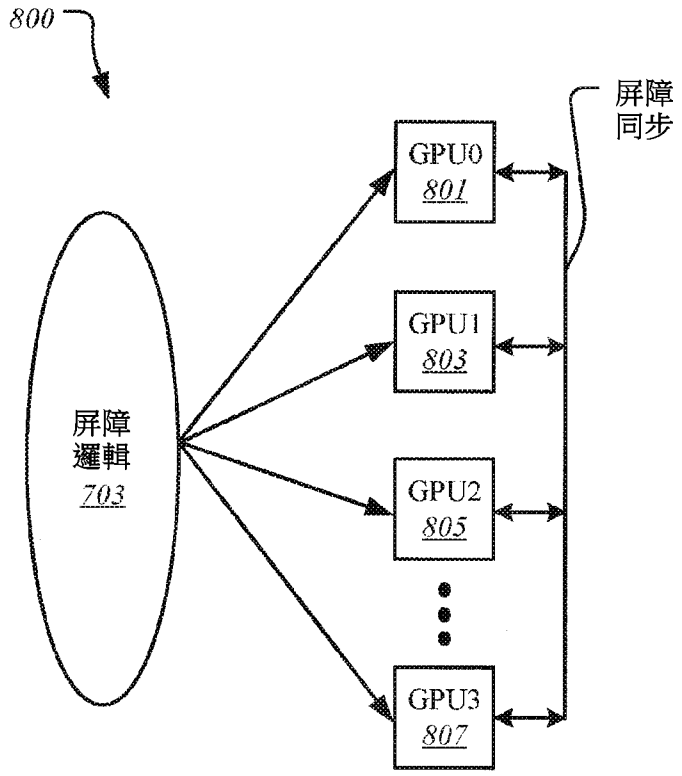
【圖 5】


(計算裝置例如，主機機器)
600
屏障和同步機制
610
作業系統 (OS)
606
圖形驅動器
616
圖形處理單元 (GPU)
614
中央處理單元
(CPU)
612
記憶體
608
輸入/輸出(I/O)源(例如，攝像機、機器人眼、微處理器、揚聲器、感測器、顯示器、螢幕、媒體播放器等)
604

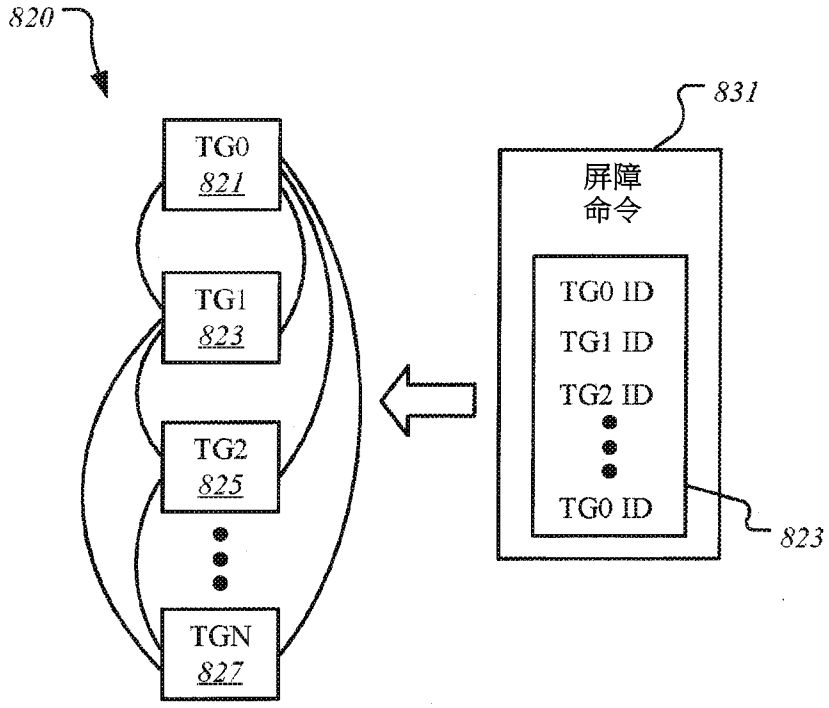
【圖 6】


計算裝置
(例如，通訊裝置、行動裝置、智慧裝置、遊戲裝置、自主機器、桌上型電腦、伺服器電腦等)
600
屏障和同步機制
610
偵測/觀察邏輯
701
屏障邏輯
703
同步邏輯
705
通訊/相容性邏輯
707
搶先和排程邏輯
709
GPU
614
CPU
612
記憶體
608
通訊媒體
(例如，雲端網路、網際網路、鄰近網路等)
725
資料庫
730

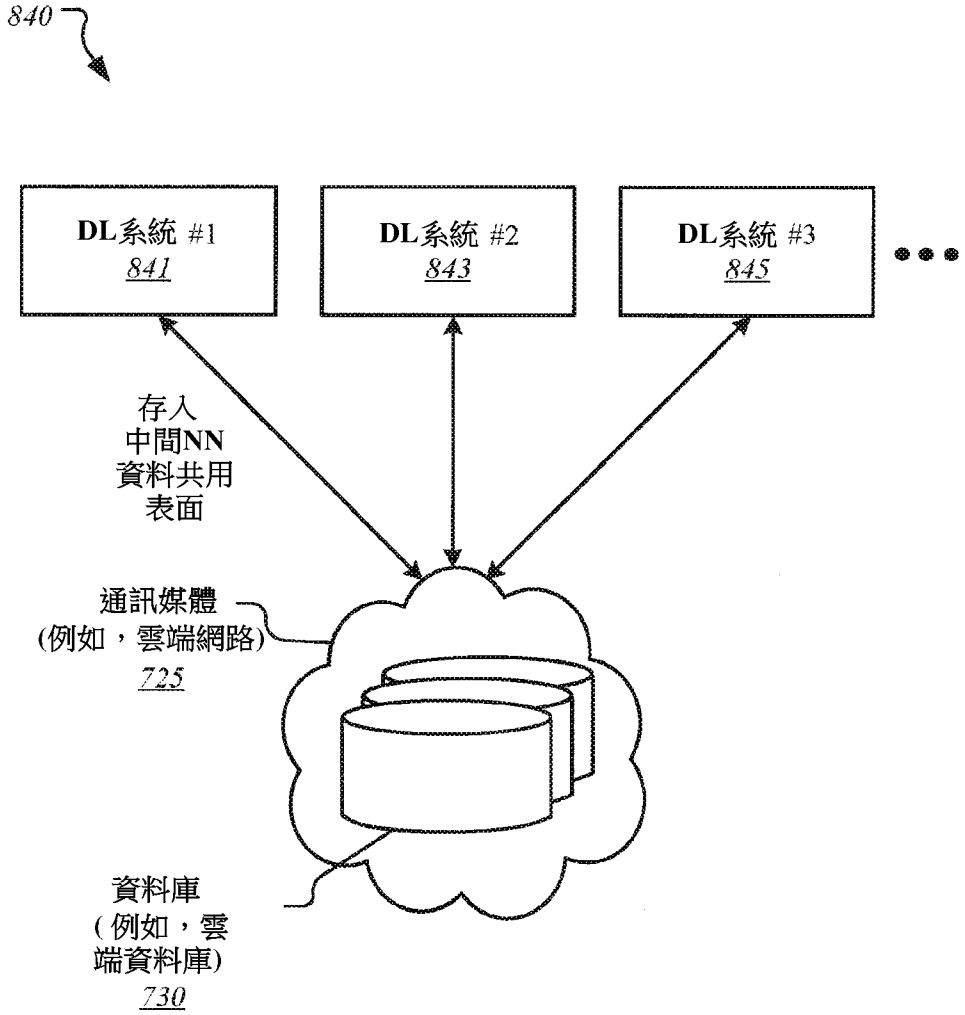
【圖 7】



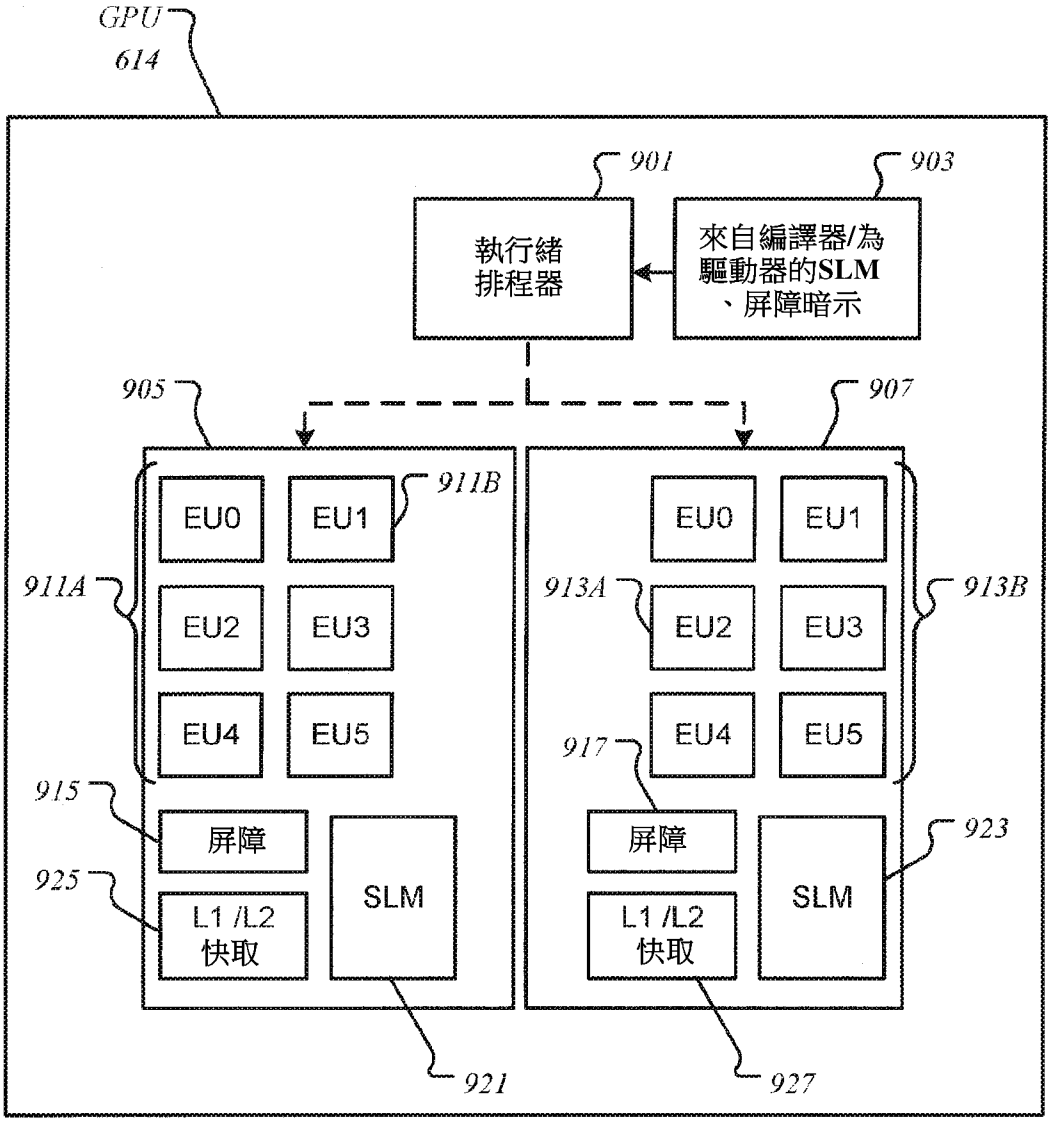
【圖 8A】


820
831
TG0
821
TG1
823
TG2
825
TGN
827
屏障
命令
TG0 ID
TG1 ID
TG2 ID
TG0 ID
823

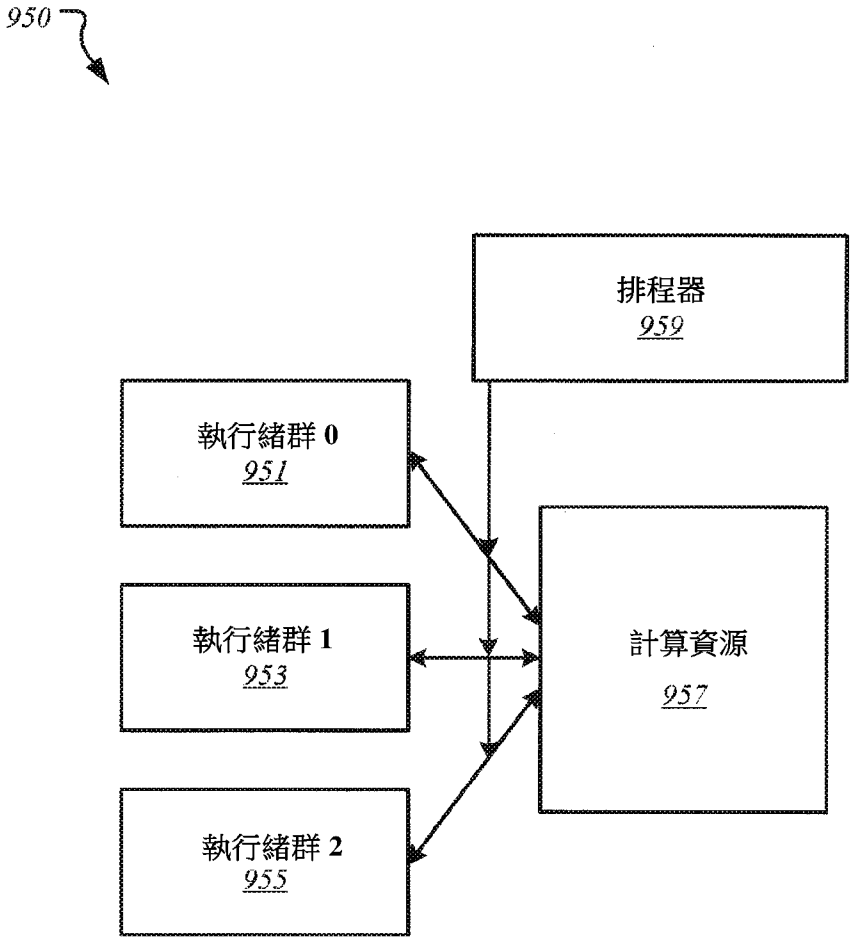
【圖 8B】


840
DL系統 #1
841
DL系統 #2
843
DL系統 #3
845
存入
中間NN
資料共用
表面
通訊媒體
(例如，雲端網路)
725
資料庫
(例如，雲
端資料庫)
730

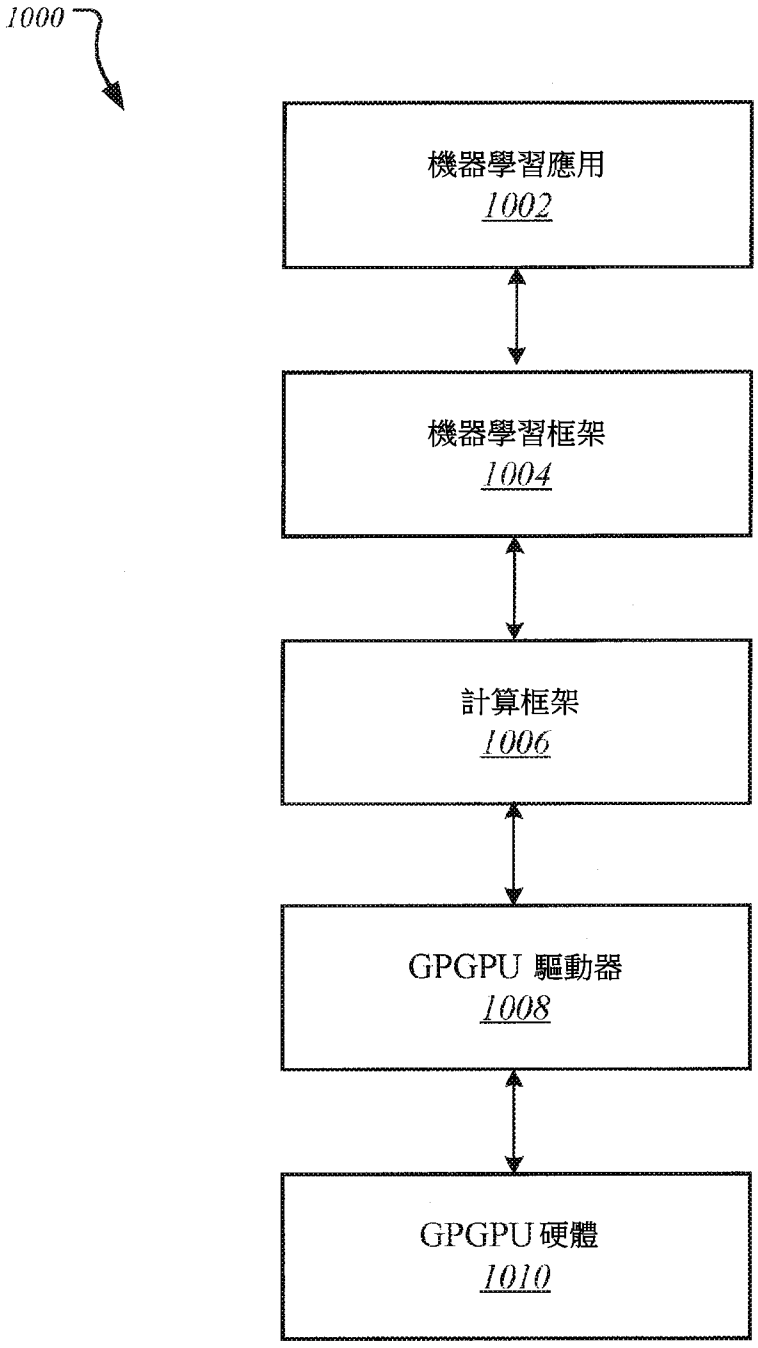
【圖 8C】


GPU
614
901
903
執行緒
排程器
來自編譯器/為
驅動器的SLM
、屏障暗示
905
907
911B
911A
913A
913B
EU0
EU1
EU2
EU3
EU4
EU5
915
917
屏障
925
L1 /L2
快取
SLM
923
921
927

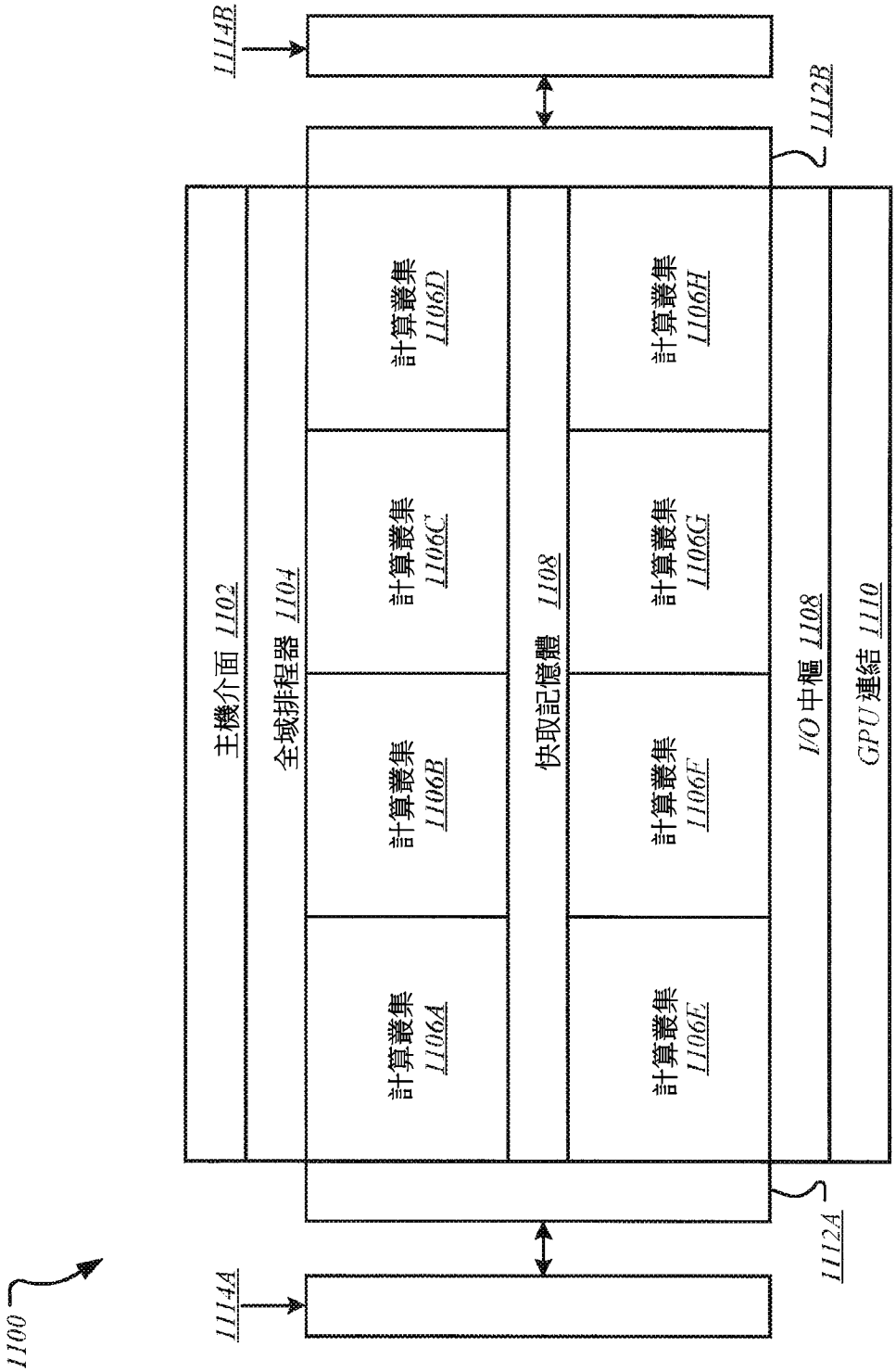
【圖 9A】



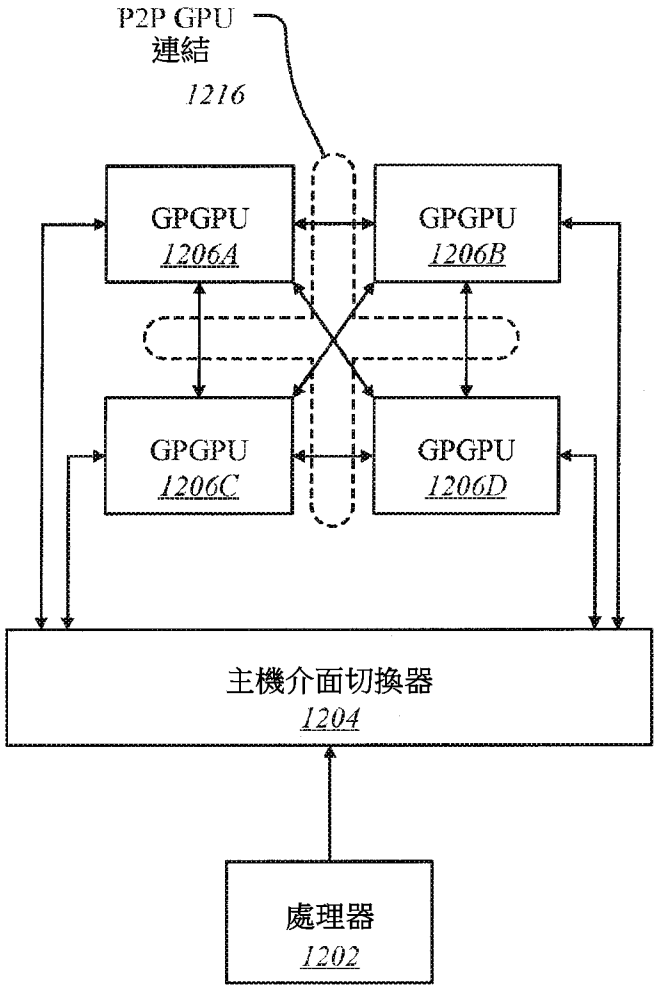
【圖 9B】


1000
機器學習應用
1002
機器學習框架
1004
計算框架
1006
GPGPU 驅動器
1008
GPGPU 硬體
1010

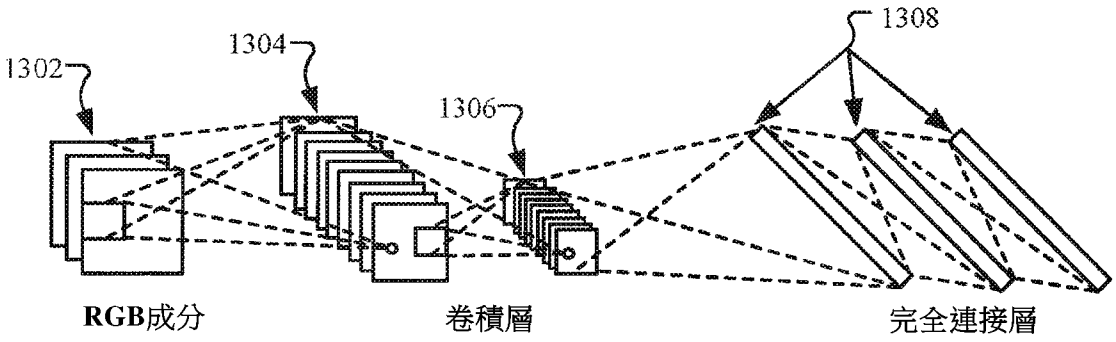
【圖 10】


1100
1114A
1114B
1112A
1112B
主機介面 1102
全域排程器 1104
計算叢集 1106A
計算叢集 1106B
計算叢集 1106C
計算叢集 1106D
計算叢集 1106E
計算叢集 1106F
計算叢集 1106G
計算叢集 1106H
快取記憶體 1108
I/O 中樞 1108
GPU 連結 1110

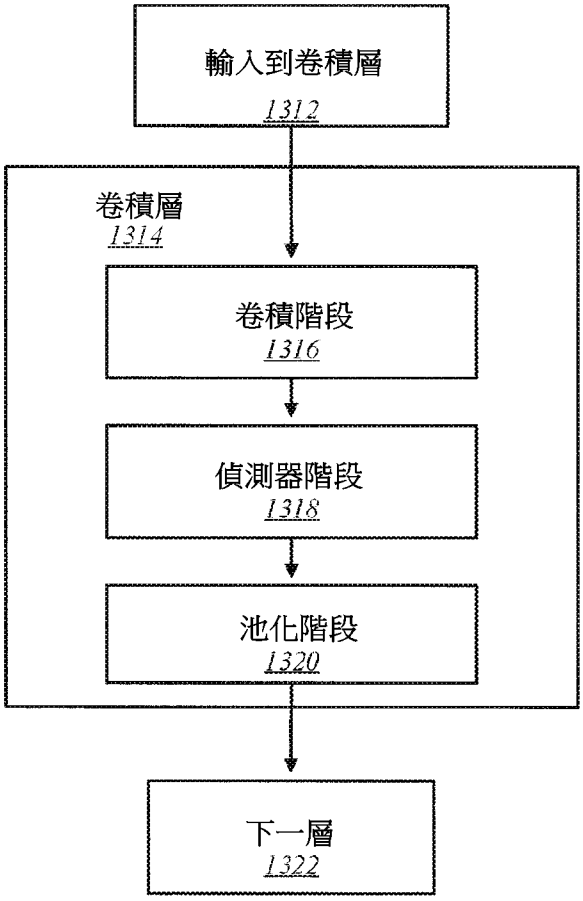
【圖 11】


P2P GPU
連結
1216
GPGPU
1206A
GPGPU
1206B
GPGPU
1206C
GPGPU
1206D
主機介面切換器
1204
處理器
1202

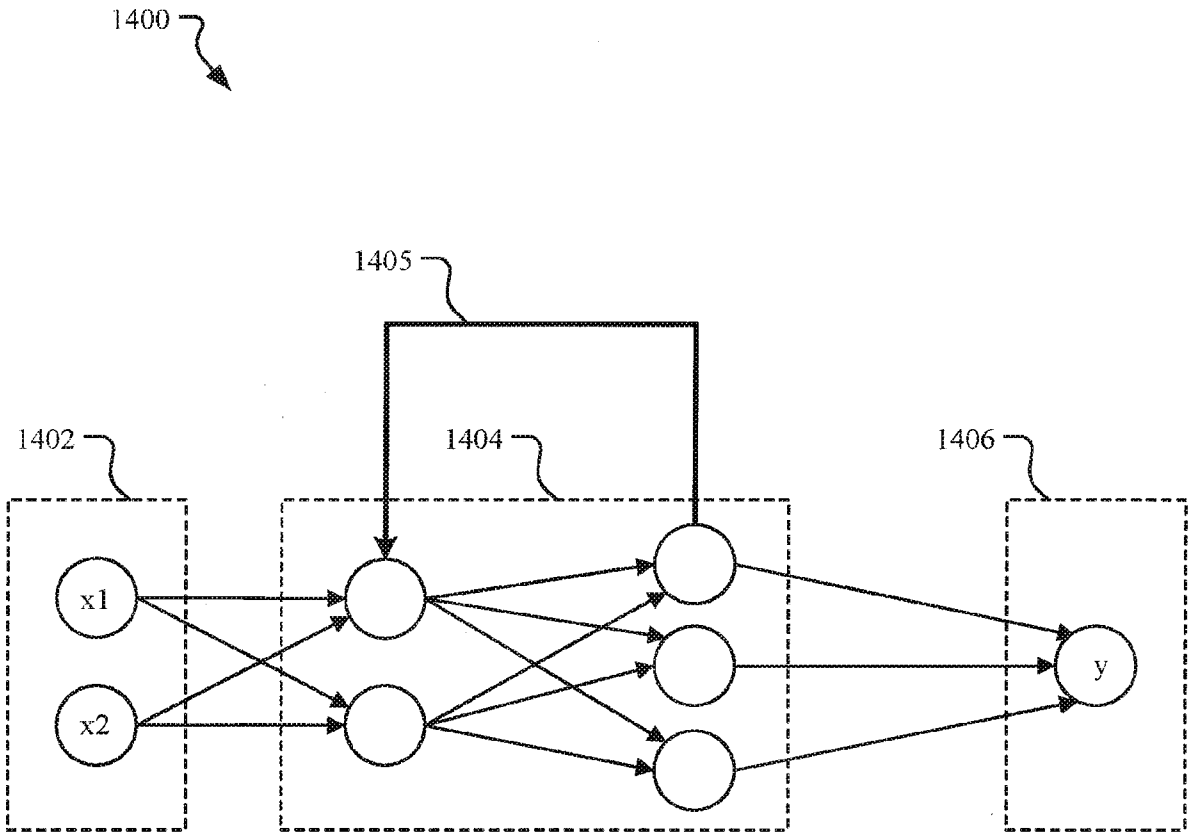
【圖 12】


1302
1304
1306
1308
RGB成分
卷積層
完全連接層

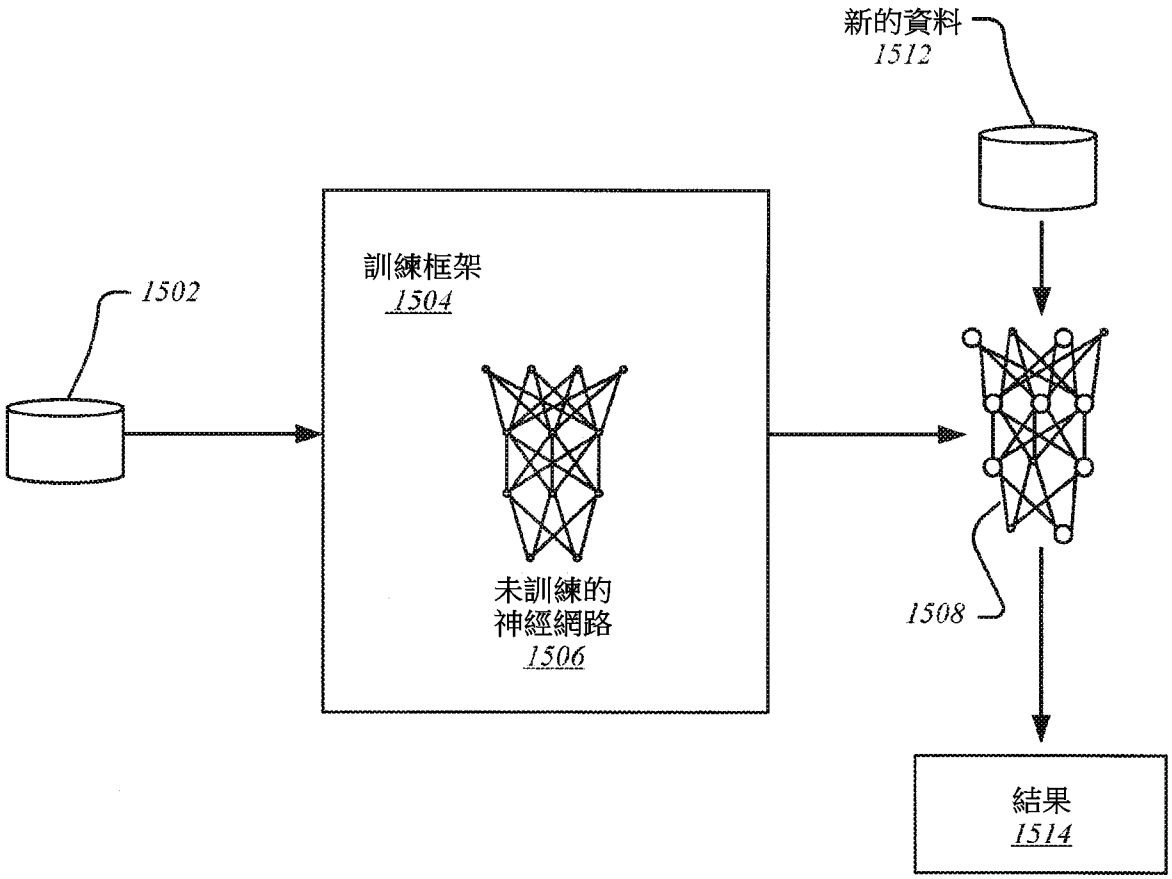
【圖 13A】


輸入到卷積層
1312
卷積層
1314
卷積階段
1316
偵測器階段
1318
池化階段
1320
下一層
1322

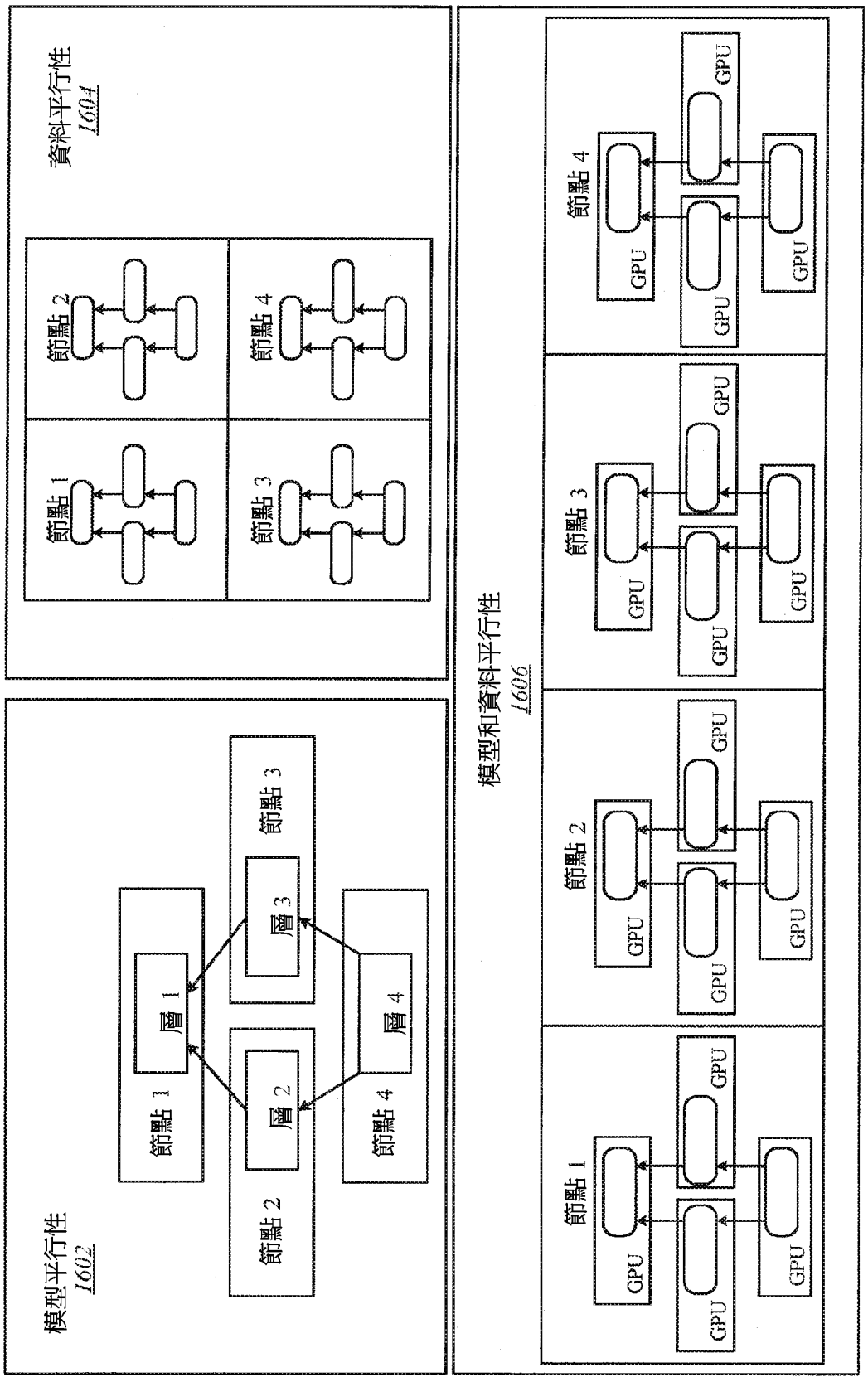
【圖 13B】



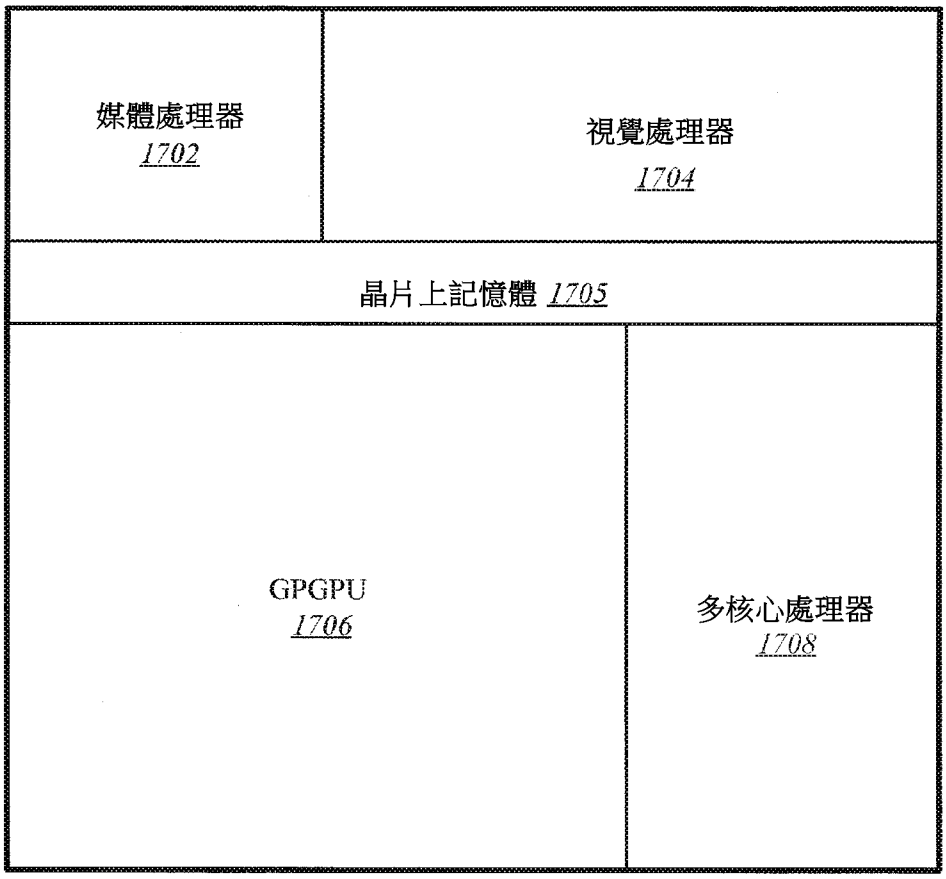
【圖 14】



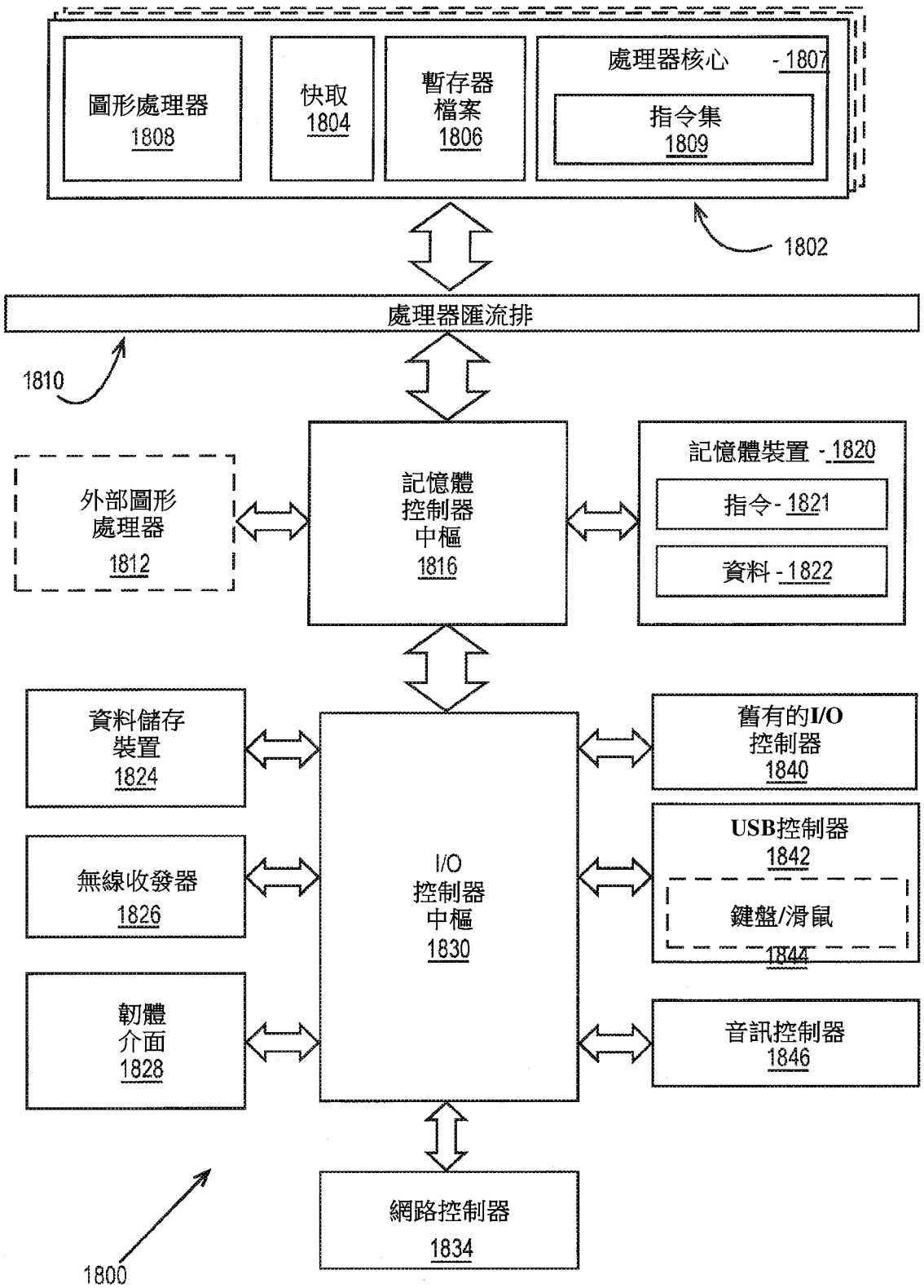
【圖 15】


模型平行性
1602
節點 1
層 1
節點 2
層 2
節點 3
層 3
節點 4
層 4
資料平行性
1604
節點 1
節點 2
節點 3
節點 4
模型和資料平行性
1606
節點 1
節點 2
節點 3
節點 4
GPU

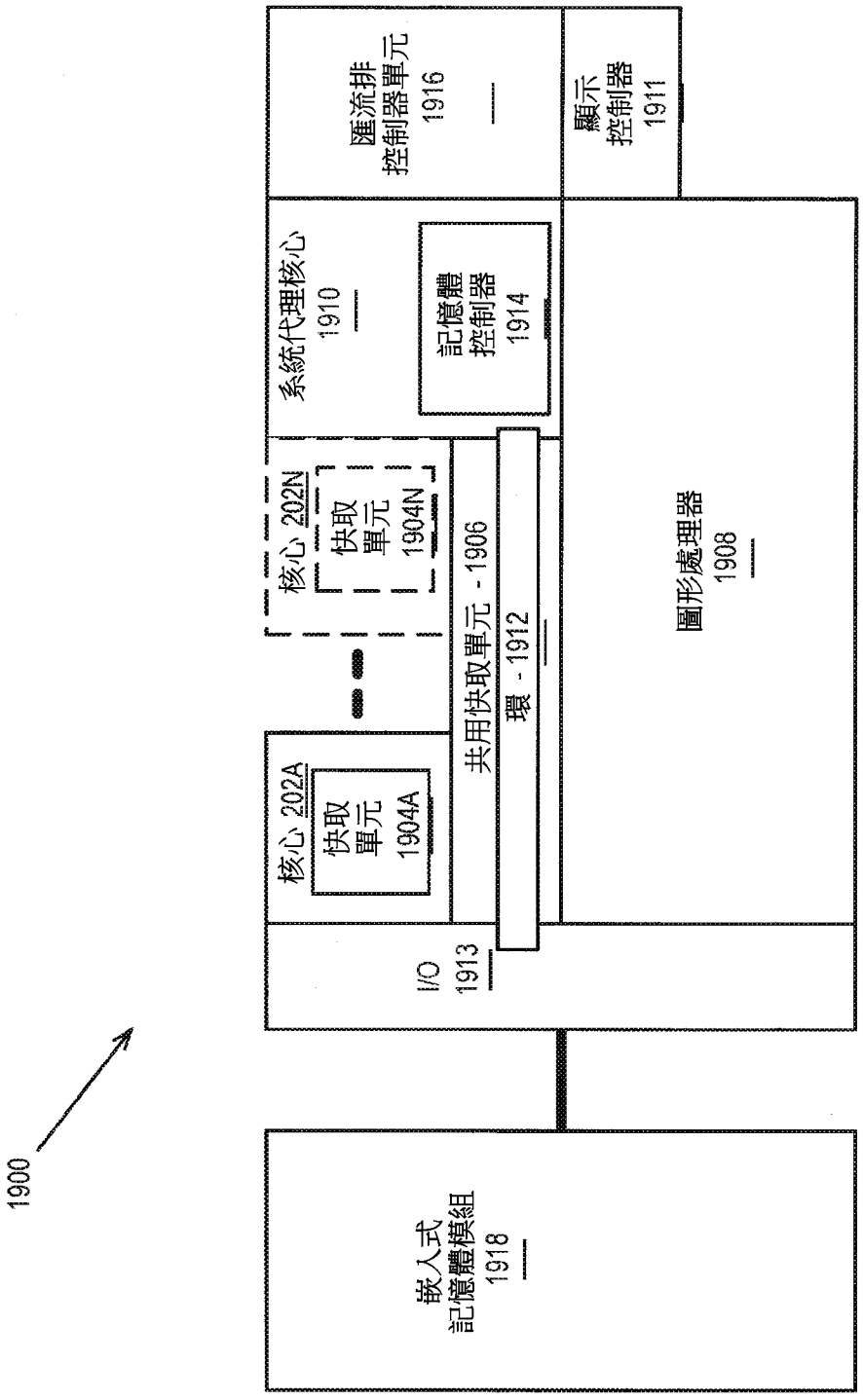
【圖 16】



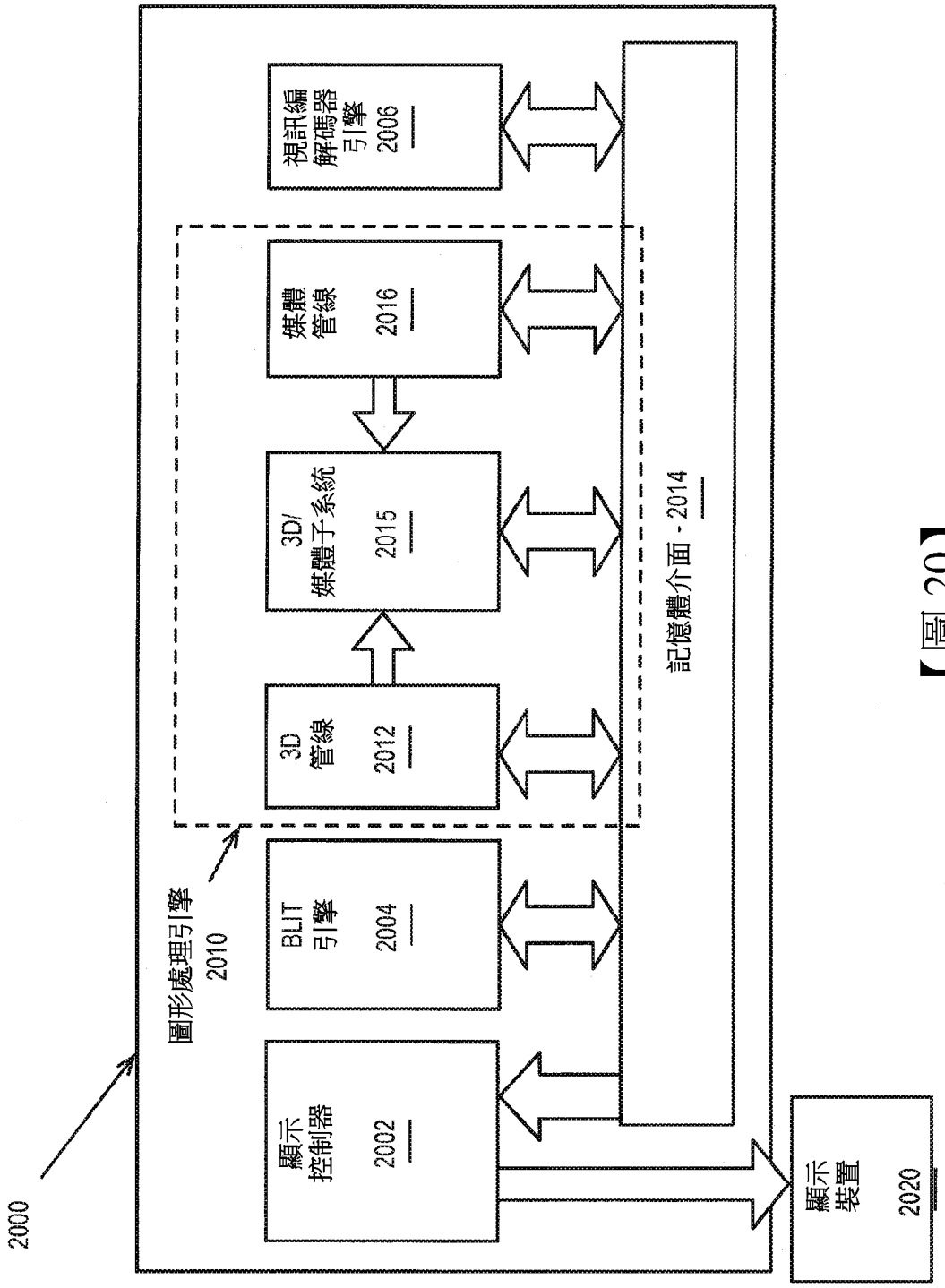
【圖 17】


圖形處理器
1808
快取
1804
暫存器
檔案
1806
處理器核心 - 1807
指令集
1809
1802
處理器匯流排
1810
外部圖形
處理器
1812
記憶體
控制器
中樞
1816
記憶體裝置 - 1820
指令 - 1821
資料 - 1822
資料儲存
裝置
1824
無線收發器
1826
韌體
介面
1828
I/O
控制器
中樞
1830
舊有的I/O
控制器
1840
USB控制器
1842
鍵盤/滑鼠
1844
音訊控制器
1846
網路控制器
1834
1800

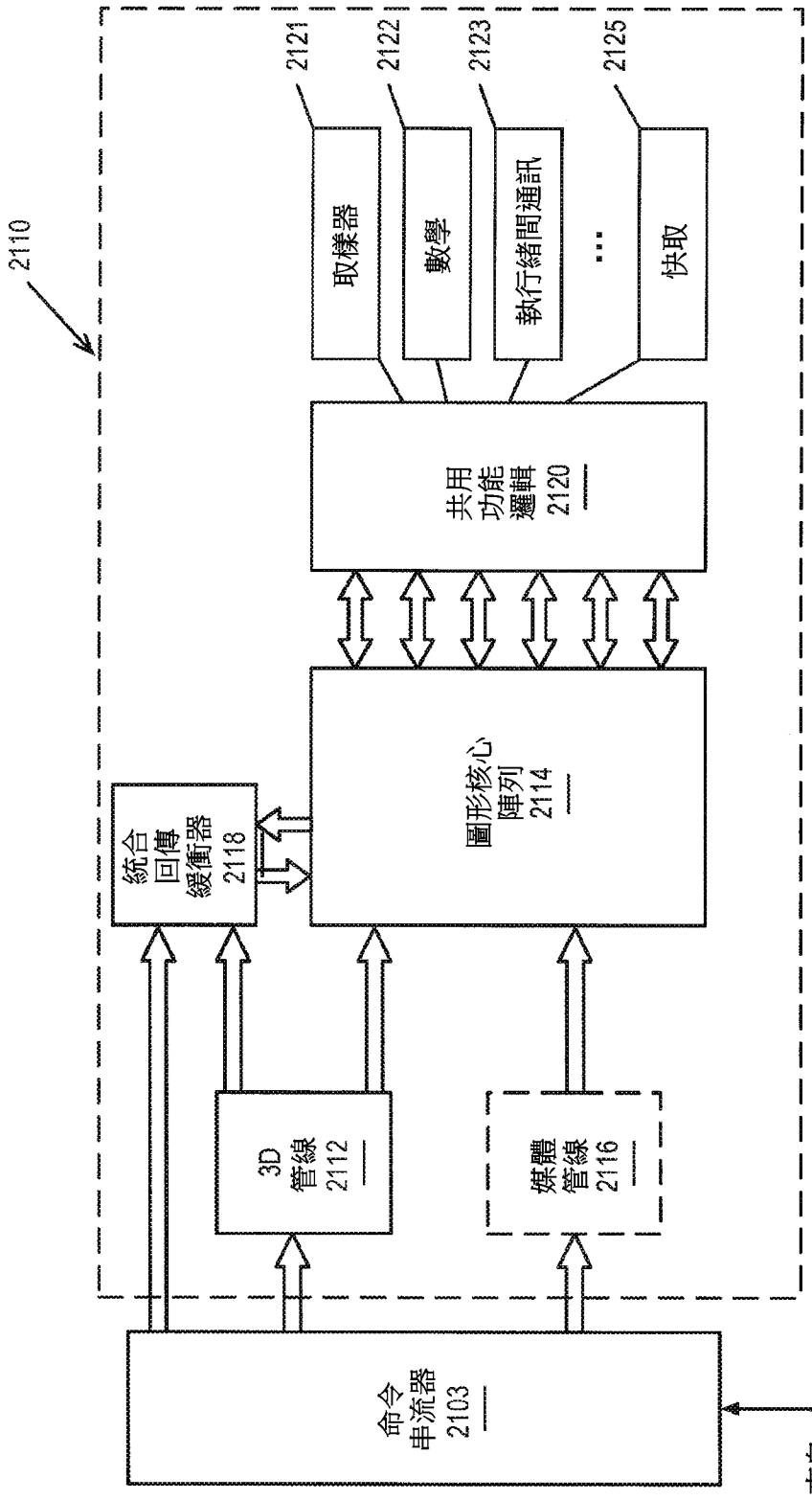
【圖 18】


1900
嵌入式
記憶體模組
1918
I/O
1913
核心 202A
快取
單元
1904A
核心 202N
快取
單元
1904N
共用快取單元 - 1906
環 - 1912
系統代理核心
1910
記憶體
控制器
1914
圖形處理器
1908
匯流排
控制器單元
1916
顯示
控制器
1911

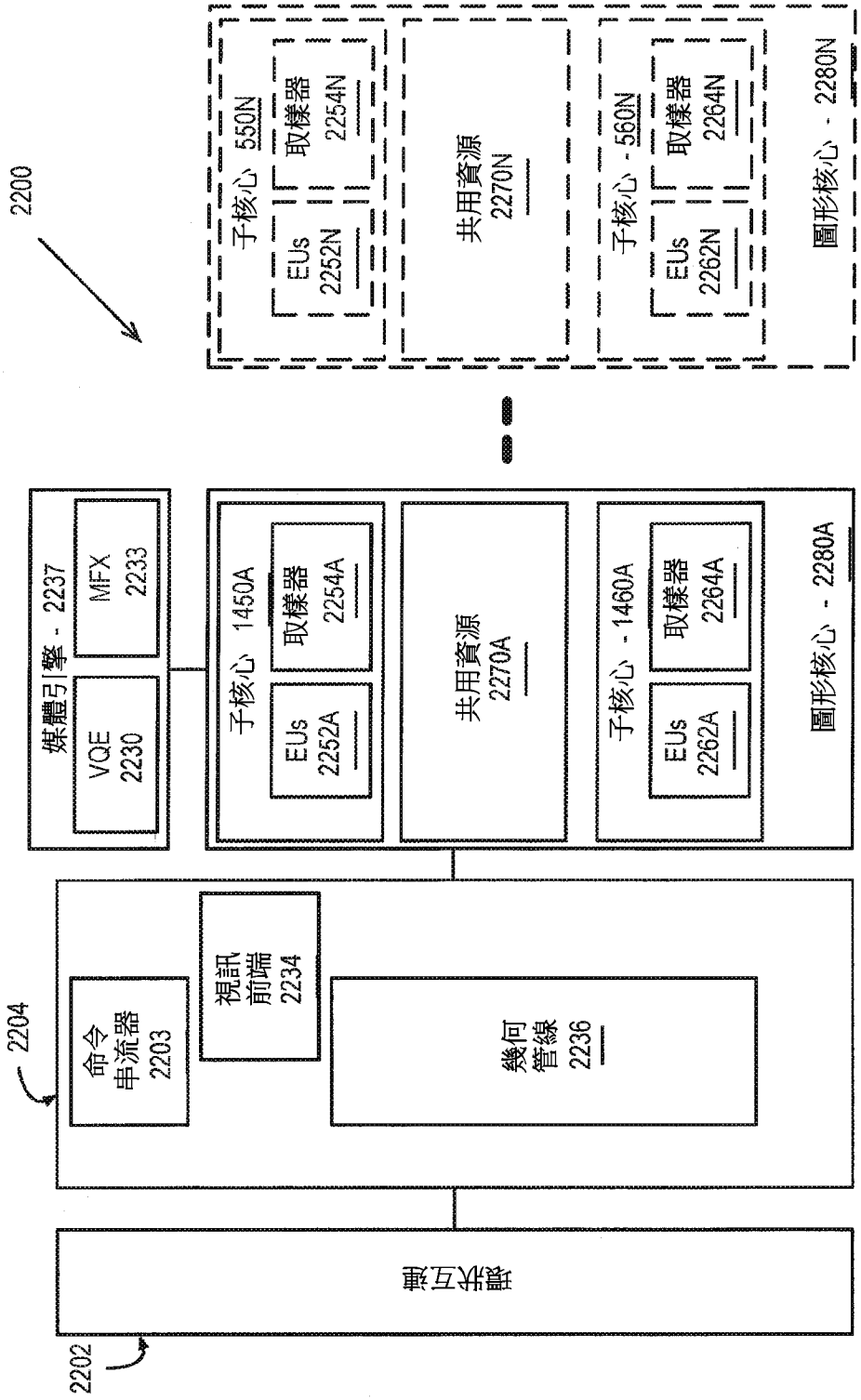
【圖 19】


2000
圖形處理引擎
2010
顯示
控制器
2002
BLIT
引擎
2004
3D
管線
2012
3D/
媒體子系統
2015
媒體
管線
2016
視訊編
解碼器
引擎
2006
記憶體介面 - 2014
顯示
裝置
2020

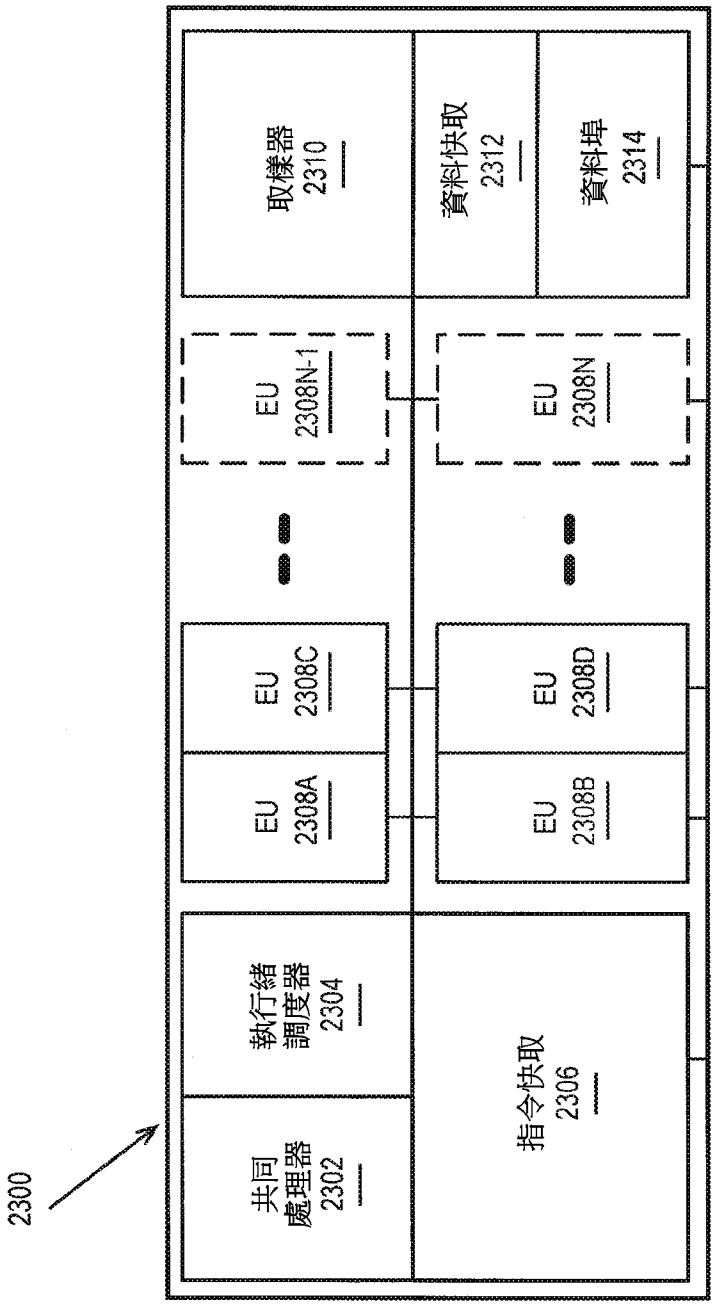
【圖 20】


2110
2121
2122
2123
2125
取樣器
數學
執行緒間通訊
...
快取
共用功能邏輯 2120
圖形核心陣列 2114
統合回傳緩衝器 2118
3D 管線 2112
媒體管線 2116
命令串流器 2103
來自記憶體

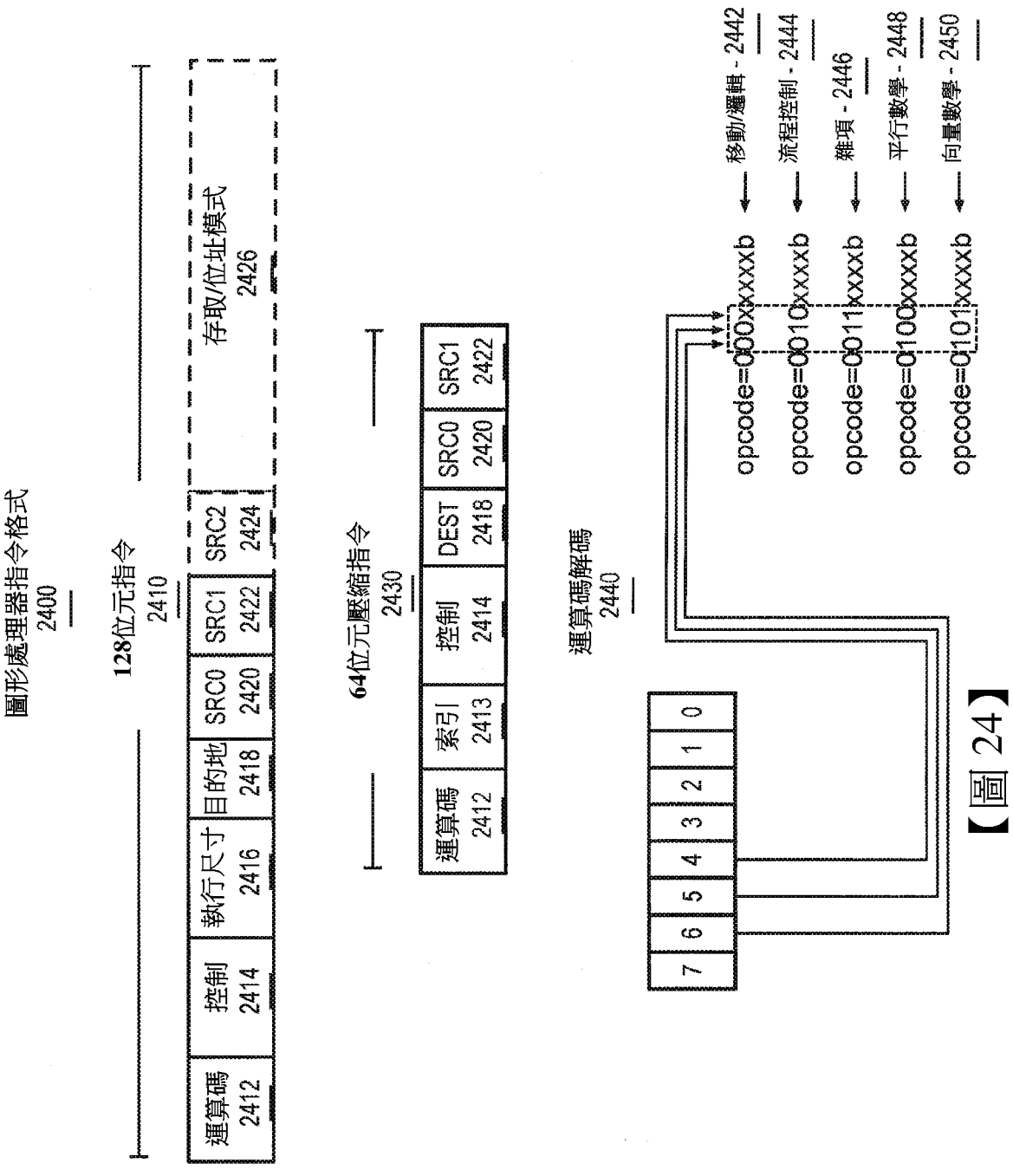
【圖 21】


2200
子核心 550N
EUs 2252N
取樣器 2254N
共用資源 2270N
子核心 - 560N
EUs 2262N
取樣器 2264N
圖形核心 - 2280N
媒體引擎 - 2237
VQE 2230
MFX 2233
子核心 1450A
EUs 2252A
取樣器 2254A
共用資源 2270A
子核心 - 1460A
EUs 2262A
取樣器 2264A
圖形核心 - 2280A
2204
命令串流器 2203
視訊前端 2234
幾何管線 2236
2202

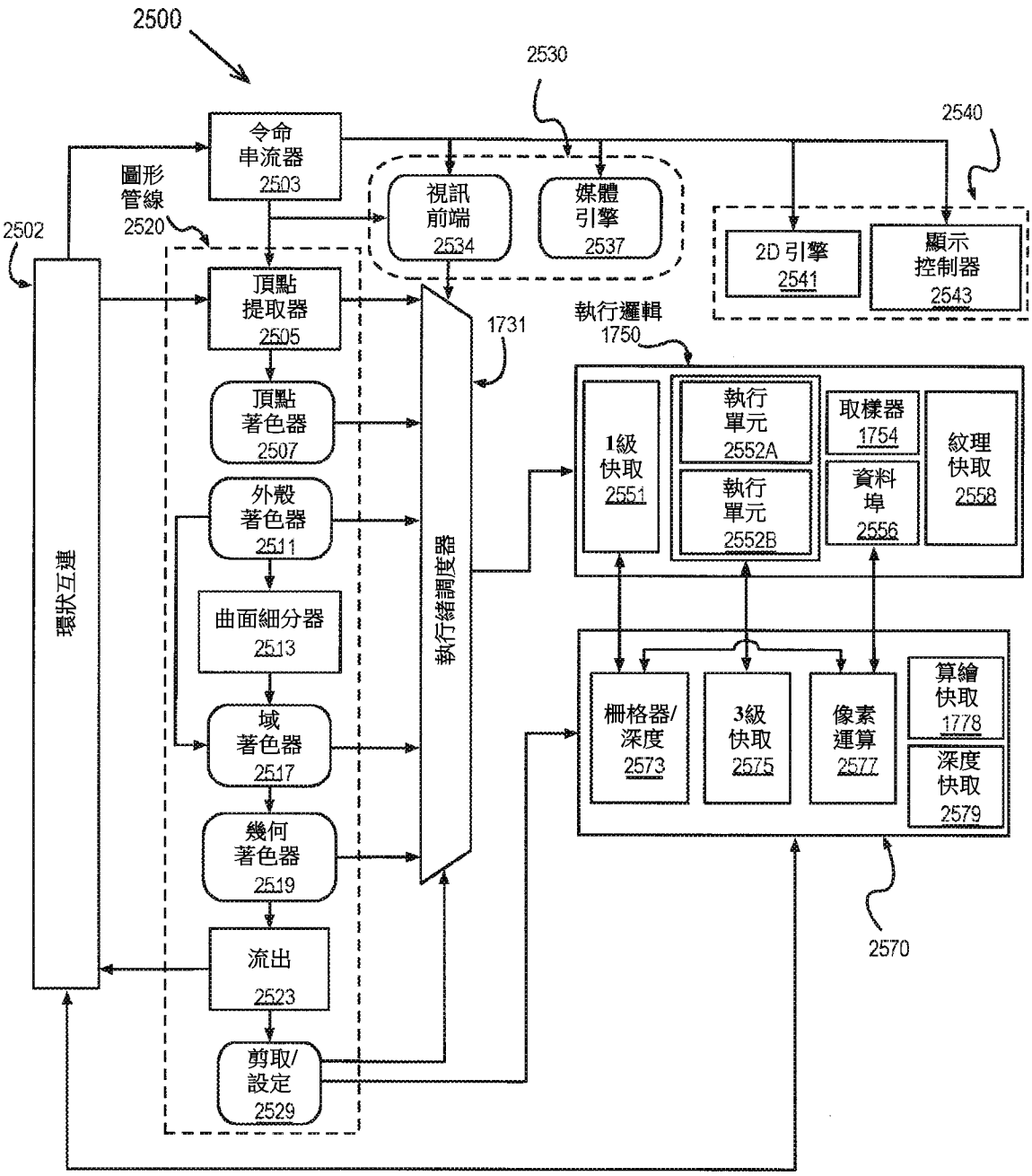
【圖 22】


2300
共同處理器 2302
執行緒調度器 2304
指令快取 2306
EU 2308A
EU 2308C
EU 2308B
EU 2308D
EU 2308N-1
EU 2308N
取樣器 2310
資料快取 2312
資料埠 2314

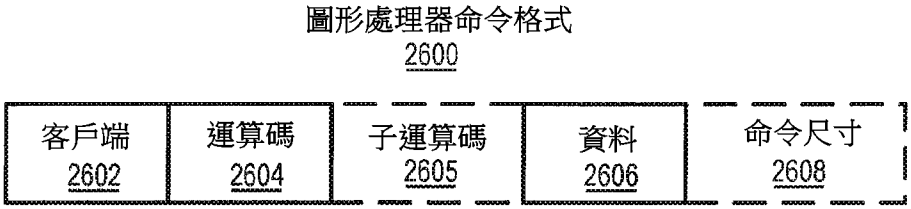
【圖 23】


圖形處理器指令格式
2400
128位元指令
2410
運算碼 2412
控制 2414
執行尺寸 2416
目的地 2418
SRC0 2420
SRC1 2422
SRC2 2424
存取/位址模式 2426
64位元壓縮指令
2430
運算碼 2412
索引 2413
控制 2414
DEST 2418
SRC0 2420
SRC1 2422
運算碼解碼
2440
7 6 5 4 3 2 1 0
opcode=000xxxxxb
opcode=001xxxxxb
opcode=010xxxxxb
opcode=011xxxxxb
opcode=100xxxxxb
opcode=101xxxxxb
移動/邏輯 - 2442
流程控制 - 2444
雜項 - 2446
平行數學 - 2448
向量數學 - 2450

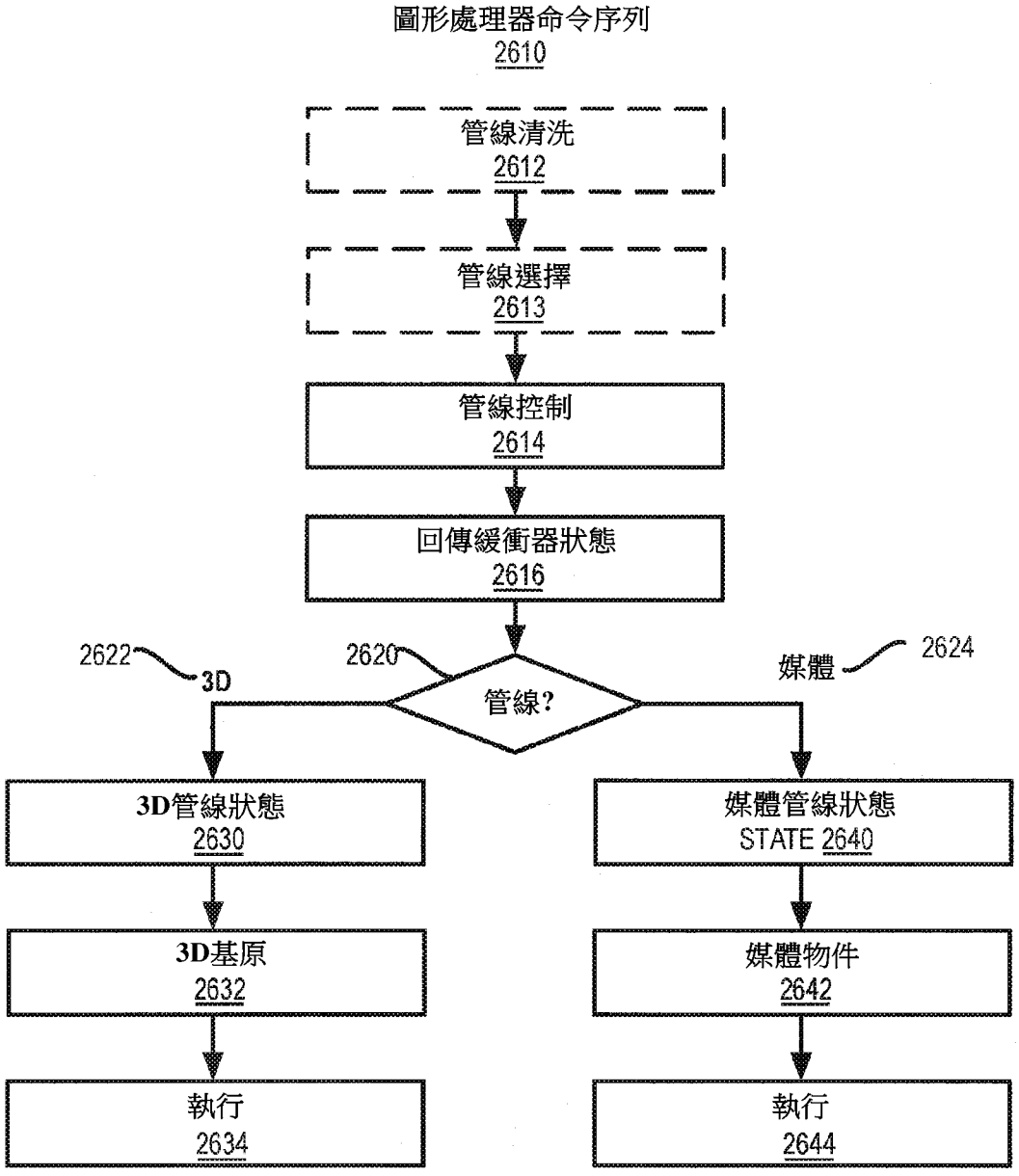
【圖 24】


2500
2530
2540
令命
串流器
2503
圖形
管線
2520
2502
視訊
前端
2534
媒體
引擎
2537
2D 引擎
2541
顯示
控制器
2543
頂點
提取器
2505
1731
執行邏輯
1750
頂點
著色器
2507
外殼
著色器
2511
曲面細分器
2513
域
著色器
2517
幾何
著色器
2519
流出
2523
剪取/
設定
2529
環狀互連
執行緒調度器
1級
快取
2551
執行
單元
2552A
執行
單元
2552B
取樣器
1754
資料
埠
2556
紋理
快取
2558
柵格器/
深度
2573
3級
快取
2575
像素
運算
2577
算繪
快取
1778
深度
快取
2579
2570

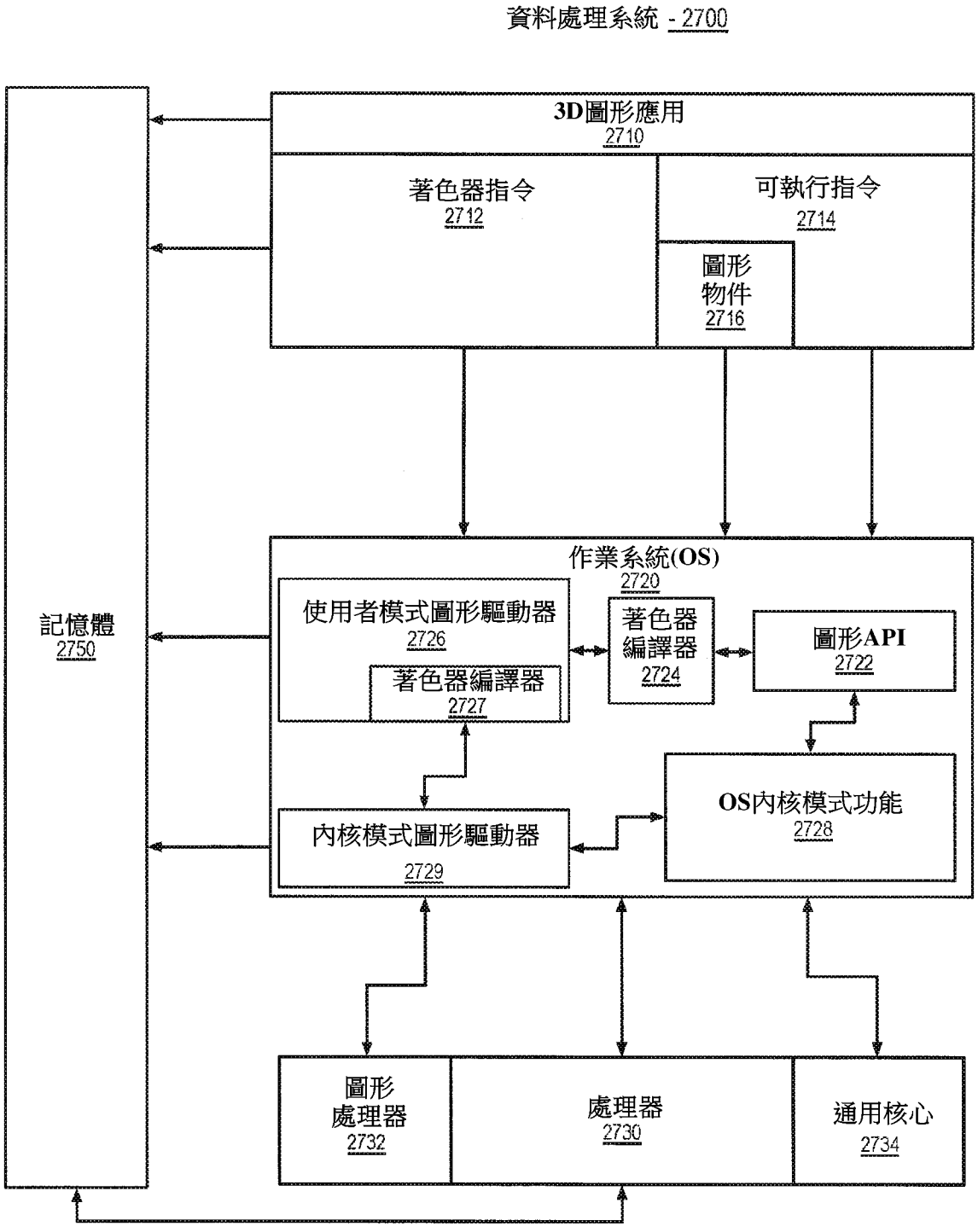
【圖 25】


圖形處理器命令格式
2600
客戶端
2602
運算碼
2604
子運算碼
2605
資料
2606
命令尺寸
2608

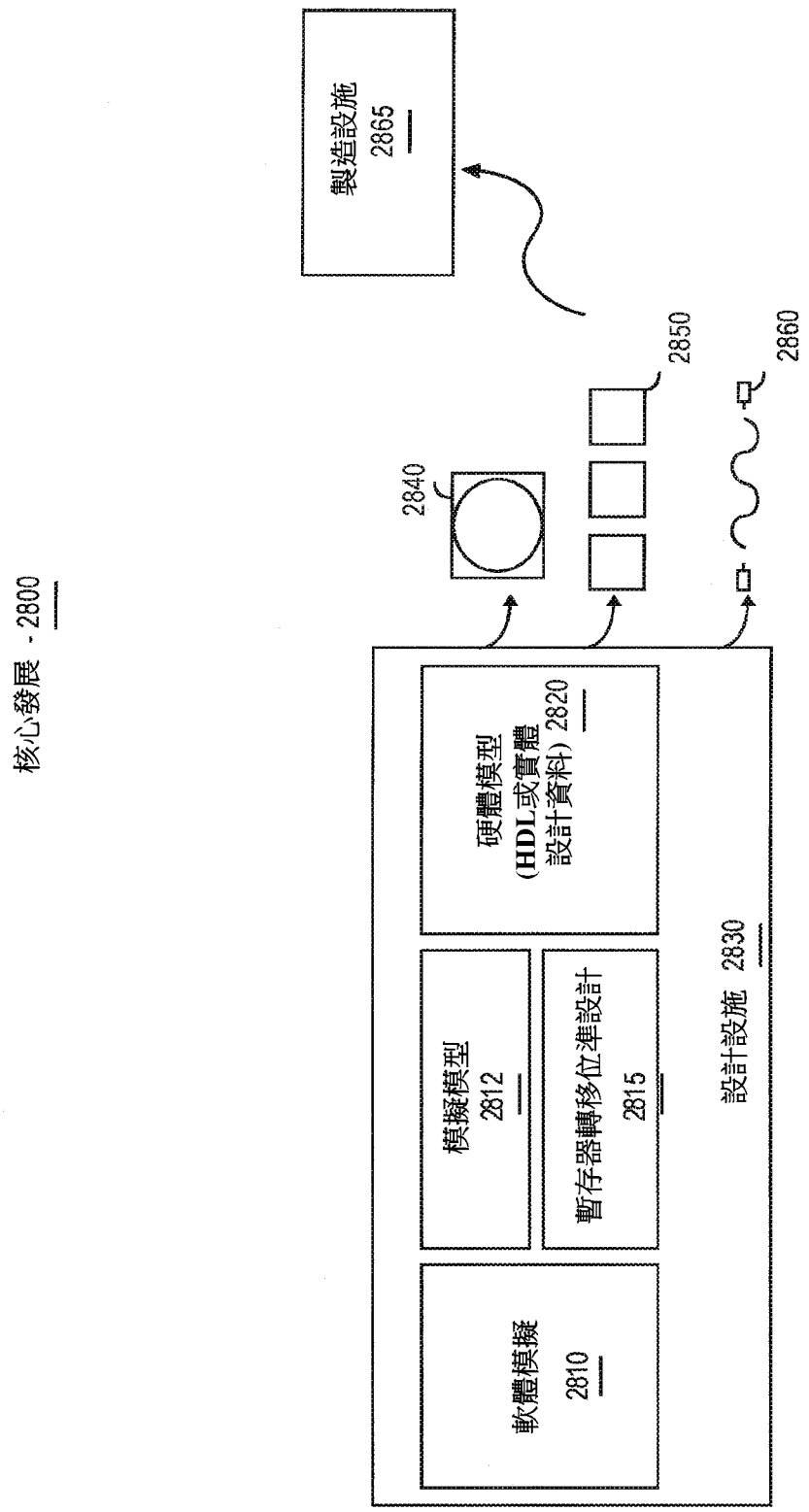
【圖 26A】


圖形處理器命令序列
2610
管線清洗
2612
管線選擇
2613
管線控制
2614
回傳緩衝器狀態
2616
2622
3D
2620
管線?
媒體
2624
3D管線狀態
2630
媒體管線狀態
STATE 2640
3D基原
2632
媒體物件
2642
執行
2634
執行
2644

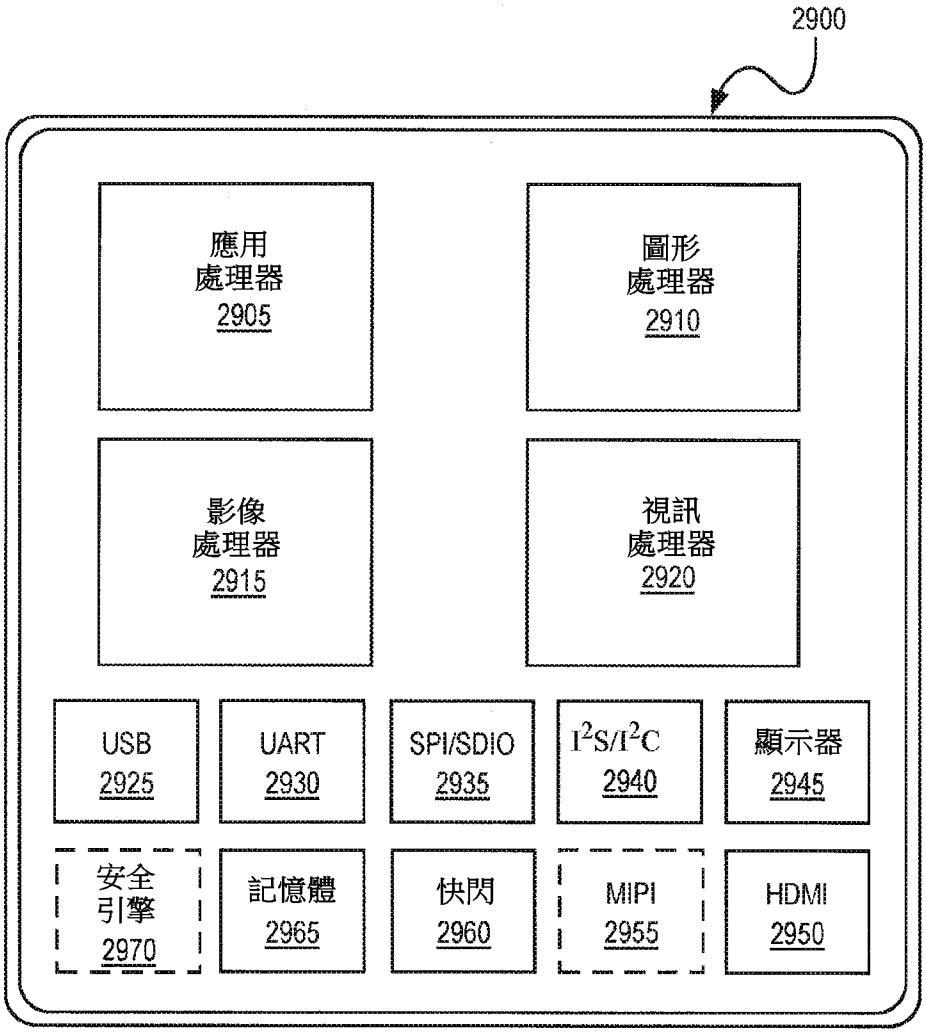
【圖 26B】


資料處理系統 - 2700
3D圖形應用
2710
著色器指令
2712
可執行指令
2714
圖形
物件
2716
作業系統(OS)
2720
記憶體
2750
使用者模式圖形驅動器
2726
著色器編譯器
2727
著色器
編譯器
2724
圖形API
2722
OS內核模式功能
2728
內核模式圖形驅動器
2729
圖形
處理器
2732
處理器
2730
通用核心
2734

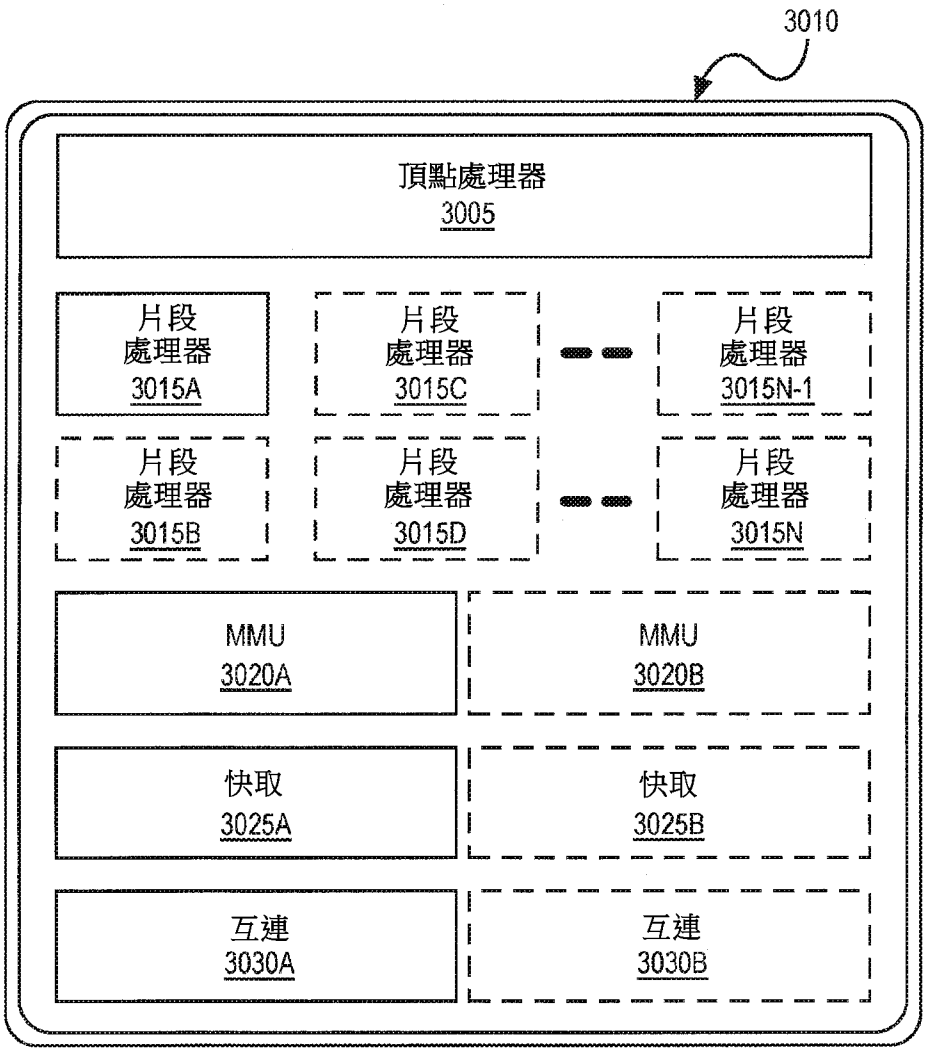
【圖 27】


核心發展 - 2800
製造設施
2865
2840
2850
2860
硬體模型
(HDL或實體
設計資料) 2820
模擬模型
2812
暫存器轉移位準設計
2815
軟體模擬
2810
設計設施 2830

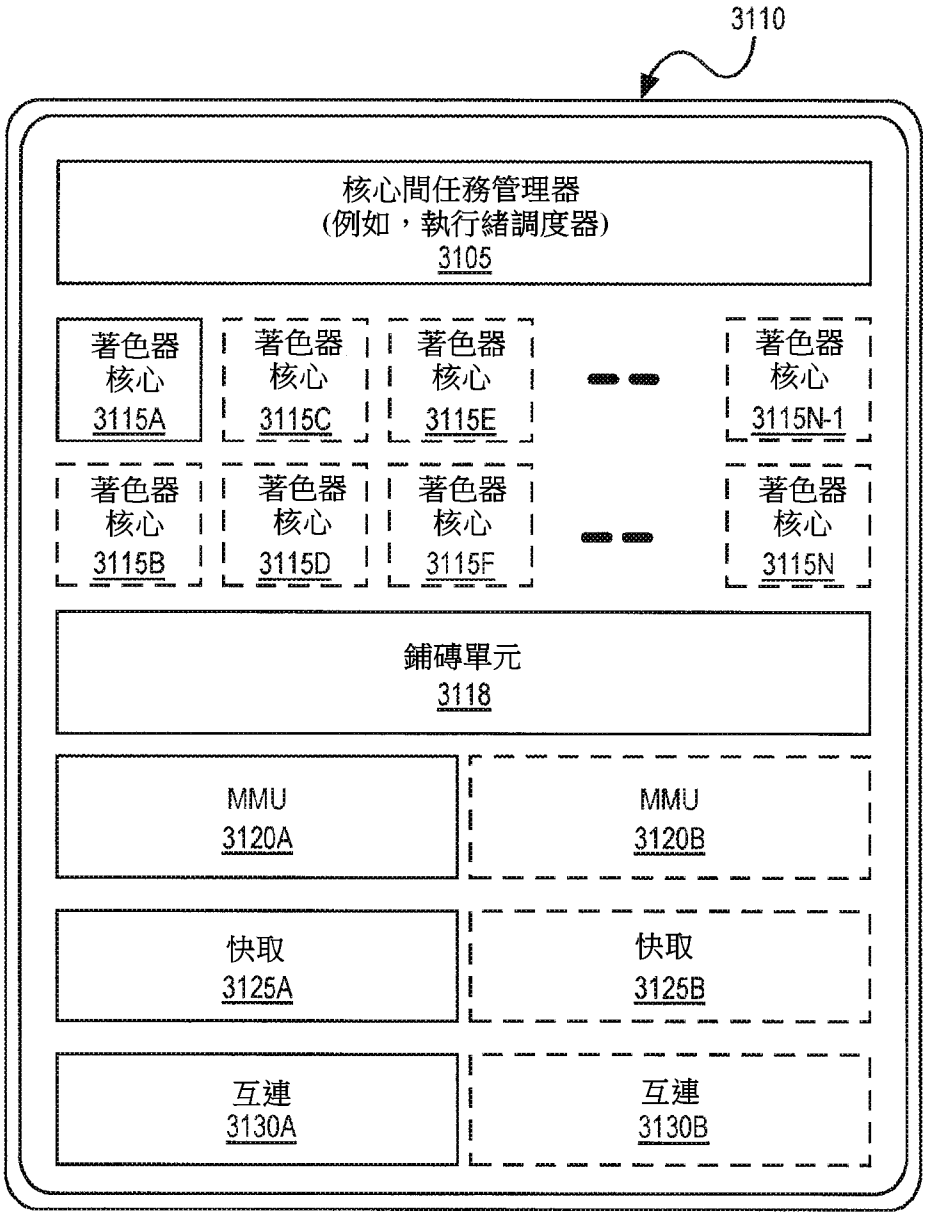
【圖 28】


2900
應用
處理器
2905
圖形
處理器
2910
影像
處理器
2915
視訊
處理器
2920
USB
2925
UART
2930
SPI/SDIO
2935
I2S/I2C
2940
顯示器
2945
安全
引擎
2970
記憶體
2965
快閃
2960
MIPI
2955
HDMI
2950

【圖 29】


3010
頂點處理器
3005
片段
處理器
3015A
片段
處理器
3015C
片段
處理器
3015N-1
片段
處理器
3015B
片段
處理器
3015D
片段
處理器
3015N
MMU
3020A
MMU
3020B
快取
3025A
快取
3025B
互連
3030A
互連
3030B

【圖 30】


3110
核心間任務管理器
(例如，執行緒調度器)
3105
著色器
核心
3115A
著色器
核心
3115C
著色器
核心
3115E
著色器
核心
3115N-1
著色器
核心
3115B
著色器
核心
3115D
著色器
核心
3115F
著色器
核心
3115N
鋪磚單元
3118
MMU
3120A
MMU
3120B
快取
3125A
快取
3125B
互連
3130A
互連
3130B

【圖 31】