



(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 I785182 B

(45)公告日：中華民國 111 (2022) 年 12 月 01 日

(21)申請案號：108100882

(22)申請日：中華民國 107 (2018) 年 02 月 21 日

(51)Int. Cl. : G06N20/00 (2019.01)

G06F16/00 (2019.01)

(30)優先權：2017/04/24 美國

15/494,773

(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：亞西鮑佛 班 ASHBAUGH, BEN (US)；歐德亞麥德維爾 艾蒙斯特阿法 OULD-AHMED-VALL, ELMOUSTAPHA (MR)；唐 屏 TANG, PING (US)；陳曉明 CHEN, XIAOMING (CN)；拉克斯曼 巴拉斯 LAKSHMANAN, BARATH (IN)；史崔克蘭 麥可 STRICKLAND, MICHAEL (US)；許斯曼 塔提安那 SHPEISMAN, TATIANA (US)；雷 喬伊迪普 RAY, JOYDEEP (US)；姚安邦 YAO, ANBANG (CN)；馬立偉 MA, LIWEI (CN)；赫德 琳達 HURD, LINDA (US)

(74)代理人：林志剛

(56)參考文獻：

US 2012/0233444A1

US 2015/0160872A1

網路文獻 NVIDIA Whitepaper-NVIDIA 's Next Generation CUDA Compute Architecture: Fermi V1.1 NVIDIA 2009 https://www.nvidia.com/content/PDF/fermi_white_papers/

NVIDIA_Fermi_Compute_Architecture_Whitepaper.pdf

網路文獻 4、Stephen Junkins The Compute Architecture of Intel Processor Graphics Gen9 Version 1.0 Intel 2015/8/14 <https://www.intel.com/content/dam/develop/external/us/en/documents/the-compute-architecture-of-intel-processor-graphics-gen9-v1d0.pdf>

審查人員：范士隆

申請專利範圍項數：20 項 圖式數：34 共 181 頁

(54)名稱

使用低和高精確度的混合推理

(57)摘要

提供了用以執行機器學習運算的計算設備的一種實施例，該計算設備包含指令解碼邏輯，用以將包含多個運算元的單一指令解碼為單一解碼指令，該多個運算元具有不同的精確度，以及通用圖形計算單元，包含第一邏輯單元和第二邏輯單元，該通用圖形計算單元用以執行該單一解碼指令，其中用以執行該單一解碼指令包含用以對於該多個運算元中的第一組運算元，以第一精確度來執行第一指令運算，以及對於該多個運算元中的第二組運算元，以第二精確度來同時執行第二指令運算。

One embodiment provides for a compute apparatus to perform machine learning operations, the compute apparatus comprising instruction decode logic to decode a single instruction including multiple operands into a single decoded instruction, the multiple operands having differing precisions and a general-purpose graphics compute unit including a first logic unit and a second logic unit, the general-purpose graphics

compute unit to execute the single decoded instruction, wherein to execute the single decoded instruction includes to perform a first instruction operation on a first set of operands of the multiple operands at a first precision and a simultaneously perform second instruction operation on a second set of operands of the multiple operands at a second precision.

指定代表圖：

1700

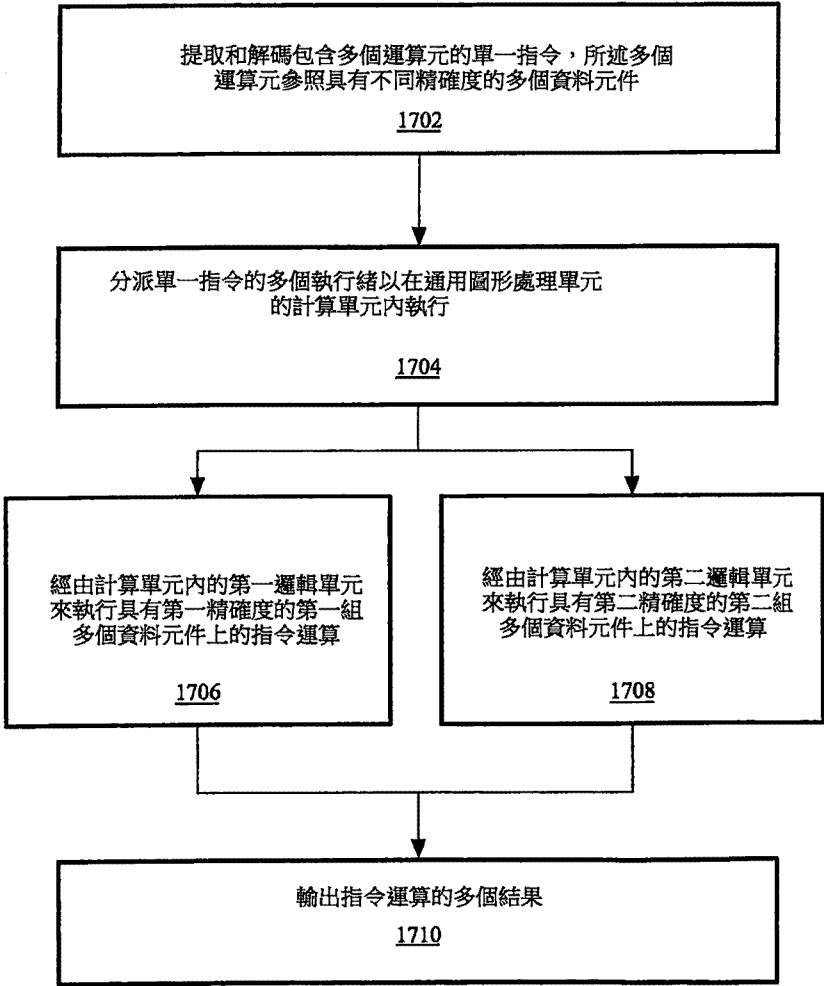


圖 17

【發明摘要】

【中文發明名稱】

使用低和高精確度的混合推理

【英文發明名稱】

MIXED INFERENCE USING LOW AND HIGH PRECISION

【中文】

提供了用以執行機器學習運算的計算設備的一種實施例，該計算設備包含指令解碼邏輯，用以將包含多個運算元的單一指令解碼為單一解碼指令，該多個運算元具有不同的精確度，以及通用圖形計算單元，包含第一邏輯單元和第二邏輯單元，該通用圖形計算單元用以執行該單一解碼指令，其中用以執行該單一解碼指令包含用以對於該多個運算元中的第一組運算元，以第一精確度來執行第一指令運算，以及對於該多個運算元中的第二組運算元，以第二精確度來同時執行第二指令運算。

【 英文 】

One embodiment provides for a compute apparatus to perform machine learning operations, the compute apparatus comprising instruction decode logic to decode a single instruction including multiple operands into a single decoded instruction, the multiple operands having differing precisions and a general-purpose graphics compute unit including a first logic unit and a second logic unit, the general-purpose graphics compute unit to execute the single decoded instruction, wherein to execute the single decoded instruction includes to perform a first instruction operation on a first set of operands of the multiple operands at a first precision and a simultaneously perform second instruction operation on a second set of operands of the multiple operands at a second precision.

【指定代表圖】第(17)圖。
【代表圖之符號簡單說明】無

【特徵化學式】無

【發明說明書】

【中文發明名稱】

使用低和高精確度的混合推理

【英文發明名稱】

MIXED INFERENCE USING LOW AND HIGH PRECISION

【技術領域】

【0001】實施例通常有關於資料處理，並且更具體地有關於經由通用圖形處理單元之資料處理。

【先前技術】

【0002】當前的平行圖形資料處理包括被開發以對圖形資料執行諸如(例如)線性內插、鑲嵌、柵格化、紋理映射、深度測試等特定操作之系統及方法。傳統上，圖形處理器使用固定功能計算單元來處理圖形資料；然而，更近期地，部分的圖形處理器已被製成可編程的，其致使這樣的處理器支援更多樣的操作以用於處理頂點及片段資料。

【0003】為了進一步增加效能，圖形處理器通常實現處理技術，諸如管線操作，其嘗試遍及圖形管線的不同部分平行地處理盡可能多的圖形資料。具有單指令、多執行緒(SIMT)架構之平行圖形處理器被設計以將圖形管線中之平行處理的量最大化。在SIMT架構中，數組平行執行緒嘗試盡可能頻繁地一起同步地執行程式指令以增加處理效

率。針對SIMT架構之軟體及硬體的一般概述可見於Shane Cook, *CUDA Programming*, 第3章, 第37-51頁(2013)和/或Nicholas Wilt, *CUDA Handbook, A Comprehensive Guide to GPU Programming*, 第2.6.2至3.1.2節(2013年六月)。

【圖式簡單說明】

【0004】為了本發明的特徵能夠被詳細地瞭解，本發明的更具體描述可藉由參考實施例而獲得，其部分係顯示在所附圖式中。然而，應注意，所附圖式僅顯示典型的實施例，並且因此不應被視為所有實施例的範圍的限制。

【0005】圖1為顯示一種組態成實現本文中所述之實施例的一或多種態樣之電腦系統的方塊圖；

【0006】圖2A-2D顯示根據實施例的平行處理器組件；

【0007】圖3A-3B為根據實施例之圖形多處理器的方塊圖；

【0008】圖4A-4F顯示複數個GPU被通訊地耦接至複數個多核心處理器的範例架構；

【0009】圖5顯示根據實施例的圖形處理器管線；

【0010】圖6顯示根據實施例的機器學習軟體堆疊；

【0011】圖7顯示根據實施例的高度平行的通用圖形處理單元；

【0012】圖8顯示根據實施例的多GPU計算系統；

【0013】圖9A-9B顯示範例深神經網路之層；

【0014】圖10顯示範例遞歸神經網路；

【0015】圖11顯示深神經網路之訓練及部署；

【0016】圖12為顯示分散式學習的方塊圖；

【0017】圖13顯示適於使用經訓練的模型來執行推理的範例推理系統單晶片(SOC)；

【0018】圖14是根據一個實施例的多處理器單元的方塊圖；

【0019】圖15顯示了根據一個實施例的混合精確度處理系統；

【0020】圖16顯示了根據一個實施例的額外的混合精確度處理系統；

【0021】圖17是根據一個實施例的用於混合精確度處理系統的運算邏輯的流程圖；

【0022】圖18是根據一個實施例的用在另一種混合精確度處理系統的運算邏輯的流程圖；

【0023】圖19顯示了根據一個實施例的機器學習系統；

【0024】圖20顯示了根據一個實施例的機器學習系統的邏輯運算；

【0025】圖21為根據實施例之處理系統的方塊圖；

【0026】圖22為根據實施例之處理器的方塊圖；

【0027】圖23為根據實施例之圖形處理器的方塊圖；

【0028】圖24為根據一些實施例之圖形處理器的圖形處理引擎的方塊圖；

【0029】圖25為由額外實施例所提供之圖形處理器的方塊圖；

【0030】圖26顯示包括一些實施例中所利用的處理元件之陣列的執行緒執行邏輯；

【0031】圖27為顯示根據一些實施例之圖形處理器指令格式的方塊圖；

【0032】圖28為根據另一實施例之圖形處理器的方塊圖；

【0033】圖29A-29B顯示根據一些實施例之圖形處理器命令格式及命令序列；

【0034】圖30顯示根據一些實施例之用於資料處理系統的範例圖形軟體架構；

【0035】圖31為顯示根據實施例之IP核心開發系統的方塊圖；

【0036】圖32為顯示根據實施例之範例系統單晶片積體電路的方塊圖；

【0037】圖33為顯示根據實施例之額外圖形處理器的方塊圖；以及

【0038】圖34為顯示根據實施例之系統單晶片積體電路的額外範例圖形處理器的方塊圖。

【發明內容】及【實施方式】

【0039】在一些實施例中，圖形處理單元(GPU)通訊地耦接到主機/處理器核心，以加速圖形運算、機器學習

運算、模式分析運算和各種通用 GPU(GPGPU)功能。GPU 可以透過匯流排或另一互連(例如，諸如 PCIe 或 NVLink 的高速互連)通訊地耦接到主機處理器/核心。在其它實施例中，GPU 可以與核心一樣整合在相同的封裝或晶片上，並透過內部處理器匯流排/互連(即，在封裝或晶片內部)通訊地耦接到核心。不管 GPU 連接的方式為何，處理器核心都可以用包含在工作描述符中的命令/指令序列的形式將工作分配給 GPU。GPU 接著使用專用電路/邏輯來有效地處理這些命令/指令。

【0040】在以下說明中，闡述了許多特定細節以提供更透徹的瞭解。然而，本領域技術人員將清楚本文中所述之實施例可在沒有這些特定細節中的一或多者下被實行。在其它實例中，眾所周知的特徵未被描述以避免混淆本實施例的細節。

系統概述

【0041】圖 1 為顯示組態成實現本文中所述之實施例的一或多種態樣之計算系統 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 及系統記憶體 104，其經由可包括記憶體集線器 105 之互連路徑來通訊。記憶體集線器 105 可以是晶片組組件內的獨立組件或可被整合於一或多個處理器 102 內。記憶體集線器 105 係經由通訊鏈路 106 而與 I/O 子系統 111 耦接。I/O 子系統 111 包括 I/O 集線器 107，其可致使計算系統 100 用以接收來自一

或多個輸入裝置108之輸入。此外，I/O集線器107可致使顯示控制器(其可被包括在一或多個處理器102中)用以提供輸出至一或多個顯示裝置110A。在一個實施例中，與I/O集線器107耦接中的一或多個顯示裝置110A可包括本地的、內部的或嵌入式顯示裝置。

【0042】在一個實施例中，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鏈路113而耦接至記憶體集線器105。通訊鏈路113可以是任何數目之標準為基的通訊鏈路技術或協定之一，諸如(但不限於)PCI Express；或可以是供應商特定的通訊介面或通訊結構。在一個實施例中，所述一或多個平行處理器112形成一種計算集中的平行或向量處理系統，其可包括大量的處理核心和/或處理叢集，諸如許多積體核心(MIC)處理器。在一個實施例中，所述一或多個平行處理器112形成一種圖形處理子系統，其可將像素輸出至經由I/O集線器107耦接的一或多個顯示裝置110A之一。所述一或多個平行處理器112還可包括顯示控制器及顯示介面(未顯示)，用以致使對於一或多個顯示裝置110B的直接連接。

【0043】在I/O子系統111內，系統儲存單元114可連接至I/O集線器107，以提供用於計算系統100之儲存機制。I/O開關116可被用來提供介面機制，以致使I/O集線器107與其它組件之間的連接，諸如可被整合在平台中的網路配接器118和/或無線網路配接器119，以及可經由一或多個外接裝置120而被加入之各種其它裝置。網路配接

器 118 可以是乙太網路配接器或其它的有線網路配接器。無線網路配接器 119 可包括 Wi-Fi、藍芽、近場通訊 (NFC) 或包括一或多個無線電的其它網路裝置中的一或多者。

【0044】計算系統 100 可包括其它未明確地顯示之組件 (包括 USB 或其它埠連接、光學儲存驅動、視頻擷取裝置等) 還可被連接至 I/O 集線器 107。將圖 1 中之各個組件互連的通訊路徑可使用任何適當的協定來實現，諸如 PCI (周邊組件互連) 為基的協定 (例如，PCI-Express) 或者任何其它匯流排或點對點通訊介面和 / 或協定，諸如 NV 鏈路高速互連或本領域中已知的互連協定。

【0045】在一個實施例中，所述一或多個平行處理器 112 結合了針對圖形及視頻處理而被最佳化的電路，包括 (例如) 視頻輸出電路，並組成圖形處理單元 (GPU)。在另一實施例中，所述一或多個平行處理器 112 結合了針對通用處理而被最佳化的電路，同時保留基本計算架構，更詳細地描述於本文中。在又另一實施例中，計算系統 100 之組件可與一或多個其它系統元件被整合在單一積體電路上。例如，一或多個平行處理器 112、記憶體集線器 105、處理器 102 及 I/O 集線器 107 可被整合在系統單晶片 (SoC) 積體電路之中。可替代地，計算系統 100 之組件可被整合在單一封裝之中，以形成系統級封裝 (SIP) 組態。在一個實施例中，計算系統 100 的組件之至少一部分可被整合在多晶片模組 (MCM) 之中，其可與其它多晶片模組被互連入模組式計算系統中。

【0046】將理解，本文中所示之計算系統100為說明性的且其變化及修改是可能的。連接拓撲(包括橋的數目和配置、處理器102的數目及平行處理器112的數目)可如所需地修改。例如，在一些實施例中，系統記憶體104被直接地連接至處理器102而非透過橋，而其它裝置係經由記憶體集線器105及處理器102與系統記憶體104通訊。在其它替代拓撲中，平行處理器112被連接至I/O集線器107或直接至一或多個處理器102之一，而非至記憶體集線器105。在其它實施例中，I/O集線器107及記憶體集線器105可被整合在單一晶片之中。一些實施例可包括經由多個插口而相連之二組或更多組的處理器102，其可與平行處理器112中的二個或更多實例耦接。

【0047】本文中所示之一些特定組件為選擇性的且可能不被包括在計算系統100的所有實現中。例如，任何數目的外接卡或周邊設備可被支援，或者一些組件可被刪除。再者，一些架構可對於類似圖1中所示之那些組件使用不同術語。例如，記憶體集線器105可在一些架構中被稱為北橋，而I/O集線器107可被稱為南橋。

【0048】圖2A顯示根據實施例的平行處理器200。平行處理器200的各個組件可使用一或多個積體電路裝置來實現，諸如可編程處理器、特殊應用積體電路(ASIC)或場可編程閘陣列(FPGA)。所示的平行處理器200為圖1中所示根據實施例的一或多個平行處理器112之變體。

【0049】在一個實施例中，平行處理器200包括平行

處理單元202。平行處理單元包括致使與其它裝置通訊之I/O單元204，其包括平行處理單元202的其它實例。I/O單元204可被直接地連接至其它裝置。在一個實施例中，I/O單元204係經由集線器或開關介面(諸如記憶體集線器105)之使用而與其它裝置連接。介於記憶體集線器105與I/O單元204之間的連接形成了通訊鏈路113。在平行處理單元202之內，I/O單元204係與主機介面206及記憶體交叉開關216連接，其中主機介面206接收了針對執行處理操作之命令，而記憶體交叉開關216接收了針對執行記憶體操作之命令。

【0050】當主機介面206經由I/O單元204接收了命令緩衝器時，主機介面206可將用以執行那些命令之工作操作指引至前端208。在一個實施例中，前端208係與排程器210耦接，排程器210係組態成將命令或其它工作項目分配至處理叢集陣列212。在一個實施例中，在工作被分配至處理叢集陣列212之處理叢集之前，排程器210確保處理叢集陣列212被適當地組態且處於有效狀態。在一個實施例中，排程器210經由在微控制器上執行的韌體邏輯來實現。微控制器實現的排程器210可配置以在粗略和精細粒度執行複雜的排程和工作分配的操作，致使快速搶占和執行緒的背景切換執行在處理陣列212上。在一個實施例中，主機軟體可以證明工作負載經由多個圖形處理門鈴中的一個來排程在處理陣列212上。工作負載接著可以藉由排程器微控制器內的排程器210邏輯來自動地跨處理陣列

212分佈。

【0051】處理叢集陣列212可包括高達「N」個處理叢集(例如，叢集214A、叢集214B至叢集214N)。處理叢集陣列212中的各個叢集214A-214N可執行大量的並行執行緒。排程器210可使用各種排程和/或工作分配演算法以將工作配置給處理叢集陣列212中的叢集214A-214N，其可根據各類型的程式或計算所產生的工作負載而改變。排程可由排程器210來動態地處置，或在針對藉由處理叢集陣列212之執行而組態的程式邏輯之編譯期間可部分地由編譯器邏輯來協助。在一個實施例中，處理叢集陣列212中的不同叢集214A-214N可被配置以供處理不同類型的程式或以供執行不同類型的計算。

【0052】處理叢集陣列212可組態成執行各種類型的平行處理操作。在一個實施例中，處理叢集陣列212係組態成執行通用平行計算操作。例如，處理叢集陣列212可包括用以執行處理工作之邏輯，該些處理工作包括視頻和/或音頻資料之過濾、執行包括物理操作的建模操作及執行資料轉換。

【0053】在一個實施例中，處理叢集陣列212係組態成執行平行圖形處理操作。在平行處理器200被組態成執行圖形處理操作之實施例中，處理叢集陣列212可包括用以支援此類圖形處理操作之執行的額外邏輯，包括但不限於用以執行紋理操作的紋理取樣邏輯、以及鑲嵌邏輯和其它頂點處理邏輯。此外，處理叢集陣列212可被組態成執

行圖形處理相關的著色器程式，諸如(但不限於)頂點著色器、鑲嵌著色器、幾何著色器及像素著色器。平行處理單元202可經由I/O單元204將來自系統記憶體之資料轉移以供處理。在處理期間，已轉移資料可於處理期間被儲存至晶載記憶體(例如，平行處理器記憶體222)，接著被寫回至系統記憶體。

【0054】在一個實施例中，當平行處理單元202被用於執行圖形處理時，排程器210可被組態成將處理工作負載劃分為約略相等大小的工作，以較佳地致使圖形處理操作之分配至處理叢集陣列212中的多個叢集214A-214N。在一些實施例中，部分的處理叢集陣列212可被組態成執行不同類型的處理。例如，第一部分可被組態成執行頂點著色及拓撲產生，第二部分可被組態成執行鑲嵌及幾何著色，而第三部分可被組態成執行像素著色或其它螢幕空白操作，以產生渲染影像以供顯示。由叢集214A-214N中的一或多者所產生的中間資料可被儲存於緩衝器中，以容許該中間資料被傳輸在叢集214A-214N之間以供進一步處理。

【0055】在操作期間，處理叢集陣列212可經由排程器210來接收待執行的處理工作，其接收來自前端208之定義處理工作的命令。針對圖形處理操作，處理工作可包括待處理資料之指標，例如，表面(修補)資料、基元資料、頂點資料和/或像素資料、以及定義該資料應如何被處理(例如，哪個程式應被執行)之狀態參數和命令。排程器

210可被組態成提取相應於該些工作之指標或者可接收來自前端208之指標。前端208可被組態成在由到來的命令緩衝器(例如，批次緩衝器、推送緩衝器，等等)所指明之工作負載被起始之前，確保處理叢集陣列212被組態為有效狀態。

【0056】平行處理單元202中的一或多個實例之各者可與平行處理器記憶體222耦接。平行處理器記憶體222可經由記憶體交叉開關216而被存取，其可接收來自處理叢集陣列212以及I/O單元204之記憶體請求。記憶體交叉開關216可經由記憶體介面218來存取平行處理器記憶體222。記憶體介面218可包括多個分割單元(例如，分割單元220A、分割單元220B至分割單元220N)，其可各自耦接至平行處理器記憶體222的一部分(例如，記憶體單元)。在一種實現中，分割單元220A-220N的數目被組態成等於記憶體單元的數目，以致使第一分割單元220A具有相應的第一記憶體單元224A、第二分割單元220B具有相應的記憶體單元224B及第N分割單元220N具有相應的第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。

【0057】在各個實施例中，記憶體單元224A-224N可包括各種類型的記憶體裝置，包括動態隨機存取記憶體(DRAM)或圖形隨機存取記憶體，諸如同步圖形隨機存取記憶體(SGRAM)，包括圖形雙資料速率(GDDR)記憶體。在一個實施例中，記憶體單元224A-224N還可包括3D堆疊

記憶體，包括但不限於高頻寬記憶體(HBM)。本領域技術人員將理解，記憶體單元224A-224N之特定實現可變化，並可被選自各種傳統設計之一。渲染目標(諸如框緩衝器或紋理映射)可跨越記憶體單元224A-224N被儲存，其容許分割單元220A-220N平行地寫入各渲染目標之部分，以有效率地使用平行處理器記憶體222之可用頻寬。在一些實施例中，平行處理器記憶體222的本地實例可被排除以利於統一的記憶體設計，其利用結合本地快取記憶體的系統記憶體。

【0058】在一個實施例中，處理叢集陣列212的叢集214A-214N中的任一者可處理將被寫入至平行處理器記憶體222內的記憶體單元224A-224N中的任一者的資料。記憶體交叉開關216可被組態成將各叢集214A-214N之輸出轉移至任何分割單元220A-220N或者至可對該輸出執行額外的處理操作之另一叢集214A-214N。各叢集214A-214N可透過記憶體交叉開關216而與記憶體介面218通訊，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。在一個實施例中，記憶體交叉開關216具有至記憶體介面218用以與I/O單元204通訊之連接，以及至平行處理器記憶體222之本地實例的連接，其致使不同處理叢集214A-214N內的處理單元與系統記憶體或其它不在平行處理單元202本地之記憶體通訊。在一個實施例中，記憶體交叉開關216可使用虛擬通道來分離介於叢集214A-214N與分割單元220A-220N之間的流量串。

【0059】雖然平行處理單元202的單一實例被顯示在平行處理器200內，但平行處理單元202之任何數目的實例均可被包括。例如，平行處理單元202的多個實例可被設置在單一外接卡上，或者多個外接卡可被互連。平行處理單元202的不同實例可被組態成相互操作，即使不同的實例具有不同數目的處理核心、不同數量的本地平行處理器記憶體和/或其它組態差異。例如以及在一個實施例中，平行處理單元202的一些實例可相對在其它實例包括較高精確度的浮點單元。結合平行處理單元202或平行處理器200中的一或多個實例的系統可用多種組態及形狀因數來實現，包括但不限於桌上型電腦、筆記型電腦或手持式個人電腦、伺服器、工作站、遊戲主控台和/或嵌入式系統。

【0060】圖2B為根據實施例的分割單元220的方塊圖。在一個實施例中，分割單元220為圖2A的分割單元220A-220N中之一者的實例。如圖所示，分割單元220包括L2快取221、框緩衝器介面225及ROP 226(柵格操作單元)。L2快取221為讀取/寫入快取，其被組態成執行從記憶體交叉開關216及ROP 226所接收之載入及儲存操作。讀取失誤與緊急寫回請求係由L2快取221輸出至框緩衝器介面225以供處理。更新還可經由框緩衝器介面225來傳送至框緩衝器以供處理。在一個實施例中，框緩衝器介面225係與平行處理器記憶體中的記憶體單元中之一者介接，諸如圖2的記憶體單元224A-224N(例如，在平行處理器記憶體222內)。

【0061】在圖形應用中，ROP 226為執行諸如模板、z測試、混合等等柵格操作的處理單元。ROP 226接著輸出被儲存在圖形記憶體中之經處理的圖形資料。在一些實施例中，ROP 226包括用以壓縮被寫入至記憶體的深度或顏色資料並解壓縮從記憶體讀取的深度或顏色資料之壓縮邏輯。壓縮邏輯可以是使用多種壓縮演算法中的一或多種的無損壓縮邏輯。由ROP 226執行的壓縮類型可基於待壓縮資料的統計特性來變化。例如，在一個實施例中，增量顏色壓縮係在每個磚片的基礎上，對於深度和彩色資料執行。

【0062】在一些實施例中，ROP 226被包括在各處理叢集(例如，圖2的叢集214A-214N)內而非在分割單元220內。在此類實施例中，針對像素資料之讀取及寫入請求透過記憶體交叉開關216，而非像素片段資料被傳輸。經處理的圖形資料可被顯示在顯示裝置(諸如圖1的一或多個顯示裝置110之一)上、被發送以供藉由處理器102之進一步處理或被發送以供圖2A之平行處理器200內的處理實體之一的進一步處理。

【0063】圖2C為根據實施例的平行處理單元內的處理叢集214的方塊圖。在一個實施例中，處理叢集為圖2A的處理叢集214A-214N之一的實例。處理叢集214可被組態成平行地執行許多執行緒，其中術語「執行緒」係指稱執行於特定組輸入資料上之特定程式的實例。在一些實施例中，單一指令、多重資料(SIMD)指令發送技術被用來支援

大量執行緒的平行執行而不需要提供多個獨立的指令單元。在其它實施例中，單一指令、多重執行緒(SIMT)技術被用來支援大量一般同步化執行緒的平行執行，其使用組態成將指令發送至處理叢集之各者內的一組處理引擎的常見指令單元。不同於SIMD執行體系(其中所有處理引擎通常執行相同的指令)，SIMT執行容許不同的執行緒透過既定的執行緒程式而更輕易地跟隨分散的執行路徑。本領域技術人員將瞭解SIMD處理體系代表SIMT處理體系之功能性子集。

【0064】處理叢集214之操作可經由管線管理器232而被控制，管線管理器232將處理工作分配至SIMT平行處理器。管線管理器232接收來自圖2的排程器210的指令，並經由圖形多處理器234和/或紋理單元236來管理那些指令之執行。所示之圖形多處理器234為SIMT平行處理器的範例實例。然而，不同架構的各種類型SIMT平行處理器可被包括在處理叢集214內。圖形多處理器234中的一或多個實例可被包括在處理叢集214內。圖形多處理器234可處理資料，而資料交叉開關240可被用來將經處理資料分配至多個可能的目的地之一，包括其它著色器單元。管線管理器232可藉由指明將經由資料交叉開關240而被分配之經處理資料的目的地以協助經處理資料之分配。

【0065】處理叢集214內的各圖形多處理器234可包括相同組的功能性執行邏輯(例如，算術邏輯單元、載入-儲存單元，等等)。功能性執行邏輯可用管線方式來組態，

其中新的指令可在先前指令完成之前被發送。功能性執行邏輯支援多種操作，包括整數和浮點算術、比較操作、布林操作、位元移位及各種代數函數之計算。在一個實施例中，相同的功能性單元硬體可被用來執行不同的操作，且可能存在功能性單元的任何組合。

【0066】被傳輸至處理叢集214之指令構成執行緒。跨越該組平行處理引擎來執行的一組執行緒為執行緒群組。執行緒群組係對於不同的輸入資料執行相同的程式。執行緒群組內的各執行緒可被指派給圖形多處理器234內的不同處理引擎。執行緒群組可包括比圖形多處理器234內的處理引擎數量更少的執行緒。當執行緒群組包括比處理引擎數量更少的執行緒時，在執行緒群組正被處理之週期期間，處理引擎中的一或多者可以是閒置的。執行緒群組還可包括比圖形多處理器234內的處理引擎數量更多的執行緒。當執行緒群組包括比圖形多處理器234內的處理引擎數量更多的執行緒時，處理可在連續的時脈週期被執行。在一個實施例中，多個執行緒群組可被同時地執行於圖形多處理器234。

【0067】在一個實施例中，圖形多處理器234包括用以執行載入及儲存操作的內部快取記憶體。在一個實施例中，圖形多處理器234可放棄內部快取並使用處理叢集214內的快取記憶體(例如，L1快取308)。各圖形多處理器234也有權存取分割單元(例如，圖2的分割單元220A-220N)內的L2快取，該些L2快取可被共用於所有處理叢集214之間

且可被用來轉移執行緒之間的資料。圖形多處理器234還可存取晶片外全域記憶體，其可包括本地平行處理器記憶體和/或系統記憶體中的一或多者。平行處理單元202外部的任何記憶體可被使用為全域記憶體。處理叢集214包括圖形多處理器234的多個實例的實施例可共用可被儲存於L1快取308中的共同指令及資料。

【0068】各處理叢集214可包括MMU 245(記憶體管理單元)，其被組態成將虛擬位址映射成實體位址。在其它實施例中，MMU 245中的一或多個實例可駐存於圖2的記憶體介面218內。MMU 245包括被用來將虛擬位址映射成磚片之實體位址的一組頁面表項目(PTE)，及(選擇性地)快取線指標。MMU 245可包括位址轉換後備緩衝(TLB)或可駐存於圖形多處理器234內的快取或L1快取或處理叢集214。實體位址被處理以分配表面資料存取局部性，以容許有效率的請求分割單元之間的交錯。快取線索引可被用來判定針對快取線之請求為命中或未命中。

【0069】在圖形及計算應用中，處理叢集214可被組態以致使各圖形多處理器234被耦接至紋理單元236以供執行紋理映射操作，例如，判定紋理樣本位置、讀取紋理資料及過濾紋理資料。紋理資料係從內部紋理L1快取(未顯示)或者(在一些實施例)從圖形多處理器234內的L1快取讀取，且依照所需從L2快取、本地平行處理器記憶體或系統記憶體提取。各圖形多處理器234將經處理的工作輸出至資料交叉開關240以將該經處理的工作提供至另一處理叢

集214以供進一步處理，或者經由記憶體交叉開關216將該經處理的工作儲存於L2快取、本地平行處理器記憶體或系統記憶體中。preROP 242(預柵格操作單元)被組態成從圖形多處理器234接收資料、導引資料至ROP單元，其可連同如本文中所述之分割單元(例如，圖2的分割單元220A-220N)被配置。preROP 242單元可執行針對顏色混合之最佳化、組織像素顏色資料及執行位址轉換。

【0070】將理解，本文中所述之核心架構為說明性的且其變化及修改是可能的。任何數目的處理單元(例如，圖形多處理器234、紋理單元236、preROP 242，等等)可被包括在處理叢集214內。再者，雖然僅顯示一個處理叢集214，但如本文中所述之平行處理單元可包括任何數量的實例的處理叢集214。在一個實施例中，各處理叢集214可被組態成使用獨立且不同的處理單元、L1快取等，獨立於其它處理叢集214操作。

【0071】圖2D顯示根據實施例的圖形多處理器234。在此類實施例中，圖形多處理器234係與處理叢集214之管線管理器232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及一或多個載入/儲存單元266。GPGPU核心262及載入/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。

【0072】在一個實施例中，指令快取252接收用以從

管線管理器232執行的指令串。該些指令被快取在指令快取252中且被調度以供指令單元254執行。指令單元254可調度指令為執行緒群組(例如，經線)，其中執行緒群組中之各執行緒被指派給GPGPU核心262內的不同執行單元。指令可藉由指明統一位址空間內的位址來存取本地、共用或全域位址空間中的任一者。位址映射單元256可被用來將統一位址空間中之位址轉換為可由載入/儲存單元266存取的相異記憶體位址。

【0073】暫存器檔案258提供用於圖形多處理器324之功能性單元的一組暫存器。暫存器檔案258提供暫時儲存給連接至圖形多處理器324之功能性單元(例如，GPGPU核心262、載入/儲存單元266)的資料路徑的運算元。在一個實施例中，暫存器檔案258被劃分在功能性單元的各者之間以致使各功能性單元被配置暫存器檔案258之專用部分。在一個實施例中，暫存器檔案258被劃分在正由圖形多處理器324所執行的不同經線之間。

【0074】GPGPU核心262可各包括浮點單元(FPU)和/或整數算術邏輯單元(ALU)，其被用來執行圖形多處理器324之指令。根據實施例，GPGPU核心262可於架構上類似或可於架構上不同。例如以及在一個實施例中，GPGPU核心262的第一部分包括單一精確度FPU及整數ALU，而GPGPU核心的第二部分包括雙精確度FPU。在一個實施例中，FPU可實現IEEE 754-2008標準於浮點算術或者致使可變精確度浮點算術。圖形多處理器324可額外地包括一或

多個固定功能或特殊功能單元，用以執行諸如複製矩形或像素混合操作的特定功能。在一個實施例中，GPGPU核心中的一或多者還可包括固定或特殊功能邏輯。

【0075】在一個實施例中，GPGPU核心262包含能夠對於多組資料執行單一指令的SIMD邏輯。在一個實施例中，GPGPU核心262可以實體地執行SIMD4、SIMD8和SIMD16指令，並且邏輯地執行SIMD1、SIMD2和SIMD32指令。用於GPGPU核心SIMD指令可在編譯時由著色器編譯器產生或在執行程式編寫時自動產生，並且針對單一程式多重資料(SPMD)或SIMT架構來編譯。針對SIMT執行模型配置的程式的多個執行緒可以經由單一SIMD指令來執行。例如並且在一個實施例中，執行相同或類似運算的八個SIMT執行緒可以經由單一SIMD8邏輯單元並行地執行。

【0076】記憶體及快取互連268為互連網路，其將圖形多處理器324的功能性單元之各者連接至暫存器檔案258以及連接至共用記憶體270。在一個實施例中，記憶體及快取互連268為交叉開關互連，其容許載入/儲存單元266實現介於共用記憶體270與暫存器檔案258之間的載入和儲存操作。暫存器檔案258可操作於如GPGPU核心262之相同頻率，因此介於GPGPU核心262與暫存器檔案258之間的資料轉移是非常低延遲時間的。共用記憶體270可被用來致使介於執行在圖形多處理器234內的功能性單元上的執行緒之間的通訊。快取記憶體272可被使用為資料快取，例如，用以快取在功能性單元與紋理單元236之間傳遞的紋

理資料。共用記憶體270還可被使用為程式管理快取的。除了被儲存於快取記憶體272內的自動快取資料以外，執行於GPGPU核心262上之執行緒可將資料藉由程式化儲存於共用記憶體內。

【0077】圖3A-3B顯示根據實施例的額外圖形多處理器。所示的圖形多處理器325、350為圖2C之圖形多處理器234的變體。所示的圖形多處理器325、350可被組態為能夠同時執行大量執行緒之串流多處理器(SM)。

【0078】圖3A顯示根據額外實施例的圖形多處理器325。圖形多處理器325包括相對於圖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可包括指令單元332A-332B、暫存器檔案334A-334B及紋理單元344A-344B的多個實例。圖形多處理器325還包括多組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)及多組載入/儲存單元340A-340B。在一個實施例中，執行資源單元具有共同指令快取330、紋理和/或資料快取記憶體342及共用記憶體346。

【0079】各個組件可經由互連構造327來通訊。在一個實施例中，互連構造327包括一或多個交叉開關，用以致使介於圖形多處理器325的各個組件之間的通訊。在一個實施例中，互連構造327是圖形多處理器325的每個組件堆疊在其上的獨立、高速網路構造層。圖形多處理器325的組件經由互連構造327與遠端組件通訊。例如，GPGPU

核心 336A-336B、337A-337B和 3378A-338B可各自經由互連構造 327與共用記憶體 346進行通訊。互連構造 327可仲裁圖形多處理器 325內的通訊，以確保組件之間的公平頻寬分配。

【0080】圖 3B顯示根據額外實施例的圖形多處理器 350。如圖 2D及圖 3A中所示，圖形處理器包括多組執行資源 356A-356D，其中各組執行資源包括多個指令單元、暫存器檔案、GPGPU核心及載入儲存單元。執行資源 356A-356D可配合紋理單元 360A-360D工作以供紋理操作，同時共用指令快取 354及共用記憶體 362。在一個實施例中，執行資源 356A-356D可共用指令快取 354和共用記憶體 362，以及紋理和/或資料快取記憶體 358A-358B的多個實例。各個組件可經由類似於圖 3A之互連構造 327的互連構造 352來通訊。

【0081】本領域技術人員將理解，圖 1、2A-2D及 3A-3B中所述之架構為說明性的而不是限制本實施例的範圍。因此，本文中所述之技術可被實現於任何適當組態的處理單元上，包括(非限制)一或多個行動應用處理器、包括多核心 CPU的一或多個桌上型電腦或伺服器中央處理單元(CPU)、一或多個平行處理單元(諸如圖 2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，而不背離本文中所述之實施例的範圍。

【0082】在一些實施例中，如本文中所述之平行處理器或 GPGPU被通訊地耦接至主機/處理器核心以加速圖形

操作、機器學習操作、模式分析操作及各種通用GPU(GPGPU)功能。GPU可透過匯流排或其它互連(例如，諸如PCIe或NVLink等高速互連)被通訊地耦接至主機處理器/核心。在其它實施例中，GPU可被整合於如該等核心之相同封裝或晶片上，且透過內部處理器匯流排/互連(亦即，封裝或晶片內部)而被通訊地耦接至該些核心。無論GPU被連接的方式為何，處理器核心可依工作描述符所含之命令/指令的序列之形式將工作配置給GPU。GPU接著使用專用電路/邏輯以便有效率地處理這些命令/指令。

用於GPU至主機處理器互連之技術

【0083】圖4A顯示複數個GPU 410-413係透過高速鏈路440-443(例如，匯流排、點對點互連，等等)而被通訊地耦接至複數個多核心處理器405-406之範例架構。在一個實施例中，高速鏈路440-443根據實現支援4GB/s、30GB/s、80GB/s或更高的通訊產量。各種互連協定可被使用，包括但不限於PCIe 4.0或5.0及NVLink 2.0。然而，本發明之基本原理不限於任何特定的通訊協定或產量。

【0084】此外，在一個實施例中，GPU 410-413之二或更多者係透過高速鏈路444-445而被互連，其可使用相同於或異於那些用於高速鏈路440-443的協定/鏈路來實現。類似地，多核心處理器405-406中的二或更多者可透過高速鏈路433而被連接，高速鏈路433可以是操作在20GB/s、30GB/s、120GB/s或更高的對稱多處理器(SMP)

匯流排。可替代地，介於圖4A中所示的各種系統組件之間的所有通訊可使用相同的協定/鏈路來完成(例如，透過共同互連構造)。然而，如上所述，本發明之基本原理不限於任何特定類型的互連技術。

【0085】在一個實施例中，各多核心處理器405-406個別地經由記憶體互連430-431被通訊地耦接至處理器記憶體401-402，且各GPU 410-413係透過GPU記憶體互連450-453而被個別通訊地耦接至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。舉例而言(且非限制性)，處理器記憶體401-402及GPU記憶體420-423可以是揮發性記憶體，諸如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)和/或可以是諸如3D XPoint或Nano-Ram的非揮發性記憶體。在一個實施例中，記憶體之某部分可以是揮發性記憶體而其它部分可以是非揮發性記憶體(例如，使用二階記憶體(2LM)階層)。

【0086】如以下所述，雖然各個處理器405-406及GPU 410-413可被個別地實體地耦接至特定記憶體401-402、420-423，可實現一種統一記憶體架構，其中相同的虛擬系統位址空間(也稱為「有效位址」空間)被分配於所有的各個實體記憶體之間。例如，處理器記憶體401-402可各包含64GB的系統記憶體位址空間，而GPU記憶體420-423可各包含32GB的系統記憶體位址空間(在此範例中，導

致總共256GB的可定址記憶體)。

【0087】圖4B顯示根據一個實施例的針對介於多核心處理器407與圖形加速模組446之間的互連的額外細節。圖形加速模組446可包括整合於經由高速鏈路440被耦接至處理器407之線卡上的一或多個GPU晶片。可替代地，圖形加速模組446可被整合於如處理器407的相同封裝或晶片上。

【0088】所示的處理器407包括複數個核心460A-460D，其各具有轉換後備緩衝器461A-461D及一或多個快取462A-462D。該些核心可包括用以執行指令及處理資料之各種其它組件，其未被顯示以免混淆本發明的基本原理(例如，指令提取單元、分支預測單元、解碼器、執行單元、記錄器緩衝器，等等)。快取462A-462D可包含第1階(L1)及第2階(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多組核心460A-460D所共用。例如，處理器407的一個實施例包括24個核心，各具有其本身的L1快取、十二個共用的L2快取，以及十二個共用的L3快取。在此實施例中，L2及L3快取之一係由兩個相鄰核心所共用。處理器407及圖形加速器整合模組446係與可包括處理器記憶體401-402的系統記憶體441連接。

【0089】儲存在各個快取462A-462D、456及系統記憶體441中的資料和指令係經由透過同調匯流排464之核心間通訊來維持同調性。例如，各快取可具有與其相關的快取同調性邏輯/電路，以回應於檢測到的讀取或寫入至特

定快取線而透過同調匯流排464來通訊。在一種實現中，快取監聽協定被實現於同調匯流排464以監聽快取存取。快取監聽/同調性技術是本領域技術人員所熟知的且在此將不被詳細地描述，以免混淆本發明之基本原理。

【0090】在一個實施例中，代理電路425將圖形加速模組446通訊地耦接至同調匯流排464，其容許圖形加速模組446參與快取同調性協定而作為該些核心的同級。特別地，介面435透過高速鏈路440(例如，PCIe匯流排、NVLink，等等)提供連接至代理電路425，而介面437將圖形加速模組446連接至高速鏈路440。

【0091】在一種實現中，加速器積體電路436為了圖形加速模組446的複數個圖形處理引擎431、432、N，提供了快取管理、記憶體存取、背景管理及中斷管理服務。圖形處理引擎431、432、N可各包含各自的圖形處理單元(GPU)。可替代地，圖形處理引擎431、432、N可包含GPU內的不同類型的圖形處理引擎，諸如圖形執行單元、媒體處理引擎(例如，視頻編碼器/解碼器)、取樣器及分裂引擎。換言之，圖形加速模組可以是具有複數個圖形處理引擎431-432、N之GPU，或者圖形處理引擎431-432、N可以是整合於共同封裝、線卡或晶片上之單獨GPU。

【0092】在一個實施例中，加速器積體電路436包括記憶體管理單元(MMU)439，其用以執行各種記憶體管理功能，諸如虛擬至實體記憶體轉換(也稱為有效至真實記憶體轉換)及用以存取系統記憶體441之記憶體存取協定。

MMU 439還可包括轉換後備緩衝器(TLB)(未顯示)，其用以將虛擬/有效快取至實體/真實位址轉換。在一種實現中，快取438儲存命令及資料以供圖形處理引擎431-432、N有效率地存取。在一個實施例中，快取438及圖形記憶體433-434、N中所儲存之資料被保持與核心快取462A-462D、456及系統記憶體411同調。如上所述，這可經由代理電路425來完成，該代理電路425為了快取438及記憶體433-434、N參與快取同調性機制(例如，傳送更新至相關於處理器快取462A-462D、456上之快取線的修改/存取的快取438，並從快取438接收更新)。

【0093】一組暫存器445儲存了背景資料以用於由圖形處理引擎431-432、N所執行之執行緒，而背景管理電路448係管理執行緒背景。例如，背景管理電路448可執行保存及復原操作，以保存及復原背景切換期間之各個執行緒的背景(例如，其中第一執行緒被保存而第二執行緒被儲存以致該第二執行緒可由圖形處理引擎來執行)。例如，在背景切換上，背景管理電路448可將目前的暫存器值儲存至記憶體中的指定區(例如，由背景指標所識別)。當返回至該背景時，其可接著復原該些暫存器值。在一個實施例中，中斷管理電路447接收並處理從系統裝置所接收的中斷。

【0094】在一種實現中，來自圖形處理引擎431之虛擬/有效位址係藉由MMU 439而被轉換為系統記憶體411中的真實/實體位址。加速器積體電路436的一個實施例支援

多個(例如，4、8、16)圖形加速模組446和/或其它加速器裝置。圖形加速模組446可專用於處理器407上所執行的單一應用程式或者可被共用於多個應用程式之間。在一個實施例中，呈現了虛擬化的圖形執行環境，其中圖形處理引擎431-432、N之資源係與多個應用程式或虛擬機器(VM)共用。該些資源可被細分為「片」，其根據與VM和/或應用程式相關的處理需求和優先權而被配置至不同的VM和/或應用程式。

【0095】因此，加速器積體電路係作用為針對圖形加速模組446的對於系統的橋接，並提供位址轉換及系統記憶體快取服務。此外，加速器積體電路436可提供虛擬化協助給主機處理器，以管理圖形處理引擎之虛擬化、中斷及記憶體管理。

【0096】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所見之真實位址空間，所以任何主機處理器可使用有效位址值來直接地定址這些資源。在一個實施例中，加速器積體電路436的一個功能為圖形處理引擎431-432、N之實體分離，以致使其對系統而言呈現為獨立的單元。

【0097】如上所述，在所示的實施例中，一或多個圖形記憶體433-434、M被個別地耦接至圖形處理引擎431-432、N之各者。圖形記憶體433-434、M儲存正由圖形處理引擎431-432、N之各者所處理的指令及資料。圖形記憶體433-434、M可以是揮發性記憶體，諸如DRAM(包括堆

疊 DRAM)、GDDR 記憶體(例如, GDDR5、GDDR6)或 HBM, 和/或可以是非揮發性記憶體, 諸如 3D XPoint 或 Nano-Ram。

【0098】在一個實施例中, 為了減少透過高速鏈路 440 之資料流量, 偏移技術被用來確保圖形記憶體 433-434、M 中所儲存之資料為將最常被圖形處理引擎 431-432、N 使用且較佳地不被核心 460A-460D 使用(至少不常)之資料。類似地, 偏移機制嘗試將該些核心(且較佳地不是圖形處理引擎 431-432、N)所需的資料保持於該些核心及系統記憶體 411 的快取 462A-462D、456 內。

【0099】圖 4C 顯示加速器積體電路 436 被整合於處理器 407 內的另一實施例。在此實施例中, 圖形處理引擎 431-432、N 透過高速鏈路 440 經由介面 437 及介面 435(其, 再次地, 可利用任何形式的匯流排或介面協定)直接地通訊至加速器積體電路 436。加速器積體電路 436 可執行如關於圖 4B 所述那些的相同操作, 但由於其接近於同調匯流排 462 及快取 462A-462D、426 潛在地具有較高的產量。

【0100】一個實施例支援包括(無圖形加速模組虛擬化的)專用程序編程模型及(具有虛擬化的)共用編程模型的不同編程模型。後者可包括由加速器積體電路 436 所控制的編程模型以及由圖形加速模組 446 所控制的編程模型。

【0101】在專用程序模型的一個實施例中, 圖形處理引擎 431-432、N 係專用於單一作業系統下的單一應用程式或程序。單一應用程式可將其它應用程式請求注入至提供

VM/分割內的虛擬化的圖形引擎431-432、N。

【0102】在專用程序編程模型中，圖形處理引擎431-432、N可由多個VM/應用程式分割所共用。該些共用模型需要系統超管理器來將圖形處理引擎431-432、N虛擬化以容許各作業系統的存取。針對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。在兩種情況下，作業系統可將圖形處理引擎431-432、N虛擬化以提供對於各程序或應用程式之存取。

【0103】對於共用的編程模型，圖形加速模組446或個別圖形處理引擎431-432、N係使用程序處置來選擇程序元件。在一個實施例中，程序元件被儲存在系統記憶體411中且使用本文中所述之有效位址至真實位址轉換技術而為可定址的。程序處置可以是當以圖形處理引擎431-432、N登錄其背景時(亦即，用以將程序元件添加至程序元件鏈路列表之呼叫系統軟體)，被提供至主機程序之實現特定的值。程序處置的較低16位元可以是程序元件鏈路列表內的程序元件的偏移。

【0104】圖4D顯示範例加速器整合片490。如本文中所使用，「片」包含加速器積體電路436的處理資源之指定部分。系統記憶體411內的應用程式有效位址空間482係儲存程序元件483。在一個實施例中，程序元件483回應於來自處理器407上所執行之應用程式480的GPU調用481而被儲存。程序元件483含有相應應用程式480之程序狀態。程序元件483中所含之工作描述符(WD)484可以是應用程

式所請求的單一工作或可含有指向工作之佇列的指標。在後者的情況下，WD 484為指向應用程式之位址空間482中的工作請求佇列之指標。

【0105】圖形加速模組446和/或個別圖形處理引擎431-432、N可由系統中程序的全部或子集所共用。本發明的實施例包括一種用以設定程序狀態並傳送WD 484至圖形加速模組446以在虛擬化環境中開始工作的基礎架構。

【0106】在一種實現中，專用程序編程模型為實現特定的。在此模型中，單一程序擁有圖形加速模組446或單獨圖形處理引擎431。因為圖形加速模組446係由單一程序所擁有，所以在當圖形加速模組446被指派時的時刻，超管理器係針對其擁有的分割來將加速器積體電路436初始化，而作業系統係針對其擁有的程序來將加速器積體電路436初始化。

【0107】在操作中，加速器整合片490中之WD提取單元491提取了下一個WD 484，其包括將由圖形加速模組446的圖形處理引擎之一者所完成的工作之指示。如圖所示，來自WD 484之資料可被儲存於暫存器445中並由MMU 439、中斷管理電路447和/或背景管理電路446所使用。例如，MMU 439的一個實施例包括用以存取OS虛擬位址空間485內的分段/頁面表486的分段/頁面步行電路。中斷管理電路447可處理從圖形加速模組446所接收的中斷事件492。當執行圖形操作時，由圖形處理引擎431-432、N所產生的有效位址493被MMU 439轉換為真實位址。

【0108】在一個實施例中，相同組的暫存器445被複製給各圖形處理引擎431-432、N和/或圖形加速模組446，並且可由超管理器或作業系統來初始化。這些複製的暫存器之各者可被包括在加速器整合片490中。可由超管理器所初始化之範例暫存器被顯示在表1中。

表1-超管理器初始化暫存器

1	片控制暫存器
2	真實位址(RA)排程的處理區域指標
3	權限遮罩改寫暫存器
4	中斷向量表項目偏移
5	中斷向量表項目限制
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

【0109】可由作業系統所初始化之範例暫存器被顯示在表2中。

表2-作業系統初始化暫存器

1	程序及執行緒識別
2	有效位址(EA)背景保存/復原指標
3	虛擬位址(VA)加速器利用記錄指標
4	虛擬位址(VA)儲存分段表指標
5	權限遮罩
6	工作描述符

【0110】在一個實施例中，各WD 484係專用於特定的圖形加速模組446和/或圖形處理引擎431-432、N。其含有圖形處理引擎431-432、N欲執行其工作所需的所有資訊或者其可以是指向應用程式已設定將完成之工作的命令佇列之記憶體位置的指標。

【0111】圖4E顯示共用模型的一個實施例的額外細節。此實施例包括程序元件列表499儲存於其中的超管理器真實位址空間498。超管理器真實位址空間498為經由將作業系統495之圖形加速模組引擎虛擬化之超管理器496可存取的。

【0112】共用編程模型容許來自該系統中的分割的全部或子集之程序的全部或子集使用圖形加速模組446。有圖形加速模組446係由多個程序及分割所共用的兩種編程模型：時間片共用及圖形導向共用。

【0113】在此模型中，系統超管理器496擁有圖形加速模組446並使其功能可用於所有作業系統495。為了使圖形加速模組446支援藉由系統超管理器496之虛擬化，圖形加速模組446可遵守以下需求：1)應用程式之工作請求需為自主性的(亦即，狀態無須被維持於工作之間)；或者圖形加速模組446需提供背景保存及復原機制。2)應用程式之工作請求係由圖形加速模組446確保在指定的時間量中完成，包括任何轉換錯誤，或者圖形加速模組446提供預估該工作之處理的能力。3)當操作於該導向的共用編程模型中時，圖形加速模組446需被確保程序之間的公平性。

【0114】在一個實施例中，對於共用模型，應用程式480被要求利用圖形加速模組446類型、工作描述符(WD)、權限遮罩暫存器(AMR)值及背景保存/復原區域指標(CSRP)進行作業系統495系統呼叫。圖形加速模組446類型描述了系統呼叫的目標加速功能。圖形加速模組446類型可以是系統特定值。WD被明確地針對圖形加速模組446格式化並可以是以下之形式：圖形加速模組446命令、指向使用者定義結構之有效位址指標、指向命令佇列之有效位址指標或用以描述將由圖形加速模組446所執行之工作的任何其它資料結構。在一個實施例中，AMR值為用於目前程序之AMR狀態。被傳遞至作業系統之值係類似於設定AMR之應用程式。假如加速器積體電路436及圖形加速模組446實現不支援使用者權限遮罩改寫暫存器(UAMOR)，則作業系統可在超管理器呼叫中傳遞AMR之前，將目前的UAMOR值應用於AMR值。在將該AMR置於程序元件483之前，超管理器496可選擇性地應用當前權限遮罩改寫暫存器(AMOR)值。在一個實施例中，CSRP為含有圖形加速模組446之應用程式位址空間482中的區域之有效位址的暫存器445之一者，其用以保存並復原背景狀態。假如沒有狀態需被保存於工作之間或者當工作被預佔時，則此指標為選擇性的。背景保存/復原區域可以是固定的系統記憶體。

【0115】在接收該系統呼叫時，作業系統495可驗證應用程式480已登錄且已被提供權限來使用圖形加速模組

446。作業系統495接著呼叫具有表3中所示之資訊的超管理器496。

表 3-OS至超管理器呼叫參數

1	工作描述符(WD)
2	權限遮罩暫存器(AMR)值(潛在地遮蔽的)
3	有效位址(EA)背景保存/復原區域指標(CSRP)
4	程序ID(PID)及選擇性執行緒ID(TID)
5	虛擬位址(VA)加速器利用記錄指標(AURP)
6	儲存分段表指標(SSTP)之虛擬位址
7	邏輯中斷服務數(LISN)

【0116】在接收超管理器呼叫時，超管理器496驗證了作業系統495已登錄且已被提供權限來使用圖形加速模組446。超管理器496接著將程序元件483置入用於相應圖形加速模組446類型的程序元件鏈路表中。程序元件可包括表4中所示之資訊。

表 4-程序元件資訊

1	工作描述符(WD)
2	權限遮罩暫存器(AMR)值(潛在地遮蔽的)
3	有效位址(EA)背景保存/復原區域指標(CSRP)
4	程序ID(PID)及選擇性執行緒ID(TID)
5	虛擬位址(VA)加速器利用記錄指標(AURP)
6	儲存分段表指標(SSTP)之虛擬位址
7	邏輯中斷服務數(LISN)
8	取自超管理器呼叫參數之中斷向量表
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)

【0117】在一個實施例中，超管理器將複數個加速器整合片490暫存器445初始化。

【0118】如圖4F中所示，本發明的一個實施例利用一種經由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可定址的統一記憶體。在此實現中，執行於GPU 410-413上之操作利用相同的虛擬/有效記憶體位址空間以來存取處理器記憶體401-402且反之亦然，藉此簡化可編程性。在一個實施例中，虛擬/有效位址空間的第一部分被配置給處理器記憶體401、第二部分被配置給第二處理器記憶體402、第三部分被配置給GPU記憶體420，依此類推。整個虛擬/有效記憶

體空間(有時稱為有效位址空間)被藉此橫跨處理器記憶體401-402及GPU記憶體420-423之各者來分配，以容許任何處理器或GPU利用映射至該記憶體之虛擬位址來存取任何實體記憶體。

【0119】在一個實施例中，MMU 439A-439E中的一或多個之內的偏移/同調管理電路494A-494E確保了介於主機處理器(例如，405)與GPU 410-413的快取之間的快取同調，並且實現了指示某些類型的資料應被儲存之實體記憶體的偏移技術。雖然偏移/同調管理電路494A-494E的多個實例被顯示在圖4F中，但偏移/同調電路還可被實現在一或多個主機處理器405的MMU內和/或加速器積體電路436內。

【0120】一個實施例容許GPU裝附的記憶體420-423被映射為系統記憶體的一部分，並使用共用虛擬記憶體(SVM)技術來存取，但不會遭受與全系統快取同調性相關的典型效能缺點。GPU裝附的記憶體420-423被存取為系統記憶體而沒有繁重的快取同調性負擔之能力提供了對於GPU卸載的有利操作環境。此配置容許主機處理器405軟體設定運算元並存取計算結果，而沒有傳統I/O DMA資料副本的負擔。這樣的傳統副本涉及驅動程式呼叫、中斷及記憶體映射的I/O(MMIO)存取，其相對於簡單記憶體存取皆為無效率的。同時，存取GPU裝附的記憶體420-423而沒有快取同調性負擔之能力可能對於卸載計算之執行時間是關鍵的。在具有實質串流寫入記憶體流量的情況下，例

如，快取同調性負擔可能會顯著地減少由GPU 410-413所見的有效寫入頻寬。運算元設定的效率、結果存取的效率及GPU計算的效率皆參與判定GPU卸載之有效性。

【0121】在一種實現中，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來驅動。偏移表可被使用，例如，其可以是頁面粒度的結構(亦即，以記憶體頁面之粒度來控制)，其每個GPU裝附的記憶體頁面包括1或2位元。偏移表可被實現於一或多個GPU裝附的記憶體420-423的竊取記憶體範圍中，其在GPU 410-413中有或沒有偏移快取(例如，用以快取偏移表中的頻繁/最近使用的項目)。可替代地，整個偏移表可被保持於GPU內。

【0122】在一種實現中，與GPU裝附的記憶體420-423之各存取相關的偏移表項目係在對於該GPU記憶體的實際存取前被存取，造成以下操作。首先，來自在GPU偏移中發現其頁面的GPU 410-413的本地請求被直接地傳遞至相應的GPU記憶體420-423。來自在主機偏移中發現其頁面之GPU的本地請求被傳遞至處理器405(例如，透過如以上所討論的高速鏈路)。在一個實施例中，來自在主機處理器偏移中發現該請求頁面之處理器405的請求完成了該請求似的一般記憶體讀取。可替代地，指向GPU偏移頁面之請求可被傳遞至GPU 410-413。假如GPU目前並非正在使用該頁面，則GPU可接著將該頁面轉換至主機處理器偏移。

【0123】頁面之偏移狀態可藉由以下中的任一者來改

變：軟體為基的機制、硬體協助之軟體為基的機制或(針對有限組的情況)純粹硬體為基的機制。

【0124】一種用於改變偏移狀態之機制利用了API呼叫(例如，OpenCL)，其接著呼叫GPU之裝置驅動程式，其接著傳送訊息(或將一命令描述符放入佇列)至GPU以指引其改變偏移狀態，並且(針對一些轉換)在該主機中執行快取清除操作。快取清除操作對於從主機處理器405偏移到GPU偏移之轉換是必要的，但對於相反轉換並非必要的。

【0125】在一個實施例中，快取同調性係藉由主機處理器405暫時地使GPU偏移的頁面無法快取來維持。為了存取這些頁面，處理器405可從GPU 410請求存取，取決於實現該GPU 410可以或可以不立刻准予存取。因此，為了減少介於處理器405與GPU 410之間的通訊，有利的是確保GPU偏移的頁面為那些由GPU所需的，而非主機處理器405所需的，且反之亦然。

圖形處理管線

【0126】圖5顯示根據實施例的圖形處理管線500。在一個實施例中，圖形處理器可實現所示的圖形處理管線500。圖形處理器可被包括在如本文中所述的平行處理子系統內，諸如圖2的平行處理器200，在一個實施例中，其為圖1的平行處理器112的變體。各個平行處理系統可經由如本文中所述的平行處理單元(例如，圖2的平行處理單元202)中的一或多個實例來實現圖形處理管線500。例如，

著色器單元(例如，圖3的圖形多處理器234)可組態成執行頂點處理單元504、鑲嵌控制處理單元508、鑲嵌評估處理單元512、幾何處理單元516及片段/像素處理單元524中的一或多者的功能。資料組譯器502、基元組譯器506、514、518、鑲嵌單元510、柵格化器522及柵格操作單元526之功能還可由處理叢集(例如，圖3的處理叢集214)及相應分割單元(例如，圖2的分割單元220A-220N)內的其它處理引擎來執行。圖形處理管線500還可使用針對一或多個功能的專用處理單元來實現。在一個實施例中，圖形處理管線500中的一或多個部分可由通用處理器(例如，CPU)內的平行處理邏輯來執行。在一個實施例中，圖形處理管線500中的一或多個部分可經由記憶體介面528(其可以是圖2的記憶體介面218的實例)來存取晶載記憶體(例如，圖2的平行處理器記憶體222)。

【0127】在一個實施例中，資料組譯器502為一種收集表面及基元之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)輸出至頂點處理單元504。頂點處理單元504為執行頂點著色器程式、照亮及轉換如由頂點著色器程式所指明的頂點資料的可編程執行單元。頂點處理單元504係讀取被儲存於快取、本地或系統記憶體中之資料以用於處理頂點資料，並可被編程以將頂點資料從物件為基的座標表示轉換為世界空間座標空間或正規化裝置座標空間。

【0128】基元組譯器506的第一實例係接收來自頂點

處理單元 50 之頂點屬性。基元組譯器 506 係根據需要讀取已儲存的頂點屬性，並建構圖形處理基元以供鑲嵌控制處理單元 508 處理。圖形基元包括三角形、線段、點、補丁(等等)，如由各種圖形處理應用程式編程介面(API)所支援的。

【0129】鑲嵌控制處理單元 508 將輸入頂點視為幾何補丁之控制點。控制點係從來自該補丁(例如，該補丁之基礎)的輸入表示被轉換為適於藉由鑲嵌評估處理單元 512 之表面評估的表示。鑲嵌控制處理單元 508 還可計算幾何補丁之邊緣的鑲嵌因數。鑲嵌因數適用於單一邊緣並量化與該邊緣相關細節的觀點相依程度。鑲嵌單元 510 被組態成接收補丁之邊緣的鑲嵌因數，並將該補丁鑲嵌入多個幾何基元(諸如線、三角形或四邊形基元)，其被傳輸至鑲嵌評估處理單元 512。鑲嵌評估處理單元 512 操作於子分割補丁之參數化座標上，以產生用於與幾何基元相關的各頂點之表面表示及頂點屬性。

【0130】基元組譯器 514 的第二實例接收來自鑲嵌評估處理單元 512 的頂點屬性、根據需要讀取已儲存的頂點屬性，並建構圖形處理基元以供幾何處理單元 516 處理。幾何處理單元 516 為執行幾何著色器程式以轉換如由幾何著色器程式所指明的接收自基元組譯器 514 之圖形基元的可編程執行單元。在一個實施例中，幾何處理單元 516 被編程以將圖形基元子分割為一或多個新的圖形基元並計算被用來將新的圖形基元柵格化之參數。

【0131】在一些實施例中，幾何處理單元516可在幾何串流中添加或刪除元件。幾何處理單元516將該些參數及指明新圖形基元之頂點輸出至基元組譯器518。基元組譯器518從幾何處理單元516接收該些參數及頂點並建構圖形基元以供視埠縮放、剔除及修剪單元520處理。幾何處理單元516讀取其被儲存於平行處理器記憶體或系統記憶體中之資料以用於處理該幾何資料。視埠縮放、剔除及修剪單元520係執行修剪、剔除及視埠縮放並將處理過的圖形基元輸出至柵格化器522。

【0132】柵格化器522可執行深度剔除及其它深度為基的最佳化。柵格化器522也在新的圖形基元上執行掃描轉換，以產生片段並且將那些片段及相關的覆蓋資料輸出至片段/像素處理單元524。片段/像素處理單元524為一種可編程執行單元，其被組態成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轉換如由片段或像素著色器程式所指明的接收自柵格化器522之片段或像素。例如，片段/像素處理單元524可被編程以執行操作，包括但不限於紋理映射、著色、混合、紋理校正及個別校正，以產生其被輸出至柵格操作單元526之已著色片段或像素。片段/像素處理單元524可讀取被儲存於平行處理器記憶體或系統記憶體中之資料以供處理該片段資料時使用。根據對於該些處理單元組態的取樣率，片段或像素著色器程式可被組態成在樣本、像素、磚片或其它粒度上著色。

【0133】柵格操作單元526為一種處理單元，其執行

柵格操作，包括但不限於模板、z測試、混合，等等，並將像素資料輸出為已處理的圖形資料以被儲存於圖形記憶體中(例如，如圖2中的平行處理器記憶體222，和/或如圖1中的系統記憶體104)、以被顯示在一或多個顯示裝置110上或者以利一或多個處理器102或平行處理器112之一者進一步處理。在一些實施例中，柵格操作單元526被組態成壓縮被寫入至記憶體之z或顏色資料，並且解壓縮被讀取自記憶體之z或顏色資料。

機器學習概述

【0134】機器學習演算法是一種可根據一組資料來學習的演算法。機器學習演算法的實施例可被設計成在資料集內建模高階抽象概念。例如，影像辨識演算法可被用來判定既定輸入係屬於數個種類之何者；迴歸演算法可輸出給定輸入中之數值；而型態辨識演算法可被用來產生經轉換文字或者執行文字至語音和/或語音辨識。

【0135】機器學習演算法的一種範例類型為神經網路。有許多類型的神經網路；神經網路的一種簡單類型為前饋網路。前饋網路可被實現為非週期性圖表，其中節點被配置於層中。通常，前饋網路拓撲包括輸入層及輸出層，其由至少一個隱藏層所分離。隱藏層將由輸入層所接收的輸入轉換為一種有用於產生輸出層中之輸出的表示。網路節點係經由邊緣而被完全地連接至相鄰層中之節點，但並無邊緣介於各層內的節點之間。在前饋網路之輸入層

的節點所接收之資料係經由啟動函數而被傳播(亦即，「前饋」)至輸出層的節點，該啟動函數係根據與連接該些層之邊緣的各者個別地相關的係數(「權重」)來計算該網路中各連續層之節點的狀態。根據由執行中演算法所表示的特定模型，來自神經網路演算法之輸出可具有多種形式。

【0136】在機器學習演算法可被用來建模特定問題之前，該演算法係使用訓練資料集來訓練。訓練神經網路涉及了選擇網路拓撲、使用表示由該網路所建模之問題的一組訓練資料，以及調整權重直到該網路模型以最小錯誤執行於該訓練資料集的所有實例為止。例如，在針對神經網路的受監督學習訓練程序期間，由該網路回應於表示訓練資料集中之實例的輸入所產生的輸出係與該實例之「正確」標示的輸出進行比較，表示介於該輸出與該標示的輸出之間的差異之錯誤信號被計算，並且與該些連接相關的權重隨著該錯誤信號透過該網路之該些層被向後傳遞而被調整以將該錯誤最小化。當從該訓練資料集之實例所產生的輸出之各者的錯誤被最小化時，該網路被視為「經訓練」。

【0137】機器學習演算法之準確度可受到用以訓練該演算法之資料集的品質顯著地影響。訓練程序可以是計算上密集的，且對於傳統通用處理器可能需要顯著的時間量。因此，平行處理硬體被用來訓練許多類型的機器學習演算法。對於最佳化神經網路之訓練，這是特別有用的，

因為在調整神經網路中之係數時所執行的計算自然地參與平行實現。特別是，許多機器學習演算法及軟體應用程式已被調適為利用通用圖形處理裝置內的平行處理硬體。

【0138】圖6為機器學習軟體堆疊600之一般化圖形。機器學習應用程式602可被組態成使用訓練資料集來訓練神經網路或者使用經訓練的深神經網路來實現機器智慧。機器學習應用程式602可包括用於可被用來在部署前訓練神經網路之神經網路和/或專門軟體的訓練及推理功能。機器學習應用程式602可實現任何類型的機器智慧，包括但不限於影像辨識、映射和本地化、自主性導航、語音合成、醫藥成像或語言翻譯。

【0139】機器學習應用程式602的硬體加速可經由機器學習框架604來啟用。機器學習框架604可提供機器學習基元之庫。機器學習基元為常由機器學習演算法所執行的基本操作。若沒有機器學習框架604，則機器學習演算法之開發者將被要求建立並最佳化其與機器學習演算法相關的主計算邏輯，然後隨著新的平行處理器被開發而再最佳化該計算邏輯。取而代之地，機器學習應用程式可被組態成使用由機器學習框架604所提供的基元來執行必要的計算。範例基元包括張量卷積、啟動功能和池化，其為訓練卷積神經網路(CNN)時所執行的計算操作。機器學習框架604還可提供基元以實現由許多機器學習演算法(諸如矩陣和向量操作)來執行的基本線性代數子程式。

【0140】機器學習框架604可處理接收自機器學習應

用程式602之輸入資料並產生適當的輸入給計算框架606。計算框架606可提取被提供給GPGPU驅動程式608的主要指令，以致使機器學習框架604利用經由GPGPU硬體610之硬體加速而不需要機器學習框架604具有GPGPU硬體610之架構的內部知識。此外，計算框架606可跨多種類型及世代的GPGPU硬體610來致使機器學習框架604之硬體加速。

GPGPU機器學習加速

【0141】圖7顯示根據實施例的高度平行通用圖形處理單元700。在一個實施例中，通用處理單元(GPGPU)700可被組態成對於處理與訓練深神經網路相關的計算工作負載之類型特別地有效率。此外，GPGPU 700可被直接地鏈路至GPGPU的其它實例，以建立多GPU叢集來增進特別是用於深神經網路之訓練速度。

【0142】GPGPU 700包括用以致使與主機處理器之連接的主機介面702。在一個實施例中，主機介面702為PCI Express介面。然而，主機介面還可以是供應商特定的通訊介面或通訊組織。GPGPU 700從主機處理器接收命令並使用全域排程器704來將與那些命令相關的執行緒分配至一組計算叢集706A-706H。該些計算叢集706A-706H共用了快取記憶體708。快取記憶體708可作為用於計算叢集706A-706H內的快取記憶體的高階快取。

【0143】GPGPU 700包括經由一組記憶體控制器712A-712B而與計算叢集706A-H耦接之記憶體714A-

714B。在各個實施例中，記憶體714A-714B可包括各個類型的記憶體裝置，包括動態隨機存取記憶體(DRAM)或圖形隨機存取記憶體，諸如同步圖形隨機存取記憶體(SGRAM)，包括圖形雙資料速率(GDDR)記憶體或3D堆疊記憶體，包括但不限於高頻寬記憶體(HBM)。

【0144】在一個實施例中，各計算叢集706A-706H包括諸如圖4A之圖形多處理器400的一組圖形多處理器。計算叢集之圖形多處理器包括多種類型的整數及浮點邏輯單元，其可執行在包括適於機器學習計算之精確度範圍的計算操作。例如以及在一個實施例中，在計算叢集706A-706H之各者中的浮點單元之子集可被組態成執行16位元或32位元浮點操作，而浮點單元的不同子集可被組態成執行64位元浮點操作。

【0145】GPGPU 700的多個實例可被組態成操作為計算叢集。由計算叢集所用於同步化及資料交換的通訊機制係跨越實施例而改變。在一個實施例中，GPGPU 700的多個實例係透過主機介面702來通訊。在一個實施例中，GPGPU 700包括I/O集線器709，其將GPGPU 700與GPU鏈路710耦接，該GPU鏈路710致使直接連接至GPGPU的其它實例。在一個實施例中，GPU鏈路710被耦接至專用GPU至GPU橋，其致使GPGPU 700的多個實例之間的通訊及同步化。在一個實施例中，GPU鏈路710係與高速互連耦接以將資料傳輸及接收至其它GPGPU或平行處理器。在一個實施例中，GPGPU 700的多個實例被置於個別的資料處理

系統中並經由網路裝置(其為經由主機介面702可存取的)來通訊。在一個實施例中，除了主機介面702之外(或當作其替代)，GPU鏈路710可被組態成致使連接至主機處理器。

【0146】雖然GPGPU 700之所示組態可被組態成訓練神經網路，但一個實施例提供了GPGPU 700之替代組態，其可被組態以供部署於高效能或低功率推理平台內。在推理組態中，相對於訓練組態，GPGPU 700包括較少的計算叢集706A-706H。此外，與記憶體714A-714B相關的記憶體技術可在推理與訓練組態之間有差異。在一個實施例中，GPGPU 700的推理組態可支援推理特定指令。例如，推理組態可提供一或多個8位元整數內積指令之支援，其常在已部署神經網路的推理操作期間使用。

【0147】圖8顯示根據實施例的多GPU計算系統800。多GPU計算系統800可包括經由主機介面開關804來耦接至多GPGPU 806A-806D的處理器802。在一個實施例中，主機介面開關804為PCI Express開關裝置，其將處理器802耦接至PCI Express匯流排，處理器802可透過該PCI Express匯流排而與該組GPGPU 806A-806D通訊。多個GPGPU 806A-806D之各者可以是圖7之GPGPU 700的實例。GPGPU 806A-806D可經由一組高速點對點GPU至GPU鏈路816來互連。高速GPU至GPU鏈路可經由專用GPU鏈路(諸如圖7中之GPU鏈路710)來連接至GPGPU 806A-806D之各者。P2P GPU鏈路816致使GPGPU 806A-806D的各者之間

的直接通訊而不需要透過主機介面匯流排(其連接至處理器802)來通訊。利用導引至P2P GPU鏈路之GPU至GPU流量，主機介面匯流排保持可用於系統記憶體存取或者與多GPU計算系統800的其它實例通訊，例如，經由一或多個網路裝置。雖然在所示的實施例中，GPGPU 806A-806D係經由主機介面開關804而連接至處理器802，但在一個實施例中，處理器802包括對於P2P GPU鏈路816之直接支援並可直接地連接至GPGPU 806A-806D。

機器學習神經網路實現

【0148】由本文中所述之實施例所提供的計算架構可被組態成執行多類型的平行處理，其特別適於用於機器學習之訓練及部署神經網路。神經網路可被一般化成為具有圖形關係之功能的網路。如本領域中眾所周知，有用於機器學習的多種類型的神经网络實現。如先前所述的，神经网络的一種範例類型為前饋網路。

【0149】神經網路的第二範例類型為卷積神經網路(CNN)。CNN為用以處理具有已知、柵格狀拓撲之資料(諸如影像資料)的一種專門前饋神經網路。因此，CNN常用於計算視覺和影像辨識應用，但其還可被用在其它類型的圖案辨識，諸如語音及語言處理。CNN輸入層中的節點被組織為一組「過濾器」(由視網膜中所發現的個別欄位所驅使的特徵檢測器)，且各組過濾器之輸出被傳播至該網路之連續層中的節點。CNN之計算包括將卷積數學運算應

用於各過濾器以產生該過濾器之輸出。卷積為由兩個函數所執行用以產生兩個原始函數之一的修改版本之第三函數之特殊種類的數學運算。於卷積網路術語中，至卷積的第一函數可被稱為輸入，而第二函數可被稱為卷積內核。輸出可被稱為特徵映射。例如，至卷積層之輸入可以是定義輸入影像之各種顏色成分的多維陣列資料。卷積內核可以是參數的多維陣列，其中該些參數係由神經網路的訓練程序所調適。

【0150】遞歸神經網路(RNN)為前饋神經網路之家族，其包括介於層之間的回饋連接。RNN係藉由共用跨神經網路之不同部分的參數資料以致使連續資料之建模。RNN之架構包括週期。該些週期代表變數的目前值對於未來時刻的其本身值的影響，因為來自RNN之輸出資料的至少一部分被使用為用以處理序列中之後續輸入的回饋。此特徵使得RNN特別有用於語言處理，因為語言資料可被組成的可變本質。

【0151】以下所述之圖形呈現範例的前饋、CNN與RNN網路，以及描述用以個別地訓練並部署那些類型的網路之各者的一般程序。將理解，關於本文中所述的任何特定實施例的這些描述是範例的而非限制的，且一般來說，所顯示之概念可被通常地應用於深神經網路及機器學習技術。

【0152】上述範例神經網路可被用來執行深學習。深學習為使用深神經網路之機器學習。用於深學習之深神經

網路為由多個隱藏層所組成的人工神經網路，不同於僅包括單一隱藏層的淺神經網路。較深的神經網路對於訓練通常在計算上是較為密集的。然而，網路的額外隱藏層致使多步驟圖案辨識，其導致相對於淺機器學習技術的降低輸出錯誤。

【0153】用於深學習之深神經網路通常包括用以執行特徵辨識之前端網路，其耦接至後端網路，該後端網路代表一種數學模型，其可根據提供至該模型之特徵表示來執行操作(例如，物件分類、語音辨識，等等)。深學習致使機器學習被執行而不需要針對該模型執行手動調整特徵工程。取而代之地，深神經網路可根據輸入資料內的統計結構或相關性來學習特徵。經學習的特徵可被提供至一種可將測得特徵映射至輸出的數學模型。由該網路所使用之數學模型一般針對待執行的特定工作而被特殊化，並且不同的模型將用以執行不同的工作。

【0154】一旦神經網路被構成，學習模型可被應用至該網路以訓練該網路來執行特定工作。該學習模型描述如何調整該模型內的權重以減少該網路之輸出錯誤。錯誤的後向傳播為用以訓練神經網路的常見方法。輸入向量被提交至該網路以供處理。該網路之輸出係使用損失函數來與所欲輸出進行比較，並針對輸出層中之神經元的各者來計算誤差值。誤差值被接著向後傳播，直到各神經元具有約略地表示其對於原始輸出的貢獻的相關誤差值。該網路可接著使用演算法(諸如隨機梯度下降演算法)自那些錯誤學

習，以更新該神經網路之權重。

【0155】圖9A及9B顯示範例卷積神經網路。圖9A顯示CNN內的各個層。如圖9A中所示，用以建模影像處理之範例CNN可接收描述輸入影像之紅、綠及藍(RGB)成分的輸入902。輸入902可由多個卷積層(例如，卷積層904、卷積層906)來處理。來自多個卷積層之輸出可選擇性地由一組完全連接層908來處理。如先前針對前饋網路所述的，完全連接層中之神經元具有對於先前層中之所有啟動的完全連接。來自完全連接層908之輸出可被用來產生來自該網路之輸出結果。完全連接層908內的啟動可使用取代卷積的矩陣乘法來計算。並非所有CNN的實現皆利用完全連接層908。例如，在一些實現中，卷積層906可產生輸出給該CNN。

【0156】卷積層被稀疏地連接，其不同於完全連接層908中所見之傳統神經網路組態。傳統神經網路層被完全連接，以致使每一輸出單元係與每一輸入單元互動。然而，如圖所示，卷積層被稀疏地連接，因為欄位之卷積的輸出被輸入(取代該欄位中之各節點的個別狀態值)至後續層之節點。與該些卷積層相關的內核係執行卷積操作，其輸出被傳送至下一層。在卷積層內所執行的維度減少為致使CNN縮放以供處理大影像之一種形態。

【0157】圖9B顯示CNN之卷積層內的範例計算階段。CNN之卷積層912的輸入可在卷積層914的三個階段中被處理。該三個階段可包括卷積階段916、檢測器階段918和池

化階段920。卷積層914可接著輸出資料至連續卷積層。該網路的最終卷積層可產生輸出特徵映射資料或提供輸入至完全連接層，例如，用以產生輸入的分類值給該CNN。

【0158】在卷積階段916中，平行地執行數個卷積以產生一組線性啟動。卷積階段916可包括仿射變換，其為可被指明為線性變換加上轉換之任何變換。仿射變換包括旋轉、轉換、縮放及這些變換之組合。卷積階段計算函數(例如，神經元)之輸出，其被連接至輸入中之特定區，其可被判定為與該神經元相關的本地區。神經元計算介於該些神經元的權重與該些神經元所連接至之該本地輸入中的該區之間的內積。來自卷積階段916之輸出係定義一組線性啟動，其由卷積層914之連續階段來處理。

【0159】該些線性啟動可由檢測器階段918來處理。在檢測器階段918中，各線性啟動係由非線性啟動函數來處理。非線性啟動函數係增加整體網路之非線性性質而不影響卷積層之個別欄位。數種類型的非線性啟動函數可被使用。一種特別類型是整流線性單元(ReLU)，其使用定義為 $f(x) = \max(0, x)$ 之啟動函數，以致使啟動係以零為臨限值。

【0160】池化階段920係使用以附近輸出之摘要統計來替換卷積層906之輸出的池化函數。池化函數可被用來將轉換不變性引入神經網路，以致使對於輸入之小轉換不會改變池化的輸出。對於本地轉換之不變性可能在輸入資料中之特徵的存在比該特徵之精確位置更為重要的情境中

是有用的。多種類型的池化函數可被用於池化階段920期間，包括最大池化、平均池化及12-norm池化。此外，一些CNN實現不包括池化階段。取而代之地，此類實現係以相對於先前卷積階段具有增加的跨步的額外卷積階段來代替。

【0161】來自卷積層914之輸出可接著由下一層922來處理。下一層922可以是額外卷積層或者完全連接層908之一。例如，圖9A的第一卷積層904可輸出至第二卷積層906，而第二卷積層可輸出至完全連接層908的第一層。

【0162】圖10顯示範例遞歸神經網路1000。在遞歸神經網路(RNN)中，網路的先前狀態係影響網路的目前狀態的輸出。RNN可使用多種功能來以多種方式被建立。RNN之使用通常圍繞著使用數學模型以根據輸入之先前序列來預測未來。例如，RNN可被用來執行統計語言建模來根據字詞之先前序列而預測將來的字詞。所示的RNN 1000可被描述為具有接收輸入向量的輸入層1002、用以實施遞迴函數的隱藏層1004、用以致使先前狀態之「記憶體」的回饋機制1005及用以輸出結果的輸出層1006。RNN 1000係根據時間步驟來操作。在既定時間步驟之RNN的狀態係經由回饋機制1005根據先前時間步驟而被影響。針對既定時間步驟，隱藏層1004之狀態係由先前狀態以及目前時間步驟之輸入來定義。在第一時間步驟之初始輸入(x_1)可由隱藏層1004來處理。第二輸入(x_2)可由隱藏層1004使用在該初始輸入(x_1)之處理期間所判定的狀態資訊來處理。既定

狀態可被計算為 $s_t = f(Ux_t + Ws_{t-1})$ ，其中 U 及 W 為參數矩陣。函數 f 通常為非線性，諸如雙曲線正切函數(Tanh)或整流函數 $f(x) = \max(0, x)$ 之變體。然而，用於隱藏層 1004 之特定數學函數可根據 RNN 1000 之特定實現細節而變化。

【0163】除了所述的基本 CNN 及 RNN 網路之外，對於那些網路的變化可被致使。一種範例 RNN 變體為長短期記憶體 (LSTM) RNN。LSTM RNN 能夠學習對於處理較長的語言序列可能是必要的之長期依存性。CNN 之變體為一種卷積深信網路，其具有類似於 CNN 之結構且係以一種類似於深信網路之方式來訓練。深信網路 (DBN) 為一種生成的神經網路，其由隨機(任意)變數的多個層所組成。DBN 可使用貪心非監督學習而被層接層地訓練。DBN 之經學習的權重可接著藉由判定針對該神經網路之最佳初始組的權重被用來提供預先訓練神經網路。

【0164】圖 11 顯示深神經網路之訓練及部署。一旦既定網路已針對一工作而被構成，則神經網路係使用訓練資料集 1102 而被訓練。各種訓練框架 1104 已被開發來致使訓練程序之硬體加速。例如，圖 6 之機器學習框架 604 可被組態成訓練框架 604。訓練框架 604 可連結入未訓練神經網路 1106 並致使未訓練神經網使用本文中所述之平行處理資源而被訓練，以產生經訓練神經網 1108。

【0165】為了開始訓練程序，初始權重可被隨機地選擇或者藉由使用深信網路來預先訓練。訓練週期接著以受監督或未監督中任一者的方式被執行。

【0166】受監督學習係一種學習方法，其中訓練被執行行為中介操作，諸如當訓練資料集1102包括與針對輸入的所欲輸出配對的輸入時，或其中訓練資料集包括具有已知輸出之輸入且神經網路之輸出被手動地分級。該網路處理該些輸入並將所得輸出與一組預期或所欲輸出進行比較。錯誤接著透過系統回傳。訓練框架1104可調整控制未訓練神經網路1106之權重。訓練框架1104可提供工具來監視未訓練神經網路1106如何適當地根據已知的輸入資料以朝向一適於產生正確答案之模型收斂。隨著該網路之權重被調整以改良由神經網路所產生的輸出，訓練程序係重複地發生。訓練程序可持續，直到神經網路達到與經訓練神經網路1108相關之統計上理想的準確度。經訓練神經網路1108可接著被部署以實現任何數目的機器學習操作。

【0167】未監督學習為一種網路嘗試使用未標示資料來訓練其本身的學習方法。因此，針對未監督學習，訓練資料集1102將包括輸入資料而沒有任何相關的輸出資料。未訓練神經網路1106可學習未標示輸入內的學習群集，並可判定個別輸入如何與整體資料集相關。未監督訓練可被用來產生自行組織映射，其為一種能夠執行可用來減少資料之維度的操作之經訓練神經網路1107的類型。未監督訓練還可被用來執行異常檢測，其容許從資料的正常型態偏離之輸入資料集中的資料點的識別。

【0168】受監督及未監督訓練之變化還可被利用。半監督學習為一種訓練資料集1102包括相同分佈之已標示及

未標示資料的混合之技術。遞增學習為受監督學習之變體，其中輸入資料被持續地用來進一步訓練模型。遞增學習致使經訓練神經網路1108調適於新資料1112而不會忘記初始訓練期間之網路內所教導的知識。

【0169】無論受監督或未監督，特別是針對深神經網路之訓練程序可能對於單一計算節點是太計算密集的。取代使用單一計算節點，計算節點之分散式網路可被用來加速訓練程序。

【0170】圖12為顯示分散式學習的方塊圖。分散式學習為一種使用多個分散式計算節點來執行神經網路之受監督或未監督訓練的訓練模型。分散式計算節點可各包括一或多個主機處理器及一或多個通用處理節點，諸如圖7中之高度平行通用圖形處理單元700。如圖所示，分散式學習可執行模型平行性1202、資料平行性1204或模型和資料平行性之組合1204。

【0171】在模型平行性1202中，分散式系統中的不同計算節點可針對單一網路之不同部分來執行訓練計算。例如，神經網路之各層可由分散式系統中的不同處理節點來訓練。模型平行性之優點包括縮放至特別大的模型之能力。將與神經網路之不同層相關的計算分離致使了非常大的神經網路(其中所有層之權重將不配適入單一計算節點之記憶體)之訓練。在一些實例中，模型平行性可特別有用於執行大型神經網路之未監督訓練。

【0172】在資料平行性1204中，分散式網路的不同節

點具有該模型的完整實例且各節點係接收資料之不同部分。來自不同節點之結果被接著結合。雖然資料平行性的不同方式是可能的，但資料平行性訓練皆需要一種結合結果並將各節點間之模型參數同步化的技術。結合資料的範例方式包括參數平均及更新為基的資料平行性。參數平均係訓練該訓練資料之子集上的各節點，並將全域參數(例如，權重、偏移)設為來自各節點之參數的平均。參數平均係使用保存參數資料的中央參數伺服器。更新為基的資料平行性係類似於參數平均，除了對於該模型的更新被轉移，而不是將參數從節點轉移至參數伺服器。此外，更新為基的資料平行性可用一種分散方式來執行，其中該些更新被壓縮並在節點之間轉移。

【0173】結合的模型及資料平行性1206可被實現在例如各計算節點包括多個GPU的分散式系統中。各節點可具有該模型的完全實例，其中各節點內的分離GPU被用來訓練該模型之不同部分。

【0174】相對於單一機器上之訓練，分散式訓練具有增加的負擔。然而，本文中所述之平行處理器及GPGPU可各實現多種技術來減少分散式訓練之負擔，包括用以致使高頻寬GPU至GPU資料轉移及加速遠端資料同步化的技術。

範例機器學習應用

【0175】機器學習可被應用來解決多種技術問題，包

括但不限於電腦視覺、自動駕駛和導航、語音辨識及語言處理。電腦視覺傳統上已成為針對機器學習應用的最活躍研究領域之一。電腦視覺之應用的範圍從再生人類視覺能力(諸如辨識人臉)至產生新種類的視覺能力。例如，電腦視覺應用可被組態成從視頻中可見的物體中所引發的振動來辨識音波。平行處理器加速機器學習係致使電腦視覺應用使用相較於先前可行者顯著地更大的訓練資料集來訓練，並致使推理系統使用低功率平行處理器來被部署。

【0176】平行處理器加速機器學習具有自動駕駛應用，包括巷道和道路標誌辨識、障礙閃避、導航及駕駛控制。加速機器學習技術可被用來根據定義對於特定訓練輸入之適當回應的資料集來訓練駕駛模型。本文中所述之平行處理器可致使用於自動駕駛解決方案的越來越複雜的神經網路之快速訓練，並致使一種適於整合在自動車內的行動平台中的低功率推理處理器之部署。

【0177】平行處理器加速深神經網路已致使自動語音辨識(ASR)的機器學習方案。ASR包括產生計算既定輸入聲波序列下最可能的語言序列之函數。使用深神經網路之加速機器學習已致使先前用於ASR的隱藏馬可夫(Markov)模型(HMM)及高斯(Gaussian)混合模型(GMM)的替換。

【0178】平行處理器加速機器學習還可被用來加速自然語言處理。自動學習程序可利用統計推理演算法來產生能抵抗錯誤或不熟悉輸入的模型。範例自然語言處理器應用包括人類語言之間的自動機器翻譯。

【0179】用於機器學習的平行處理平台可被劃分為訓練平台及部署平台。訓練平台通常是高度平行的，且包括用以加速多GPU單節點訓練及多節點多GPU訓練的最佳化。適於訓練的範例平行處理器包括圖7的高度平行通用圖形處理單元700及圖8的多GPU計算系統800。反之，已部署的機器學習平台通常包括較低功率的平行處理器，其適於諸如相機、自動機器人及自動車等產品的使用。

【0180】圖13顯示一種適於使用經訓練的模型來執行推理的範例推理系統單晶片(SOC)1300。SOC 1300可整合包括媒體處理器1302、視覺處理器1304、GPGPU 1306及多核心處理器1308之處理組件。SOC 1300可額外地包括晶載記憶體1305，其可致使可由各處理組件所存取之共用的晶載資料池。處理組件可針對低功率操作來最佳化，以致使對於包括自動車及自動機器人的多種機器學習平台之部署。例如，SOC 1300的一種實現可被使用為自動車的主控制系統的一部分。當SOC 1300被組態為用於自動車時，SOC被設計並組態成符合部署管轄權之相關的功能安全性標準。

【0181】在操作期間，媒體處理器1302及視覺處理器1304可配合加速電腦視覺操作來運作。媒體處理器1302可致使多個高解析度(例如，4K、8K)視頻串流的低延遲時間解碼。經解碼的視頻串流可被寫入至晶載記憶體1305中的緩衝器。視覺處理器1304可接著解析該經解碼視頻並對於該經解碼視頻之框執行初步處理操作，以準備使用經訓練

的影像辨識模型來處理該些框。例如，視覺處理器1304可加速被用來對於高解析度視頻資料執行影像辨識之CNN的卷積操作，而後端模型計算係由GPGPU 1306來執行。

【0182】多核心處理器1308可包括用以協助由媒體處理器1302和視覺處理器1304所執行的資料轉移及共用記憶體操作之排序和同步化的控制邏輯。多核心處理器1308還可作用為用以執行可利用GPGPU 1306之推理計算能力的軟體應用程式的應用處理器。例如，導航和駕駛邏輯之至少一部分可被實現在多核心處理器1308上所執行的軟體。此軟體可直接地發送計算工作負載至GPGPU 1306，或者該計算工作負載可被發送至多核心處理器1308，其可將那些操作之至少一部分卸載至GPGPU 1306。

【0183】GPGPU 1306可包括計算叢集，諸如高度平行通用圖形處理單元700內的計算叢集706A-706H的低功率組態。GPGPU 1306內的計算叢集可支援被明確地最佳化以執行推理計算於經訓練神經網路上的指令。例如，GPGPU 1306可支援用以執行諸如8位元及4位元整數向量操作的低精確度計算的指令。

使用低和高精確度的混合推理

【0184】GPGPU加速的計算致使了對於GPGPU的應用的平行部分的卸載，而程式碼的其餘部分在主處理器(例如，CPU)上執行。如本文所述，GPGPU包含具有執行整數和浮點運算之能力的計算單元。通常，這些運算在計

算單元中是獨占的，其被分配以執行將會閘控電源或以其它方式禁用負責浮點運算的計算元件之整數運算。相反的情況也有可能，計算單元在執行浮點運算時可以禁用整數分量。這樣的配置能夠降低每個計算單元的運行功率消耗。或者，一些計算單元可以被配置成以多個精確度中的一個來選擇性地執行計算。例如，計算單元可以被配置成執行FP32運算或雙FP16運算。配置成執行32位元整數運算的計算單元可以執行四個同時的8位元整數運算。選擇性的多精確度或多資料類型的計算單元可以使計算單元具有強大的功能，這些功能也是省電的，因為可以由單一計算單元執行各種運算，而每個計算單元內的閒置邏輯單元被禁用以減少運行功率消耗。

【0185】然而，也有可能藉由利用計算單元內的其它閒置邏輯單元來運算使計算單元省電。在這裡描述的實施例中，計算單元內的可變精確度和/或可變資料類型邏輯組件可以被同時操作，使得不用於服務初始運算的計算單元可以能夠處理一或多個額外的運算。例如並且在一個實施例中，具有浮點和整數邏輯單元的計算單元可以同時處理整數和浮點運算。在一個實施例中，配置成選擇性地執行32位元運算或雙16位元運算的計算單元可以被配置以執行32位元運算和雙16位元運算或32位元運算以及多個獨立的16位元運算。在一個實施例中，藉由允許將不同類型的多個指令發佈到單一計算單元來致使這樣的運算。在一個實施例中，混合資料類型指令被啟用以允許單一指令多執

行緒運算接受混合資料類型和/或混合精確度運算元。

【0186】圖14是根據一個實施例的多處理器單元1400的方塊圖。多處理器單元1400可以是圖2D的圖形多處理器234的變體。多處理器單元1400包含提取和解碼單元1402、分支單元1404、暫存器檔案1406、執行緒管理器1406、單一指令多執行緒單元(SIMT單元1410)以及電壓和頻率管理器1420。提取和解碼單元1402可以提取指令以供多處理器單元1400執行。分支單元1404可以基於執行的跳轉指令來計算指令指標調整。暫存器檔案1406可以儲存由SIMT單元1410使用的通用和架構暫存器。執行緒管理器1406可以在SIMT單元1410的計算單元之間分配和重新分配執行緒。在一個實施例中，SIMT單元1410被配置以執行單一指令為多個執行緒，其中指令的每個執行緒由獨立的計算單元執行。在一個實施例中，計算單元1411到計算單元1418中的每一個都包含整數ALU(例如，ALU 1411A-1418A)和浮點單元(例如FPU 1411B-1418B)。SIMT單元1410內的每個計算單元1411-1418的電壓和頻率可以由電壓和頻率管理器1420來動態地管理，當計算單位的組件被啟用和禁用時，該電壓和頻率管理器1420可以增加或減少提供給各個計算單元的電壓和時脈頻率。

【0187】在一些先前實現的配置中，每個計算單元可以執行整數指令或浮點指令的單一執行緒。如果ALU 1411A-1418A中的任何一個負責執行整數指令的執行緒，則各個FPU 1411B-FPU 1418B不能用於執行浮點指令的執

行緒，並且可能在對應的ALU 1411A-ALU 1418A的運算期間被電源閘控。例如，當ALU 1411A可以執行整數指令的執行緒，而FPU 1413B執行浮點指令的執行緒，則當ALU 1411A處於活動狀態時，FPU 1411B被電源閘控。這裡描述的實施例藉由例如使得ALU 1411A執行指令的執行緒而ALU 1411B執行不同指令的執行緒來克服這些限制。此外，一個實施例提供了對混合精確度或混合資料類型運算元的支援，使得單一指令可以同時執行具有浮點和整數運算元的指令和/或具有不同精確度的運算元之運算。

【0188】本文描述的實施例能夠藉由使每個計算單元內的所有邏輯單元可用於執行計算來使得計算單元的集群的運算吞吐量增加。在這種實施例中，被設計為選擇性地在多個精確度或多個資料類型之一執行計算的計算單元內的邏輯單元可以被配置成針對計算單元支援的每種精確度或資料類型來執行多個同時運算。對於給定的計算單元1411-1418，ALU 1411A-1418A可以執行整數運算，而FPU 1411B-1418B執行浮點運算。這些運算可以針對單一指令或多個指令來執行。在一個實施例中，啟用了新類型的混合精確度指令，其中一或多個運算元為一種資料類型或精確度的，而一或多個不同運算元為不同的資料類型或精確度的。例如，指令可以接受兩個或多個包含浮點和整數資料類型的多元件運算元，並且可以根據每個資料類型或每個精確度來執行單一指令。

【0189】圖15顯示根據一個實施例的混合精確度處理

系統1500。混合精確度處理系統1500包含含有FPU 1508A和ALU 1508B的計算單元1509。在此描述的實施例中，計算單元1509可以執行混合精確度/混合資料類型指令。例如並且在一個實施例中，指令可以被處理以在包含多個資料元件的多個運算元上執行單一運算或多個融合運算。在一個實施例中，運算元內的資料元件可以是混合精確度或混合資料類型元件。例如，第一輸入暫存器1501可以儲存包含浮點元件1502A和包含整數元件1504A和整數元件1506A的多個整數元件的第一運算元。第二輸入暫存器1503可以儲存包含浮點元件1502B和包含整數元件1504B和整數元件1506B的多個整數元件的第二運算元。第三輸入暫存器1505可以儲存包含浮點元件1502C和包含整數元件1504C和整數元件1506C的多個整數元件的第三運算元。

【0190】所述元件可被用作輸入以執行由單一運算碼(例如，運算碼1510)定義的單一運算。例如，運算碼1510可以指定多元件融合的乘加運算，其中FPU 1508A將浮點元件相乘並相加。FPU 1508A可以將浮點元件(FP元件1502A和FP元件1502B)相乘，並將乘法的乘積與第三浮點元件(FP元件1502C)相加。並行地，ALU 1508B可以執行雙整數融合的乘加運算，其中第一組整數元件(INT元件1504A和INT元件1504B)和第二組整數元件(INT元件1506A和INT元件1506B)被相乘，並且每個乘法的乘積被加到第三整數元件(INT元件1504C和INT元件1506C)。獨立的狀

態旗標 1512A-1512C 可以基於每個運算來設置。獨立的狀態旗標 1512A-1512C 可以被設置來指示本領域中已知的狀態輸出，包含但不限於進位、負數、零和溢位。在一個實施例中，獨立的狀態旗標 1512A-1512C 被輸出為狀態向量，其中狀態向量的每個元件與每個運算相關。可以生成多個結果(例如，結果 1522A、結果 1522B、結果 1522C)，其中第一結果 1522A 是浮點結果，而第二和第三結果 1522B-1522C 是整數結果。

【0191】在一個實施例中，元件的精確度也可被混合。例如並且在一個實施例中，輸入暫存器 1501、輸入暫存器 1503 和輸入暫存器 1505 是 32 位元暫存器。FP 元件 1502A-1502C 可以是 16 位元浮點元件(例如，FP16)，而 INT 元件 1504A-1504C 和 INT 元件 1506A-1506C 可以各自是 8 位元整數元件(例如，INT8)。輸出暫存器 1520 也可以是 32 位元暫存器，其中結果 1522A 是 16 位元，而結果 1522B 和結果 1522C 每個都是 8 位元。在各種實施例中，可以使用不同的暫存器大小，包含 64 位元、128 位元、256 位元和 512 位元暫存器，使得輸入元件的範圍在 8 位元和 64 位元之間。

【0192】混合精確度處理系統 1500 的運算邏輯 1700 在圖 17 中顯示，並且可以經由圖 14 的多處理器單元 1400 來實現。額外地參照圖 14，多處理器單元 1400 的提取和解碼單元 1402 可以提取和解碼包含多個運算元的單一指令，所述多個運算元參照具有不同精確度的多個資料元件，如圖 17 的方塊 1702 所示。如方塊 1704 所示，執行緒管理器 1406 可

以分派單一指令的多個執行緒以在 GPGPU 的計算單元(例如，圖 15 中的計算單元 1509)內執行。如方塊 1706 所示，並行地，計算單元可以經由計算單元內的第一邏輯單元來執行具有第一精確度的第一組運算元上的指令運算，並如方塊 1708 所示，經由計算單元內的第二邏輯單元來執行具有第二精確度的第二組運算元上的指令運算。如方塊 1710 所示，邏輯單元可以輸出運算的多個結果。不同精確度的運算元可以是具有不同精確度(例如 8 位元、16 位元、32 位元等)的相同資料類型的(例如，浮點、固定點、整數)、不同資料類型的相同精確度運算元(例如，FP-16 和 INT-16)或不同精確度的不同資料類型(例如，雙 INT-8 和 FP16)。當執行混合精確度或混合資料類型的神經網路的處理運算時，這種指令可能特別有用，其中層的輸入資料為與施加到輸入資料的權重不同的精確度或資料類型。

【0193】雖然始終在圖 15 中描述整數 ALU 150B，實施例不特別限於使用只有整數的算術邏輯單元。在一個實施例中，這裡描述的整數 ALU 可以是被配置以執行整數運算的浮點單元。

【0194】圖 16 顯示了根據一個實施例的額外的混合精確度處理系統 1600。所顯示的混合精確度處理系統 1600 經配置以致使跨多個計算單元(例如，計算單元 1607 和計算單元 1609)來並行執行浮點工作負載(例如，FP 工作負載 1602)和整數工作負載(例如，INT 工作負載 1604)。計算單元 1607 和計算單元 1609 均包含一組浮點單元(FPU 1606A、

FPU 1608A)和一組整數算術邏輯單元(例如, ALU 1606B、ALU 1608B)。在先前的實現中,例如FP工作負載1602的執行緒將例如在計算單元1607的FPU 1606A上執行,而INT工作負載1604的執行緒將在計算單元1609的ALU 1608B上並行執行。這裡描述的實施例致使計算單元的FPU和ALU並行運算。

【0195】圖18中顯示用於混合精確度處理系統1600的運算邏輯1800。如方塊1802所示,在一個實施例中,運算邏輯1800可以提取並解碼將要跨GPGPU內的多個執行緒執行的整數指令。如在方塊1804所示,邏輯1800還可以提取和解碼將要跨GPGPU內的多個執行緒執行的浮點指令。在方塊1805處,運算邏輯1800可以致使整數和浮點指令的並行執行,和經由第一計算單元和第二計算單元的整數單元來執行整數指令,而如在方塊1806處,經由第一計算單元和第二計算單元的浮點單元來執行浮點指令。在方塊1807處,邏輯1800可以致使整數指令輸出整數結果,而在方塊1808處,致使浮點運算輸出浮點結果。

用於神經網路的專用推理系統

【0196】如果用於執行推理運算的並行處理器或GPGPU係專門用於在神經網路的推理期間執行的計算的類型,則執行推理運算可以在部署機器學習系統中變得更有效。在一個實施例中,專用推理邏輯可以使用記憶體中的壓縮和/或編碼權重資料來執行計算。可以基於針對神經

網路生成的側寫來啟用自適應編碼。可以基於確定出現在權重值內的一組常用的權重值或常用模式來生成該側寫。用於常見權重值的位元組精簡編碼可以儲存在記憶體中，從而降低電力需求，或允許將較大的網路儲存在記憶體中。

【0197】圖19顯示了根據一個實施例的機器學習系統1900。在一個實施例中，機器學習系統1900是用於神經網路的專用推理系統，其支援神經網路的權重資料的編碼。經編碼的權重資料可以致使用於部署的神經網路的權重資訊的精簡尺寸表示。權重資料的精簡尺寸可以致使省電的推理或者可以致使給定記憶體尺寸的更大的神經網路的處理。

【0198】在一個實施例中，機器學習系統1900開始於可以由訓練系統1903進行處理的未訓練神經網路1902。訓練系統1903可以生成經訓練神經網路1904。權重資料側寫系統1905可以被用於側寫經訓練神經網路1904的權重資料。權重資料側寫系統1905可以生成頻率編碼權重資料1906以及用於生成頻率編碼權重資料1906的編碼側寫1910。頻率編碼權重資料1906和編碼側寫1910可以儲存在GPGPU記憶體1908中。

【0199】儲存在GPGPU記憶體1908中的頻率編碼權重資料1906和編碼側寫1910可以用來在GPGPU計算單元1914上執行神經網路層計算。在一個實施例中，編碼側寫1910可以從GPGPU記憶體1908中讀取並且用於配置GPGPU權

重解碼器 1912。GPGPU 權重解碼器 1912 可以解碼頻率編碼權重資料 1906 以向 GPGPU 計算單元 1914 提供經解碼權重資料 1913。神經網路層的輸入資料 1911 也可以是從 GPGPU 記憶體 1908 讀取。輸入資料 1911 和經解碼權重資料 1913 可以由 GPGPU 計算單元 1914 處理以生成 GPGPU 計算結果 1915。

【0200】在一個實施例中，GPGPU 計算單元 1914 可以被配置以包含 GPGPU 權重解碼器 1912，以使編碼側寫 1910 和頻率編碼權重資料 1906 可以與輸入資料 1911 被直接提供給計算單元 1914 以生成 GPGPU 計算結果 1915。

【0201】機器學習系統 1900 的邏輯運算 2000 係藉由圖 20 的流程圖來顯示。在一個實施例中，如方塊 2002 所示，邏輯運算 2000 可以配置 GPGPU 來側寫經訓練神經網路的權重以產生神經網路的權重資料的權重側寫。如方塊 2004 所示，邏輯 2000 接著可以使 GPGPU 使用權重側寫來編碼神經網路的權重。如方塊 2006 所示，邏輯 2000 接著可以使 GPGPU 將經編碼的權重和權重側寫儲存到 GPGPU 記憶體。如方塊 2008 所示，當執行神經網路的計算時，邏輯運算 2000 可以使 GPGPU 在神經網路處理期間從 GPGPU 記憶體讀取經編碼的權重。如方塊 2010 所示，在一個實施例中，邏輯運算 2000 可以在 GPGPU 在方塊 2012 處執行神經網路的計算之前，使 GPGPU 基於權重側寫來解碼經編碼的權重。如圖 19 所示，可以使用 GPGPU 權重解碼器 1912 來經編碼的權重進行解碼。當 GPGPU 被配置成直接從 GPGPU 記

憶體接受經編碼的權重時，可以跳過解碼程序。在這種實施例中，可以配置邏輯運算2000，使得GPGPU將在方塊2012處執行神經網路的計算，而不預先解碼權重資料。

額外的範例圖形處理系統

【0202】上述實施例的細節可被併入以下所述之圖形處理系統及裝置中。圖21-34之圖形處理系統及裝置係顯示可實現上述任何及所有技術的替代系統及圖形處理硬體。

額外的範例圖形處理系統概述

【0203】圖21為根據實施例的處理系統2100的方塊圖。在各個實施例中，系統2100包括一或多個處理器2102及一或多個圖形處理器2108，且可以是單處理器桌上型系統、多處理器工作站系統或具有大量處理器2102或處理器核心2107之伺服器系統。在一個實施例中，系統2100為一種結合在系統單晶片(SoC)積體電路內以供用於行動、手持式或嵌入式裝置的處理平台。

【0204】系統2100之實施例可包括或被結合入伺服器為基的遊戲平台、遊戲控制台，包括遊戲和媒體控制台、行動遊戲控制台、手持式遊戲控制台或線上遊戲控制台。在一些實施例中，系統2100為行動電話、智慧型手機、平板計算裝置或行動網際網路裝置。資料處理系統2100還可包括、耦接或整合在穿戴式裝置，諸如智慧錶穿戴式裝

置、智慧眼鏡裝置、擴增實境裝置或虛擬實境裝置。在一些實施例中，資料處理系統2100為電視或機上盒裝置，其具有一或多個處理器2102及由一或多個圖形處理器2108所產生的圖形介面。

【0205】在一些實施例中，一或多個處理器2102各包括用以處理指令的一或多個處理器核心2107，當所述指令被執行時，執行針對系統及使用者軟體之操作。在一些實施例中，一或多個處理器核心2107之各者被組態成處理特定指令集2109。在一些實施例中，指令集2109可協助複雜指令集計算(CISC)、精簡指令集計算(RISC)或經由超長指令字組(VLIW)之計算。多處理器核心2107可各處理不同的指令集2109，其可包括用以協助其它指令集之仿真的指令。處理器核心2107還可包括其它處理裝置，諸如數位信號處理器(DSP)。

【0206】在一些實施例中，處理器2102包括快取記憶體2104。根據該架構，處理器2102可具有單一內部快取或者多階內部快取。在一些實施例中，快取記憶體被共用於處理器2102的各個組件之間。在一些實施例中，處理器2102亦使用外部快取(例如，第3階(L3)快取或最末階快取(LLC))(未顯示)，其可使用已知的快取同調性技術而被共用於處理器核心2107之間。暫存器檔案2106被額外地包括在處理器2102中，其可包括不同類型的暫存器，用以儲存不同類型的資料(例如，整數暫存器、浮點暫存器、狀態暫存器及指令指標暫存器)。一些暫存器可以是通用暫存

器，而其它暫存器可特別針對處理器2102之設計。

【0207】在一些實施例中，處理器2102與處理器匯流排2110耦接，用以在處理器2102與系統2100中的其它組件之間傳輸通訊信號(諸如位址、資料或控制信號)。在一個實施例中，系統2100係使用包括記憶體控制器集線器2116及輸入輸出(I/O)控制器集線器2130之範例「集線器」系統架構。記憶體控制器集線器2116係促成記憶體裝置與系統2100的其它組件之間的通訊，而I/O控制器集線器(ICH)2130經由本地I/O匯流排來提供連接至I/O裝置。在一個實施例中，記憶體控制器集線器2116的邏輯被整合在處理器內。

【0208】記憶體裝置2120可以是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用以作用為程序記憶體之適當效能的一些其它記憶體裝置。在一個實施例中，記憶體裝置2120可操作為用於系統2100之系統記憶體，用以儲存資料2122及指令2121，以便在當一或多個處理器2102執行應用程式或程序時使用。記憶體控制器集線器2116亦與選擇性的外部圖形處理器2112耦接，該外部圖形處理器2112可與處理器2102中的一或多個圖形處理器2108通訊以執行圖形及媒體操作。

【0209】在一些實施例中，ICH 2130係致使周邊裝置經由高速I/O匯流排來連接至記憶體裝置2120及處理器2102。I/O周邊裝置包括但不限於音頻控制器2146、韌體

介面2128、無線收發器2126(例如，Wi-Fi、藍芽)、資料儲存裝置2124(例如，硬碟驅動、快閃記憶體，等等)及傳統I/O控制器2140，用以將傳統(例如，個人系統2(PS/2))裝置耦接至系統。一或多個通用串列匯流排(USB)控制器2142係連接輸入裝置，諸如鍵盤及滑鼠2144組合。網路控制器2134還可與ICH 2130耦接。在一些實施例中，高效能網路控制器(未顯示)係與處理器匯流排2110耦接。將理解，所示之系統2100為範例性而非限制性的，因為被不同地組態的其它類型的資料處理系統也可被使用。例如，I/O控制器集線器2130可被整合於一或多個處理器2102內，或者記憶體控制器集線器2116及I/O控制器集線器2130可被整合到個別的外部圖形處理器內，諸如外部圖形處理器2112。

【0210】圖22為處理器2200之實施例的方塊圖，處理器2200具有一或多個處理器核心2202A-2202N、積體記憶體控制器2214和積體圖形處理器2208。具有如本文中任何其它圖示之元件的相同參考數字(或名稱)之圖22的那些元件可用類似於本文中其它處所述的任何方式來操作或作用，但不限在此。處理器2200可包括額外核心高達(並包括)由虛線方盒所表示的額外核心2202N。處理器核心2202A-2202N之各者包括一或多個內部快取單元2204A-2204N。在一些實施例中，各處理器核心亦得以存取一或多個共用快取單元2206。

【0211】內部快取單元2204A-2204N及共用快取單元

2206係代表處理器2200內的快取記憶體階層。快取記憶體階層在各處理器核心內可包括至少一階的指令和資料快取，以及一或多階的共用中階快取，諸如第2階(L2)、第3階(L3)、第4階(L4)或其它階快取，其中在外部記憶體前的最高階快取被歸類為LLC。在一些實施例中，快取同調性邏輯係維持介在各個快取單元2206及2204A-2204N之間的同調性。

【0212】在一些實施例中，處理器2200還可包括一組一或多個匯流排控制器單元2216及系統代理核心2210。一或多個匯流排控制器單元2216管理一組周邊裝置匯流排，諸如一或多個周邊裝置組件互連匯流排(例如，PCI、PCI Express)。系統代理核心2210針對各個處理器組件提供管理功能。在一些實施例中，系統代理核心2210包括一或多個積體記憶體控制器2214，用以管理對各種外部記憶體裝置(未顯示)之存取。

【0213】在一些實施例中，一或多個處理器核心2202A-2202N包括支援同時多執行緒。在此實施例中，系統代理核心2210包括用以在多執行緒處理期間協調並操作核心2202A-2202N之組件。系統代理核心2210可額外地包括電力控制單元(PCU)，其包括用以調節處理器核心2202A-2202N及圖形處理器2208之電力狀態的邏輯和組件。

【0214】在一些實施例中，處理器2200額外地包括用以執行圖形處理操作之圖形處理器2208。在一些實施例

中，圖形處理器2208與該組共用快取單元2206及系統代理核心2210，包括一或多個積體記憶體控制器2214耦接。在一些實施例中，顯示控制器2211係與圖形處理器2208耦接以驅動圖形處理器輸出至一或多個耦接的顯示器。在一些實施例中，顯示控制器2211可以是個別的模組，其經由至少一互連而與圖形處理器耦接，或者可被整合於圖形處理器2208或系統代理核心2210內。

【0215】在一些實施例中，環狀互連單元2212被用來耦接處理器2200之內部組件。然而，可使用替代的互連單元，諸如點對點互連、切換式互連或其它技術，包括本領域中眾所周知的技術。在一些實施例中，圖形處理器2208係經由I/O鏈路2213而與環狀互連2212耦接。

【0216】範例I/O鏈路2213代表多種I/O互連中的至少一者，包括封裝上I/O互連，其協助各個處理器組件與高效能嵌入式記憶體模組2218(諸如eDRAM模組)之間的通訊。在一些實施例中，處理器核心2202A-2202N及圖形處理器2208之各者係使用嵌入式記憶體模組2218為共用的最末階快取。

【0217】在一些實施例中，處理器核心2202A-2202N為執行相同指令集架構之同質核心。在另一實施例中，處理器核心2202A-2202N就指令集架構(ISA)而言為異質的，其中處理器核心2202A-2202N中的一或多者係執行第一指令集，而其它核心中的至少一者係執行該第一指令集之子集或不同的指令集。在一個實施例中，處理器核心2202A-

2202N就微架構而言為異質的，其中具有相對較高功率消耗中的一或多個核心係與具有較低功率消耗中的一或多個電力核心耦接。此外，處理器2200可被實現於一或多個晶片上，或者作為具有除了其他組件之外的所示的組件之SoC積體電路。

【0218】圖23為圖形處理器2300的方塊圖，圖形處理器2300可以是獨立的圖形處理單元或者可以是與複數處理核心整合的圖形處理器。在一些實施例中，圖形處理器係經由記憶體映射的I/O介面而通訊至圖形處理器上之暫存器，並與置入處理器記憶體內的命令通訊。在一些實施例中，圖形處理器2300包括用以存取記憶體之記憶體介面2314。記憶體介面2314可以是對於本地記憶體、一或多個內部快取、一或多個共用外部快取和/或對於系統記憶體之介面。

【0219】在一些實施例中，圖形處理器2300還包括用以驅動顯示輸出資料至顯示裝置2320之顯示控制器2302。顯示控制器2302包括用於多層視頻或使用界面元件的顯示及組成的一或多個重疊平面之硬體。在一些實施例中，圖形處理器2300包括視頻編碼解碼引擎2306，用以將媒體編碼、解碼或轉碼至、自或介於一或多個媒體編碼格式之間，包括但不限於動態圖像專家群組(MPEG)格式，諸如MPEG-2、先進視頻編碼(AVC)格式，諸如H.264/MPEG-4 AVC，以及電影電視工程師協會(SMPTE)421M/VC-1及聯合圖像專家群組(JPEG)格式，諸如JPEG和動態JPEG

(MJPEG)格式。

【0220】在一些實施例中，圖形處理器2300包括區塊影像轉移(BLIT)引擎2304，用以執行二維(2D)柵格化器操作，包括(例如)位元邊界區塊轉移。然而，在一個實施例中，2D圖形操作係使用圖形處理引擎(GPE)2310中的一或多個組件來執行。在一些實施例中，GPE 2310為用以執行包括三維(3D)圖形操作及媒體操作之圖形操作的計算引擎。

【0221】在一些實施例中，GPE 310包括用以執行諸如使用作用於3D基元形狀(例如，矩形、三角形，等等)上之處理功能以渲染三維影像及場景之3D操作的3D管線2312。3D管線2312包括可編程及固定功能元件，其執行該元件內的各種工作和/或生產執行緒至3D/媒體子系統2315。雖然3D管線2312可被用來執行媒體操作，但GPE 2310之實施例還包括媒體管線2316，其被明確地用來執行媒體操作，諸如視頻後製處理及影像強化。

【0222】在一些實施例中，媒體管線2316包括固定功能或可編程邏輯單元，用以執行一或多個特殊化媒體操作，諸如視頻解碼加速、視頻去交錯及視頻編碼加速，以取代或代替視頻編碼解碼引擎2306。在一些實施例中，媒體管線2316額外地包括執行緒生產單元，用以生產執行緒以供執行於3D/媒體子系統2315上。所生產的執行緒係執行針對3D/媒體子系統2315中所包括的一或多個圖形執行單元上的媒體操作之計算。

【0223】在一些實施例中，3D/媒體子系統2315包括用以執行由3D管線2312及媒體管線2316所生產的執行緒之邏輯。在一個實施例中，該些管線係傳送執行緒執行請求至3D/媒體子系統2315，其包括執行緒調度邏輯，用以將各個請求仲裁並調度至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2315包括用於執行緒指令及資料中的一或多個內部快取。在一些實施例中，子系統還包括共用記憶體，包括暫存器及可定址記憶體，用以共用執行緒之間的資料並儲存輸出資料。

額外的範例圖形處理引擎

【0224】圖24為根據一些實施例的圖形處理器之圖形處理引擎2410的方塊圖。在一個實施例中，圖形處理引擎(GPE)2410為圖23中所示之GPE 2310的一種版本。具有如本文中任何其它圖示之元件的相同參考數字(或名稱)之圖24的元件可用類似於本文中其它處所述的任何方式來操作或作用，但不限在此。例如，圖23之3D管線2312及媒體管線2316被顯示。媒體管線2316在GPE 2410的一些實施例中是選擇性的，且可能不被明確地包括在GPE 2410內。例如以及在至少一個實施例中，個別的媒體和/或影像處理器被耦接至GPE 2410。

【0225】在一些實施例中，GPE 2410係與(或包括)命令串流器2403耦接，命令串流器2403提供命令串流至3D管

線 2312 和 / 或媒體管線 2316。在一些實施例中，命令串流器 2403 係與記憶體耦接，該記憶體可以是系統記憶體或內部快取記憶體及共用快取記憶體中的一或多者。在一些實施例中，命令串流器 2403 從記憶體接收命令並將該些命令傳送至 3D 管線 2312 和 / 或媒體管線 2316。該些命令被直接提取自環狀緩衝器，其儲存 3D 管線 2312 及媒體管線 2316 之命令。在一個實施例中，環狀緩衝器可額外地包括批次命令緩衝器，其儲存多個命令之批次。3D 管線 2312 之命令還可包括對於記憶體中所儲存之資料的參考，諸如但不限於用於 3D 管線 2312 之頂點和幾何資料和 / 或用於媒體管線 2316 之影像資料和記憶體物件。3D 管線 2312 及媒體管線 2316 係藉由經由個別管線內的邏輯以執行操作或將一或多個執行緒調度至圖形核心陣列 2414 來處理該些命令及資料。

【0226】在各個實施例中，3D 管線 2312 可執行一或多個著色器程式，諸如頂點著色器、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，藉由處理該些指令並將執行緒調度至圖形核心陣列 2414。圖形核心陣列 2414 提供了執行資源的統一區塊。圖形核心陣列 2414 內的多用途執行邏輯(例如，執行單元)包括支援各種 3D API 著色器語言並可執行與多個著色器相關的多個同時執行緒。

【0227】在一些實施例中，圖形核心陣列 2414 還包括用以執行諸如視頻和 / 或影像處理之媒體功能的執行邏

輯。在一個實施例中，執行單元額外地包括通用邏輯，其可編程以執行除了圖形處理操作之外的平行通用計算操作。通用邏輯可平行地或聯合圖16之處理器核心1607或圖22中之核心2202A-2202N內的通用邏輯來執行處理操作。

【0228】由圖形核心陣列2414上所執行之執行緒所產生的輸出資料可將資料輸出至統一返回緩衝器(URB)2418中之記憶體。URB 2418可儲存多個執行緒之資料。在一些實施例中，URB 2418可被用來傳送資料於圖形核心陣列2414上所執行的執行緒之間。在一些實施例中，URB 2418可額外地被用於圖形核心陣列上的執行緒與共用功能邏輯2420內的固定功能邏輯之間的同步化。

【0229】在一些實施例中，圖形核心陣列2414為可縮放的，以致使該陣列包括可變數目的圖形核心，其根據GPE 2410之目標功率及效能位準而各具有可變數目的執行單元。在一個實施例中，執行資源為動態可縮放的，以致使執行資源可根據所需來啟用或禁用。

【0230】圖形核心陣列2414係與共用功能邏輯2420耦接，共用功能邏輯2420包括被共用於圖形核心陣列中的圖形核心之間的多個資源。共用功能邏輯2420內的共用功能為硬體邏輯單元，其提供特殊化補充功能給圖形核心陣列2414。在各個實施例中，共用功能邏輯2420包括但不限於取樣器2421、數學2422及執行緒間通訊(ITC)2423邏輯。此外，一些實施例係實現共用功能邏輯2420內的一或多個快取2425。共用功能被實現，其中針對既定特殊化功能之

需求對於包括在圖形核心陣列2414內仍不足。取而代之地，該特殊化功能之單一例示被實現為共用功能邏輯2420中之獨立單體且被共用於圖形核心陣列2414內的執行資源之間。被共用於圖形核心陣列2414之間且被包括在圖形核心陣列2414內的功能的精確集合係在實施例之間改變。

【0231】圖25為圖形處理器2500之另一實施例的方塊圖。具有如本文中任何其它圖示之元件的相同參考數字(或名稱)之圖25的元件可用類似於本文中其它處所述的任何方式來操作或作用，但不限在此。

【0232】在一些實施例中，圖形處理器2500包括環狀互連2502、管線前端2504、媒體引擎2537及圖形核心2580A-2580N。在一些實施例中，環狀互連2502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者一或多個通用處理器核心。在一些實施例中，圖形處理器為整合於多核心處理系統內的許多處理器之一者。

【0233】在一些實施例中，圖形處理器2500經由環狀互連2502來接收命令之批次。進來的命令係由管線前端2504中之命令串流器2503來解讀。在一些實施例中，圖形處理器2500包括用以經由圖形核心2580A-2580N來執行3D幾何處理及媒體處理的可縮放執行邏輯。針對3D幾何處理命令，命令串流器2503將命令供應至幾何管線2536。針對至少一些媒體處理命令，命令串流器2503將該些命令供應至與媒體引擎2537耦接的視頻前端2534。在一些實施例中，媒體引擎2537包括用於視頻和影像後製處理的視頻品

質引擎(VQE)2530及用以提供硬體加速的媒體資料編碼和解碼的多格式編碼/解碼(MFX)2533引擎。在一些實施例中，幾何管線2536及媒體引擎2537各針對由至少一圖形核心2580A所提供的執行緒執行資源來產生執行緒。

【0234】在一些實施例中，圖形處理器2500包括可縮放執行緒執行資源特徵模組式核心2580A-2580N(有時稱為核心片)，其各具有多個子核心2550A-550N、2560A-2560N(有時稱為核心子片)。在一些實施例中，圖形處理器2500可具有任何數目的圖形核心2580A至2580N。在一些實施例中，圖形處理器2500包括至少具有第一子核心2550A及第二子核心2560A的圖形核心2580A。在其它實施例中，圖形處理器為具有單一子核心(例如，2550A)之低功率處理器。在一些實施例中，圖形處理器2500包括多個圖形核心2580A-2580N，其各包括一組第一子核心2550A-2550N及一組第二子核心2560A-2560N。該組第一子核心2550A-2550N中之各子核心至少包括第一組執行單元2552A-2552N及媒體/紋理取樣器2554A-2554N。該組第二子核心2560A-2560N中之各子核心至少包括第二組執行單元2562A-2562N及取樣器2564A-2564N。在一些實施例中，各子核心2550A-2550N、2560A-2560N係共用一組共用資源2570A-2570N。在一些實施例中，共用資源包括共用快取記憶體及像素操作邏輯。其它共用資源還可被包括在圖形處理器之各個實施例中。

額外的範例執行單元

【0235】圖26顯示包括GPE的一些實施例中所利用的處理元件之陣列的執行緒執行邏輯2600。具有如本文中任何其它圖示之元件的相同參考數字(或名稱)之圖26的元件可用類似於本文中其它處所述的任何方式來操作或作用，但不限在此。

【0236】在一些實施例中，執行緒執行邏輯2600包括著色器處理器2602、執行緒調度器2604、指令快取2606、包括複數個執行單元2608A-2608N的可縮放執行單元陣列、取樣器2610、資料快取2612及資料埠2614。在一個實施例中，可縮放執行單元陣列可藉由根據工作負載之計算需求以啟用或禁用一或多個執行單元(例如，執行單元2608A、2608B、2608C、2608D、至2608N-1及2608N)來動態地縮放。在一個實施例中，所包括的組件係經由鏈接至該些組件之各者的互連構造來互連。在一些實施例中，執行緒執行邏輯2600包括對於記憶體的一或多個連接，該記憶體諸如系統記憶體或快取記憶體，透過一或多個指令快取2606、資料埠2614、取樣器2610及執行單元2608A-2608N。在一些實施例中，各執行單元(例如，2608A)為獨立可編程通用計算單元，其能夠執行多個同步硬體執行緒，而同時針對各執行緒平行地處理多個資料元件。在各種實施例中，執行單元2608A-2608N之陣列為可縮放的，以包括任何數目的個別執行單元。

【0237】在一些實施例中，執行單元2608A-2608N主

要被用來執行著色器程式。著色器處理器2602可處理各種著色器程式，並經由執行緒調度器2604來調度與該些著色器程式相關的執行緒。在一個實施例中，執行緒調度器包括用以仲裁來自圖形和媒體管線之執行緒起始請求，並將該些請求的執行緒實例化於執行單元2608A-2608N中的一或多個執行單元上之邏輯。例如，幾何管線(例如，圖25之2536)可將頂點、鑲嵌或幾何著色器調度至執行緒執行邏輯2600(圖26)以供處理。在一些實施例中，執行緒調度器2604還可處理來自執行中著色器程式的運行時間執行緒生產請求。

【0238】 在一些實施例中，執行單元2608A-2608N支援包括對於許多標準3D圖形著色器指令之本機支援的指令集，以致使來自圖形庫(例如，Direct 3D及OpenGL)之著色器程式以最少轉換被執行。執行單元支援頂點和幾何處理(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)及通用處理(例如，計算和媒體著色器)。執行單元2608A-2608N之各者能夠多重發送單指令多資料(SIMD)執行，而多執行緒操作係致使在面對較高延遲時間記憶體存取時之有效率的執行環境。各執行單元內的各硬體執行緒具有專用的高頻寬暫存器檔案及相關的獨立執行緒狀態。執行係每個時脈多重發送至管線，其得以進行整數、單和雙精確度浮點操作、SIMD分支能力、邏輯操作、超越操作及其它各種操作。當等待來自記憶體之資料或共用功能之一時，執行單元2608A-

2608N內的相依性邏輯係致使等待執行緒休眠，直到該請求的資料已被返回。當該等待執行緒正在休眠時，硬體資源可被用於處理其它執行緒。例如，在與頂點著色器操作相關的延遲期間，執行單元可執行用於像素著色器、片段著色器或其它類型的著色器程式，包括不同的頂點著色器之操作。

【0239】執行單元2608A-2608N中之各執行單元係操作於資料元件之陣列上。資料元件的數目為「執行大小」或針對該指令之通道數目。執行通道為針對指令內的資料元件存取、遮蔽及流程控制的執行之邏輯單元。通道數目可獨立於針對特定圖形處理器之實體算術邏輯單元(ALU)或浮點單元(FPU)的數目。在一些實施例中，執行單元2608A-2608N支援整數及浮點資料類型。

【0240】執行單元指令集包括SIMD指令。各個資料元件可被儲存為暫存器中之緊縮資料類型，且執行單元將根據該些元件之資料大小來處理各個元件。例如，當操作於256位元寬的向量時，該向量的256位元被儲存於暫存器中，且執行單元係操作於該向量上而成為四個個別的64位元緊縮資料元件(四字組(QW)大小資料元件)、八個個別的32位元緊縮資料元件(雙字組(DW)大小資料元件)、十六個個別的16位元緊縮資料元件(字組(W)大小資料元件)或三十二個個別的8位元緊縮資料元件(位元組(B)大小資料元件)。然而，不同的向量寬度及暫存器大小是可能的。

【0241】一或多個內部指令快取(例如，2606)被包括

在執行緒執行邏輯2600中，以將執行單元之執行緒指令快取。在一些實施例中，一或多個資料快取(例如，2612)被包括以在執行緒執行期間將執行緒資料快取。在一些實施例中，取樣器2610被包括以提供針對3D操作之紋理取樣及針對媒體操作之媒體取樣。在一些實施例中，取樣器2610包括特殊化紋理或媒體取樣功能，用以在提供已取樣資料至執行單元之前，處理在取樣程序期間之紋理或媒體資料。

【0242】在執行期間，圖形及媒體管線係經由執行緒生產和調度邏輯來將執行緒起始請求傳送至執行緒執行邏輯2600。一旦幾何物件之群組已被處理並柵格化為像素資料，則著色器處理器2602內的像素處理器邏輯(例如，像素著色器邏輯、片段著色器邏輯，等等)被調用以進一步計算輸出資訊並致使結果被寫入至輸出表面(例如，顏色緩衝器、深度緩衝器、模板緩衝器，等等)。在一些實施例中，像素著色器或片段著色器係計算將跨該柵格化物件被內插的各個頂點屬性之值。在一些實施例中，著色器處理器2602內的像素處理器邏輯接著執行應用程式編程介面(API)供應的像素或片段著色器程式。為了執行著色器程式，著色器處理器2602經由執行緒調度器2604來將執行緒調度至執行單元(例如，2608A)。在一些實施例中，像素著色器2602係使用取樣器2610中之紋理取樣邏輯來存取記憶體中所儲存之紋理映射中的紋理資料。紋理資料及輸入幾何資料上的算術操作係計算各幾何片段之像素顏色資料

或丟棄一或多個像素而不做進一步處理。

【0243】在一些實施例中，資料埠2614係提供記憶體存取機制給執行緒執行邏輯2600，用以將經處理資料輸出至記憶體以在圖形處理器輸出管線上處理。在一些實施例中，資料埠2614包括或耦接至一或多個快取記憶體(例如，資料快取2612)，用以經由資料埠來快取資料以供記憶體存取。

【0244】圖27為顯示根據一些實施例的圖形處理器指令格式2700的方塊圖。在一或多個實施例中，圖形處理器執行單元係支援一種具有多個格式之指令的指令集。實線方盒係顯示通常地被包括在執行單元指令中之組件，而虛線則包括為選擇性的或者僅被包括在該些指令之子集中的組件。在一些實施例中，所述且所示的指令格式2700為巨集指令，一旦該指令被處理後，其為供應至執行單元之指令，而不是得自指令解碼之微操作。

【0245】在一些實施例中，圖形處理器執行單元係本機地支援128位元指令格式2710之指令。64位元壓縮指令格式2730可根據選定的指令、指令選項及運算元的數目而用於一些指令。本機128位元指令格式710係提供對於所有指令選項之存取，而一些選項及操作被侷限於64位元格式2730。可用於64位元格式2730之本機指令隨實施例而改變。在一些實施例中，該指令係使用索引欄位2713中之一組索引值而被部分地壓縮。執行單元硬體係根據索引值而參考一組壓縮表，並使用壓縮表輸出以重新建構128位元

指令格式2710之本機指令。

【0246】針對各格式，指令運算碼2712係定義該執行單元應執行之操作。執行單元跨各運算元之多資料元件來平行地執行各指令。例如，回應於加法指令，執行單元跨代表紋理元件或圖片元件之各顏色通道來執行同步加法運算。預設地，執行單元跨運算元的所有資料通道來執行各指令。在一些實施例中，指令控制欄位2714致使對於某些執行選項之控制，諸如通道選擇(例如，斷定)及資料通道順序(例如，拌合)。針對128位元指令格式2710之指令，執行大小欄位2716係限制將被平行地執行之資料通道的數目。在一些實施例中，執行大小欄位2716不得用於64位元壓縮指令格式2730。

【0247】一些執行單元指令具有高達三運算元，其包括兩個來源運算元(src0 2720、src1 2722)及一個目的地2718。在一些實施例中，執行單元支援雙目的地指令，其中該些目的地之一者被隱含。資料操縱指令可具有第三來源運算元(例如，SRC2 2724)，其中指令運算碼2712係判定來源運算元的數目。指令的最後來源運算元可以是利用該指令傳遞的立即(例如，硬編碼)值。

【0248】在一些實施例中，128位元指令格式2710包括存取/位址模式欄位2726，其指明(例如)直接暫存器定址模式或是間接暫存器定址模式是否被使用。當直接暫存器定址模式被使用時，一或多個運算元之暫存器位址係直接地由該指令中之位元來提供。

【0249】在一些實施例中，128位元指令格式2710包括存取/位址模式欄位2726，其指明該指令之位址模式和/或存取模式。在一個實施例中，存取模式被用來定義該指令之資料存取對準。一些實施例支援存取模式，包括16位元組對準的存取模式以及1位元組對準的存取模式，其中存取模式之位元組對準係判定指令運算元之存取對準。例如，當在第一模式時，該指令對於來源和目的地運算元可使用位元組對準的定址，而當在第二模式時，該指令對於來源和目的地運算元可使用16位元組對準的定址。

【0250】在一個實施例中，存取/位址模式欄位2726之位址模式部分判定該指令是否使用直接或者間接定址。當直接暫存器定址模式被使用時，該指令中之位元係直接地提供一或多個運算元之暫存器位址。當間接暫存器定址模式被使用時，一或多個運算元之暫存器位址可根據該指令中之位址暫存器值及位址立即欄位而被計算。

【0251】在一些實施例中，指令係根據運算碼2712位元欄位而被群組以簡化運算碼解碼2740。針對8位元運算碼，位元4、5及6容許執行單元判定運算碼之類型。所示之精確運算碼群組僅為範例。在一些實施例中，移動和邏輯運算碼群組2742包括資料移動和邏輯指令(例如，移動(mov)、比較(cmp))。在一些實施例中，移動和邏輯群組2742係共用五個最高有效位元(MSB)，其中移動(mov)指令為0000xxxxb之形式，而邏輯指令為0001xxxxb之形式。流程控制指令群組2744(例如，呼叫、跳躍(jmp))包括為

0010xxxxb(例如，0x20)之形式的指令。雜項指令群組2746包括指令之混合，其包括為0011xxxxb(例如，0x30)之形式的同步化指令(例如，等待、傳送)。平行數學指令群組2748包括為0100xxxxb(例如，0x40)之形式的組件式算術指令(例如，加、乘(mul))。平行數學群組2748係跨資料通道而平行地執行算術運算。向量數學群組2750包括為0101xxxxb(例如，0x50)之形式的算術指令(例如，dp4)。向量數學群組係執行諸如對於向量運算元之內積計算之算術。

額外的範例圖形管線

【0252】圖28為圖形處理器2800之另一實施例的方塊圖。具有如本文中任何其它圖示之元件的相同參考數字(或名稱)之圖28的元件可用類似於本文中其它處所述的任何方式來操作或作用，但不限在此。

【0253】在一些實施例中，圖形處理器2800包括圖形管線2820、媒體管線2830、顯示引擎2840、執行緒執行邏輯2850及渲染輸出管線2870。在一些實施例中，圖形處理器2800為包括一或多個通用處理核心之多核心處理系統內的圖形處理器。圖形處理器係由暫存器寫入至一或多個控制暫存器(未顯示)所控制，或者經由發送至圖形處理器2800(經由環狀互連2802)之命令來控制。在一些實施例中，環狀互連2802將圖形處理器2800耦接至其它處理組件，諸如其它圖形處理器或通用處理器。來自環狀互連

2802之命令係由命令串流器2803所解讀，命令串流器2803將指令供應至圖形管線2820或媒體管線2830的個別組件。

【0254】在一些實施例中，命令串流器2803指引頂點提取器2805之操作，其從記憶體讀取頂點資料並執行由命令串流器2803所提供的頂點處理命令。在一些實施例中，頂點提取器2805提供頂點資料至頂點著色器2807，其對於每個頂點執行座標空間轉換及照亮操作。在一些實施例中，頂點提取器2805及頂點著色器2807係執行頂點處理指令，藉由經執行緒調度器2831來將執行緒調度至執行單元2852A-2852B。

【0255】在一些實施例中，執行單元2852A-2852B為具有用以執行圖形及媒體操作之指令集的向量處理器之陣列。在一些實施例中，執行單元2852A-2852B具有裝附的L1快取2851，其專用於各陣列或者共用於多個陣列之間。快取可被組態成資料快取、指令快取或單一快取，其被分割成在不同的分割中含有資料及指令。

【0256】在一些實施例中，圖形管線2820包括鑲嵌組件，用以執行3D物件之硬體加速鑲嵌。在一些實施例中，可編程殼體(hull)著色器811係組態鑲嵌操作。可編程領域著色器817提供鑲嵌輸出的後端評估。鑲嵌器2813係操作於殼體著色器2811之方向並含有特殊用途邏輯，用以根據被提供為圖形管線2820之輸入的粗略幾何模型來產生一組詳細的幾何物件。在一些實施例中，假如未使用鑲嵌，則鑲嵌組件(例如，殼體著色器2811、鑲嵌器2813及領域著

色器 2817)可被省略。

【0257】在一些實施例中，完整幾何物件可經由其被調度至執行單元 2852A-2852B 的一或多個執行緒，藉由幾何著色器 2819 來處理，或者可直接地前進至截波器 2829。在一些實施例中，幾何著色器係操作於整個幾何物件上，而非如圖形管線中的先前階段中之頂點或頂點的補丁。假如鑲嵌被禁用，則幾何著色器 2819 接收來自頂點著色器 2807 之輸入。在一些實施例中，假如鑲嵌單元被禁用，則幾何著色器 2819 可由幾何著色器程式所編程，以執行幾何鑲嵌。

【0258】在柵格化之前，截波器 2829 係處理頂點資料。截波器 2829 可以是固定功能的截波器或者具有截波及幾何著色器功能之可編程截波器。在一些實施例中，渲染輸出管線 2870 中之柵格化器及深度測試組件 2873 係調度像素著色器以將幾何物件轉換為其每個像素表示。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 2850 中。在一些實施例中，應用程式可省略柵格化器及深度測試組件 2873，並經由串流輸出單元 2823 來存取未柵格化的頂點資料。

【0259】圖形處理器 2800 具有互連匯流排、互連構造或一些其它互連機制，其容許資料及訊息傳遞於處理器的主要組件之間。在一些實施例中，執行單元 2852A-2852B 及相關快取 2851、紋理和媒體取樣器 2854 及紋理/取樣器快取 2858 係經由資料埠 2856 而互連，以執行記憶體存取並

與處理器之渲染輸出管線組件通訊。在一些實施例中，取樣器 2854、快取 2851、2858 及執行單元 2852A-2852B 各具有個別的記憶體存取路徑。

【0260】在一些實施例中，渲染輸出管線 2870 含有柵格化器及深度測試組件 2873，其將頂點為基的物件轉換為相關之像素為基的表示。在一些實施例中，柵格化器邏輯包括視窗器/遮蔽器單元，用以執行固定功能三角及直線柵格化。相關的渲染快取 2878 及深度快取 2879 還可用在一些實施例中。像素操作組件 2877 係對於資料執行像素為基的操作，雖然在一些例子中，與 2D 操作相關的像素操作(例如，利用混合之位元區塊影像轉移)係由 2D 引擎 2841 來執行或者於顯示時刻使用重疊顯示平面由顯示控制器 2843 來取代。在一些實施例中，共用 L3 快取 2875 可用於所有圖形組件，其容許資料之共用而不使用主系統記憶體。

【0261】在一些實施例中，圖形處理器媒體管線 2830 包括媒體引擎 2837 及視頻前端 2834。在一些實施例中，視頻前端 2834 接收來自命令串流器 2803 之管線命令。在一些實施例中，媒體管線 2830 包括個別的命令串流器。在一些實施例中，在傳送該命令至媒體引擎 2837 之前，視頻前端 2834 處理媒體命令。在一些實施例中，媒體引擎 2837 包括執行緒生產功能，用以生產執行緒以便經由執行緒調度器 2831 來調度至執行緒執行邏輯 2850。

【0262】在一些實施例中，圖形處理器 2800 包括顯示引擎 2840。在一些實施例中，顯示引擎 2840 位於處理器

2800外部，並經由環狀互連2802(或某些其它互連匯流排或組織)而與圖形處理器耦接。在一些實施例中，顯示引擎2840包括2D引擎2841及顯示控制器2843。在一些實施例中，顯示引擎2840含有特殊用途邏輯，其能夠獨立自3D管線而操作。在一些實施例中，顯示控制器2843與顯示裝置(未顯示)耦接，顯示裝置可以是系統整合顯示裝置(如在膝上型電腦中)或經由顯示裝置連接器而裝附的外部顯示裝置。

【0263】在一些實施例中，圖形管線2820及媒體管線2830可組態成根據多個圖形及媒體編程介面來執行操作，而非專用於任一應用程式編程介面(API)。在一些實施例中，圖形處理器之驅動程式軟體將專用於特定圖形或媒體庫的API呼叫轉換為可由圖形處理器所處理的命令。在一些實施例中，提供給開放式圖形庫(OpenGL)、開放式計算語言(OpenCL)和/或Vulkan圖形和計算API之支援，其均來自Khronos集團。在一些實施例中，還可提供支援給來自微軟公司的Direct3D庫。在一些實施例中，可支援這些庫之組合。還可提供支援給開放式來源電腦視覺庫(OpenCV)。假如可從未來API之管線執行至圖形處理器之管線的映射，則具有可相容3D管線之未來API亦將被支援。

圖形管線編程

【0264】圖29A為根據一些實施例的顯示圖形處理器

命令格式2900的方塊圖。圖29B為根據實施例的顯示圖形處理器命令序列2910的方塊圖。圖29A中之實線方盒係顯示通常被包括在圖形命令中之組件，而虛線則包括為選擇性的或者其僅被包括在圖形命令之子集中的組件。圖29A之範例圖形處理器命令格式2900包括用以識別該命令之目標客戶2902的資料欄位、命令操作碼(運算碼)2904及該命令之相關資料2906。子運算碼2905及命令大小2908亦被包括在一些命令中。

【0265】在一些實施例中，客戶2902係指明處理該命令資料之圖形裝置的客戶單元。在一些實施例中，圖形處理器命令解析器係檢查各命令之客戶欄位以調適該命令的進一步處理並將命令資料發送至適當的客戶單元。在一些實施例中，圖形處理器客戶單元包括記憶體介面單元、渲染單元、2D單元、3D單元及媒體單元。各客戶單元具有處理該些命令之相應處理管線。一旦該命令由客戶單元所接收，客戶單元便讀取運算碼2904，並且(假如存在的話)子運算碼2905用以判定應執行的操作。客戶單元係使用資料欄位2906中之資訊來執行該命令。對於一些命令，預期有明確的命令大小2908以指明命令之大小。在一些實施例中，命令解析器自動地根據命令運算碼來判定至少一些命令的大小。在一些實施例中，命令係經由多個雙字組來對準。

【0266】圖29B中之流程圖顯示了範例圖形處理器命令序列2910。在一些實施例中，特徵在於圖形處理器之實

施例的資料處理系統的軟體或韌體使用了所顯示之命令序列的版本以設定、執行及終止一組圖形操作。範例命令序列被顯示並描述以僅供範例之目的，因為實施例並不限定於這些特定命令或此命令序列。此外，該些命令可被發送為命令序列中之命令的批次，以致使圖形處理器將以至少部分並行性來處理命令之序列。

【0267】在一些實施例中，圖形處理器命令序列2910可開始於管線清除命令2912，用以致使任何活動圖形管線完成該管線的目前擱置命令。在一些實施例中，3D管線2922及媒體管線2924不會並行地操作。管線清除被執行，以致使活動圖形管線完成任何擱置的命令。回應於管線清除，圖形處理器之命令解析器將暫停命令處理，直到活動繪圖引擎完成擱置的操作且相關讀取快取被無效化。選擇性地，渲染快取中被標記為「髒」的任何資料可被清除至記憶體。在一些實施例中，管線清除命令2912可被使用於管線同步化，或者在將圖形處理器置入低功率狀態之前。

【0268】在一些實施例中，當命令序列需要圖形處理器來明確地在管線之間切換時，管線選擇命令2913被使用。在一些實施例中，在發送管線命令之前，管線選擇命令2913在執行背景內僅需要一次，除非該背景將發送命令給兩管線。在一些實施例中，需要管線清除命令2912緊接在經由管線選擇命令2913的管線切換之前。

【0269】在一些實施例中，管線控制命令2914係組態圖形管線以供操作且被用來編程3D管線2922及媒體管線

2924。在一些實施例中，管線控制命令2914係組態活動管線之管線狀態。在一個實施例中，管線控制命令2914被用於管線同步化，並用以在處理命令之批次之前清除來自活動管線內的一或多個快取記憶體之資料。

【0270】在一些實施例中，返回緩衝器狀態命令2916被用來組態一組返回緩衝器以供個別管線來寫入資料。一些管線操作需要一或多個返回緩衝器之配置、選擇或組態，其中該些操作在處理期間將中間資料寫入該些返回緩衝器。在一些實施例中，圖形處理器也使用一或多個返回緩衝器來儲存輸出資料並執行跨執行緒通訊。在一些實施例中，返回緩衝器狀態2916包括選擇返回緩衝器的大小與數目以用於一組管線操作。

【0271】命令序列中的殘留命令係根據針對操作之活動管線而不同。根據管線判定2920，命令序列被調整至以3D管線狀態2930開始的3D管線2922或以媒體管線狀態2940開始的媒體管線2924。

【0272】用以組態3D管線狀態2930之命令包括用於頂點緩衝器、頂點元件狀態、恆定顏色狀態、深度緩衝器狀態及其它狀態變數的3D狀態設定命令，其應在3D基元命令被處理之前被組態。這些命令之值係至少部分地根據使用中之特定3D API來判定。在一些實施例中，3D管線狀態2930命令亦能夠選擇性地禁用或省略一些管線元件，假如那些元件將不被使用的話。

【0273】在一些實施例中，3D基元2932命令被用來提

交3D基元以供3D管線來處理。經由3D基元2932命令而被傳遞至圖形處理器之命令及相關參數被遞送至圖形管線中之頂點提取功能。頂點提取功能係使用3D基元2932命令資料來產生頂點資料結構。頂點資料結構被儲存於一或多個返回緩衝器中。在一些實施例中，3D基元2932命令被用來經由頂點著色器來對於3D基元執行頂點操作。為了處理頂點著色器，3D管線2922將著色器執行緒調度至圖形處理器執行單元。

【0274】在一些實施例中，3D管線2922係經由執行2934命令或事件而被觸發。在一些實施例中，暫存器寫入觸發了命令執行。在一些實施例中，執行係經由命令序列中之「去(go)」或「踢(kick)」命令而被觸發。在一個實施例中，命令執行係使用管線同步化命令而被觸發以透過圖形管線來清除該命令序列。3D管線將執行針對3D基元之幾何處理。一旦操作完成，所得的幾何物件被柵格化且像素引擎將所得像素上色。用以控制像素著色及像素後端操作之額外命令還可被包括以用於那些操作。

【0275】在一些實施例中，當執行媒體操作時，圖形處理器命令序列2910係遵循媒體管線2924路徑。通常，針對媒體管線2924之編程的特定使用及方式係取決於待執行之媒體或計算操作。在媒體解碼期間，特定媒體解碼操作可被卸載至媒體管線。在一些實施例中，媒體管線也可被省略，而媒體解碼可使用由一或多個通用處理核心所提供的資源而被整體地或部分地執行。在一個實施例中，媒體

管線還包括用於通用圖形處理器單元(GPGPU)操作之元件，其中圖形處理器通常使用並非明確地相關於圖形基元之渲染的計算著色器程式來執行SIMD向量操作。

【0276】在一些實施例中，媒體管線2924被組態成以如3D管線2922的類似方式。在媒體物件命令2942之前，用以組態媒體管線狀態2940的一組命令被調度或置入命令佇列。在一些實施例中，媒體管線狀態命令2940包括用以組態將被用來處理媒體物件的媒體管線元件之資料。此包括用以組態媒體管線內的視頻解碼及視頻編碼邏輯的資料，諸如編碼或解碼格式。在一些實施例中，媒體管線狀態命令2940亦支援使用一或多個指向含有狀態設定之批次的「間接」狀態元件之指標。

【0277】在一些實施例中，媒體物件命令2942係供應指向媒體物件之指標以供媒體管線處理。媒體物件包括含有待處理之視頻資料的記憶體緩衝器。在一些實施例中，在發送媒體物件命令2942之前，所有媒體管線狀態需為有效的。一旦管線狀態被組態且媒體物件命令2942被排列，則媒體管線2924係經由執行命令2944或同等執行事件(例如，暫存器寫入)而被觸發。來自媒體管線2924之輸出可接著藉由3D管線2922或媒體管線2924所提供的操作來後處理。在一些實施例中，GPGPU操作被以如媒體操作之類似方式來組態與執行。

圖形軟體架構

【0278】圖30顯示根據一些實施例的用於資料處理系統3000的範例圖形軟體架構。在一些實施例中，軟體架構包括3D圖形應用程式3010、作業系統3020及至少一處理器3030。在一些實施例中，處理器3030包括圖形處理器3032及一或多個通用處理器核心3034。圖形應用程式3010與作業系統3020各執行於資料處理系統之系統記憶體3050中。

【0279】在一些實施例中，3D圖形應用程式3010含有包括著色器指令3012的一或多個著色器程式。著色器語言指令可以是高階著色器語言，諸如高階著色器語言(HLSL)或OpenGL著色器語言(GLSL)。應用程式還包括以適於由通用處理器核心3034執行的機器語言的可執行指令3014。應用程式還包括由頂點資料所定義的圖形物件3016。

【0280】在一些實施例中，作業系統3020是來自微軟公司的Microsoft® Windows®作業系統、專用UNIX類作業系統或使用Linux內核之變體的開放式來源UNIX類作業系統。作業系統3020可支援圖形API 3022，諸如Direct3D API、OpenGL API或Vulkan API。當使用Direct3D API時，作業系統3020係使用前端著色器編譯器3024來將HLSL中的任何著色器指令3012編譯為低階著色器語言。該編譯可以是及時(JIT)編譯，或者該應用程式可執行著色器預編譯。在一些實施例中，在3D圖形應用程式3010之編譯期間，高階著色器被編譯為低階著色器。在一些實施例中，著色器指令3012以中間形式來提供，諸如由Vulkan API所使用之標準可攜式中間表示(SPIR)的版本。

【0281】在一些實施例中，使用者模式圖形驅動程式3026含有後端著色器編譯器3027，用以將著色器指令3012轉換為硬體特定的表示。當使用OpenGL API時，以GLSL高階語言之著色器指令3012被傳遞至使用者模式圖形驅動程式3026以供編譯。在一些實施例中，使用者模式圖形驅動程式3026係使用作業系統內核模式功能3028來與內核模式圖形驅動程式3029通訊。在一些實施例中，內核模式圖形驅動程式3029係與圖形處理器3032通訊以調度命令及指令。

IP核心實現

【0282】至少一個實施例的一或多種態樣可由機器可讀取媒體上所儲存的代表性碼來實現，該代表性碼係代表和/或定義諸如處理器的積體電路內的邏輯。例如，機器可讀取媒體可包括代表處理器內的各種邏輯的指令。當由機器所讀取時，該些指令可致使該機器製造用以執行本文中所述之技術的邏輯。這樣的表示(已知為「IP核心」)為用於積體電路之邏輯的可再使用單元，其可被儲存於有形的、機器可讀取媒體上而成為描述積體電路之結構的硬體模型。硬體模型可被供應至各個消費者或製造設施，其將硬體模型載入至製造積體電路的製造機器上。積體電路可被製造以致使電路執行配合本文中所述之任何實施例而描述的操作。

【0283】圖31為顯示可被用來製造積體電路以執行根

據實施例的操作的IP核心開發系統3100之方塊圖。IP核心開發系統3100可被用來產生模組式、可再使用設計，其可被結合入更大的設計或者被用來建構完整的積體電路(例如，SOC積體電路)。設計設施3130可用高階編程語言(例如，C/C++)產生IP核心設計之軟體模擬3110。軟體模擬3110可使用模擬模型3112被用來設計、測試及驗證IP核心之行為。模擬模型3112可包括功能、行為和/或時序模擬。暫存器轉移階層(RTL)設計3115可接著從模擬模型3112被產生或合成。RTL設計3115為積體電路之行為的抽象概念，其建模了硬體暫存器之間的數位信號之流程，該些硬體暫存器包括使用所建模的數位信號而執行的相關邏輯。除了RTL設計3115之外，在邏輯階層或電晶體階層的低階設計也可被產生、設計或合成。因此，初始設計及模擬之特定細節可改變。

【0284】RTL設計3115或同等物可由設計設施進一步合成為硬體模型3120，其可以是硬體描述語言(HDL)或實體設計資料的一些其它表示。HDL可被進一步模擬或測試以驗證IP核心設計。IP核心設計可使用非揮發性記憶體3140(例如，硬碟、快閃記憶體或任何非揮發性儲存媒體)被儲存以供遞送至第三方製造設施3165。可替代地，IP核心設計可透過有線連接3150或無線連接3160而被傳輸(例如，經由網際網路)。製造設施3165可接著至少部分地基於IP核心設計來製造積體電路。所製造的積體電路可組態成根據至少本文中所述的至少一個實施例來執行操作。

範例系統單晶片積體電路

【0285】圖 32-34 顯示根據本文中所述的各個實施例之可使用一或多個 IP 核心來製造的範例積體電路及相關的圖形處理器。除了所顯示的之外，可包括其它的邏輯和電路，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。

【0286】圖 32 為顯示根據實施例的可使用一或多個 IP 核心來製造的範例系統單晶片積體電路 3200 的方塊圖。範例積體電路 3200 包括一或多個應用處理器 3205 (例如，CPU)、至少一個圖形處理器 3210，並且可額外地包括影像處理器 3215 和/或視頻處理器 3220，其任一者可以是來自相同或多個不同設計設施之模組式 IP 核心。積體電路 3200 包括周邊或匯流排邏輯，包括 USB 控制器 3225、UART 控制器 3230、SPI/SDIO 控制器 3235 及 I²S/I²C 控制器 3240。此外，積體電路可包括顯示裝置 3245，其耦接至一或多個高解析度多媒體介面 (HDMI) 控制器 3250 及行動產業處理器介面 (MIPI) 顯示介面 3255。可藉由包括快閃記憶體及快閃記憶體控制器的快閃記憶體子系統 3260 來提供儲存。記憶體介面可經由記憶體控制器 3265 來提供，以對於 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性引擎 3270。

【0287】圖 33 為根據實施例的顯示可使用一或多個 IP 核心來製造的系統單晶片積體電路的範例圖形處理器 3310

的方塊圖。圖形處理器3310可以是圖32的圖形處理器3210的變體。圖形處理器3310包括頂點處理器3305及一或多個片段處理器3315A-3315N(例如，3315A、3315B、3315C、3315D至3315N-1及3315N)。圖形處理器3310可經由個別的邏輯來執行不同的著色器程式，以致使頂點處理器3305被最佳化以執行針對頂點著色器程式之操作，而一或多個片段處理器3315A-3315N係執行針對片段或像素著色器程式之片段(例如，像素)著色操作。頂點處理器3305係執行3D圖形管線之頂點處理階段並產生基元及頂點資料。片段處理器3315A-3315N係使用由頂點處理器3305所產生的基元及頂點資料以產生被顯示在顯示裝置上的框緩衝器。在一個實施例中，片段處理器3315A-3315N被最佳化以執行如針對OpenGL API中所提供的片段著色器程式，其可被用來執行如針對Direct 3D API中所提供的像素著色器程式之類似操作。

【0288】圖形處理器3310額外地包括一或多個記憶體管理單元(MMU)3320A-3320B、快取3325A-3325B及電路互連3330A-3330B。一或多個MMU 3320A-3320B係提供針對積體電路3310之虛擬至實體位址映射，包括針對頂點處理器3305和/或片段處理器3315A-3315N，其除了一或多個快取3325A-3325B中所儲存的頂點或影像/紋理資料以外，還可參考記憶體中所儲存的頂點或影像/紋理資料。在一個實施例中，一或多個MMU 3325A-3325B可被與該系統內的其它MMU合成，包括與圖32中的一或多個應用處理

器 3205、影像處理器 3215 和 / 或視頻處理器 3220 相關的一或多個 MMU，以致使各處理器 3205-3220 可加入共用的或統一的虛擬記憶體系統。根據實施例，一或多個電路互連 3330A-3330B 經由 SoC 的內部匯流排或經由直接連接致使了圖形處理器 3310 與 SoC 內的其它 IP 核心介接。

【0289】圖 34 為根據實施例的顯示可使用一或多個 IP 核心來製造的系統單晶片積體電路的額外範例圖形處理器 3410 的方塊圖。圖形處理器 3410 可以是圖 32 之圖形處理器 3210 的變體。圖形處理器 3410 包括圖 33 之積體電路 3300 的一或多個 MMU 3320A-3320B、快取 3325A-3325B 及電路互連 3330A-3330B。

【0290】圖形處理器 3410 包括一或多個著色器核心 3415A-3415N (例如，3415A、3415B、3415C、3415D、3415E、3415F 至 3415N-1 及 3415N)，其提供統一的著色器核心架構，其中單一核心或核心類型可執行所有類型的可編程著色器碼 (包括著色器程式碼) 以實現頂點著色器、片段著色器和 / 或計算著色器。所存在之著色器核心的精確數目可在實施例及實現之間變化。此外，圖形處理器 3410 包括核心間工作管理器 3405，其作用為用以將執行緒調度至一或多個著色器核心 3415A-3415N 的執行緒調度器及用以加速針對磚片為基的渲染之填磚操作的填磚單元 3418，其中針對一場景之渲染操作被細分於影像空間中，例如，用以利用一場景內的局部空間同調性或者最佳化內部快取之使用。

【0291】下面的條款和/或範例關於具體實施例或其範例。範例中的細節可以在一或多個實施例中的任何地方使用。不同實施例或範例的各種特徵可以與所包含的一些特徵以及排除在外的一些特徵進行各種組合，以適應各種不同的應用。範例可以包含諸如方法之類的主題、用於執行該方法的動作的裝置、包含指令的至少一個機器可讀取媒體，所述指令在由機器執行時，致使機器執行該方法的動作，或者根據本文描述的實施例和範例的設備或系統。各種組件可以是用於執行所描述的運算或功能的裝置。

【0292】一個實施例提供了一種用以執行機器學習運算的計算設備，該計算設備包含指令解碼邏輯，用以將包含多個運算元的單一指令解碼為單一解碼指令，該多個運算元具有不同的精確度，以及通用圖形計算單元，包含第一邏輯單元和第二邏輯單元，該通用圖形計算單元用以執行該單一解碼指令，其中用以執行該單一解碼指令包含用以對於該多個運算元中的第一組運算元，以第一精確度來執行第一指令運算，以及對於該多個運算元中的第二組運算元，以第二精確度來同時執行第二指令運算。

【0293】一個實施例提供了一種執行機器學習運算的方法，該方法包含將包含多個運算元的單一指令提取與解碼，該多個運算元參照具有不同精確度的多個資料元件，經由計算單元內的第一邏輯單元，對於該多個資料元件的第一組執行第一指令運算，該多個資料元件的該第一組具有第一精確度，與經由該第一邏輯單元執行該第一指令運

算元並行地經由該計算單元內的第二邏輯單元，對於該多個資料元件的第二組執行第二指令運算，該多個資料元件的該第二組具有第二精確度，以及將該第一指令運算和該第二指令運算的結果輸出。

【0294】一個實施例提供了一種資料處理系統，包含非暫態機器可讀取媒體，用以儲存供該資料處理系統中的一或多個處理器執行的指令，以及通用圖形處理單元，包含指令解碼邏輯，該指令解碼邏輯用以將包含多個運算元的單一指令解碼為單一解碼指令，該多個運算元具有不同的精確度，以及包含第一邏輯單元和第二邏輯單元的計算單元，該計算單元用以執行該單一解碼指令，其中用以執行該單一解碼指令包含用以對於該多個運算元中的第一組運算元，以第一精確度來執行第一指令運算，以及對於該多個運算元中的第二組運算元，以第二精確度來同時執行第二指令運算。

【0295】本文中所述實施例涉及硬體的具體配置，諸如被配置以執行一些運算或具有預定功能的特殊應用積體電路(ASIC)。這種電子裝置通常包含耦接到一或多個其它組件的一或多個處理器的集合，所述其它組件諸如一或多個儲存裝置(非暫態機器可讀取儲存媒體)、使用者輸入/輸出裝置(例如鍵盤、觸控螢幕和/或顯示器)和網路連接。所述處理器的集合和其它組件的耦接通常透過一或多個匯流排和橋接器(也稱為匯流排控制器)。攜帶網路流量的儲存裝置和訊號分別代表一或多個機器可讀取儲存媒體和機器

可讀取通訊媒體。因此，給定電子裝置的儲存裝置通常儲存代碼和/或資料以供在該電子裝置的一或多個處理器的集合上執行。

【0296】當然，實施例的一或多個部分可使用軟體、韌體和/或硬體的不同組合來實現。在整個詳細說明中，出於解釋的目的，闡述了許多具體細節以便提供對本發明的透徹理解。然而，對於本領域技術人員顯而易見的是，可以在沒有這些具體細節中的一些的情況下實施這些實施例。在一些情況下，沒有詳細描述眾所皆知的結構和功能以避免模糊實施例的發明主題。因此，本發明的範圍和精神應該根據下面的申請專利範圍來判斷。

【符號說明】

【0297】

100：計算系統

101：處理子系統

102：處理器

104：系統記憶體

105：記憶體集線器

106：通訊鏈路

107：I/O 集線器

108：輸入裝置

110A：顯示裝置

110B：顯示裝置

111：I/O 子系統

112：平行處理器
 113：通訊鏈路
 114：系統儲存單元
 116：I/O 開關
 118：網路配接器
 119：無線網路配接器
 120：外接裝置
 200：平行處理器
 202：平行處理單元
 204：I/O 單元
 206：主機介面
 208：前端
 210：排程器
 212：處理叢集陣列
 214A：叢集
 214B：叢集
 214N：叢集
 216：記憶體交叉開關
 218：記憶體介面
 220A：分割單元
 220B：分割單元
 220N：分割單元
 221：L2 快取
 222：平行處理器記憶體

224A：記憶體單元
224B：記憶體單元
224N：記憶體單元
225：框緩衝器介面
226：ROP(柵格操作單元)
232：管線管理器
234：圖形多處理器
236：紋理單元
240：資料交叉開關
242：preROP
245：MMU(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(GPGPU)核心
266：載入/儲存單元
268：記憶體及快取互連
270：共用記憶體
272：快取記憶體
324：圖形多處理器
325：圖形多處理器
327：互連構造
330：指令快取

332A-332B：指令單元

334A-334B：暫存器檔案

336A-336B：GPGPU 核心

337A-337B：GPGPU 核心

338A-338B：GPGPU 核心

340A-340B：載入/儲存單元

342：快取記憶體

344A-344B：紋理單元

346：共用記憶體

350：圖形多處理器

352：互連構造

354：指令快取

356A-356D：執行資源

358A-358B：快取記憶體

360A-360D：紋理單元

362：共用記憶體

401-402：處理器記憶體

405-406：多核心處理器

407：多核心處理器

410-413：GPU

420-423：GPU 記憶體

425：代理電路

426：共用快取

430-431：記憶體互連

432：圖形處理引擎
 433：高速鏈路
 435：介面
 436：加速器積體電路
 437：介面
 438：快取
 439：記憶體管理單元(MMU)
 440-443：高速鏈路
 441：系統記憶體
 444-445：高速鏈路
 446：圖形加速模組
 447：中斷管理電路
 448：背景管理電路
 450-453：GPU 記憶體互連
 456：快取
 460A-460D：核心
 461A-461D：轉換後備緩衝器
 462A-462D：快取
 464：同調匯流排
 480：應用程式
 481：GPU 調用
 482：應用程式有效位址空間
 483：程序元件
 484：工作描述符(WD)

485：OS 虛擬位址空間
 486：分段/頁面表
 490：加速器整合片
 491：WD 提取單元
 492：中斷事件
 493：有效位址
 494A-494E：偏移/同調管理電路
 495：作業系統
 496：超管理器
 498：超管理器真實位址空間
 499：程序元件列表
 500：圖形處理管線
 502：資料組譯器
 504：頂點處理單元
 506, 514, 518：基元組譯器
 508：鑲嵌控制處理單元
 510：鑲嵌單元
 512：鑲嵌評估處理單元
 516：幾何處理單元
 520：視埠縮放、剔除、及修剪單元
 522：柵格化器
 524：片段/像素處理單元
 526：柵格操作單元
 528：記憶體介面

602：機器學習應用程式
 604：機器學習框架
 606：計算框架
 608：GPGPU 驅動程式
 610：GPGPU 硬體
 700：通用圖形處理單元
 702：主機介面
 704：全域排程器
 706A-H：計算叢集
 708：快取記憶體
 710：GPU 鏈路
 712A-B：記憶體控制器
 714A-B：記憶體
 800：多 GPU 計算系統
 802：處理器
 804：主機介面開關
 806A-D：GPGPU
 816：點對點 GPU 至 GPU 鏈路
 902：輸入
 904：卷積層
 906：卷積層
 908：完全連接層
 912：卷積層
 914：卷積層

916：卷積階段
918：檢測器階段
920：池化階段
922：下一層
1000：遞歸神經網路
1002：輸入層
1004：隱藏層
1005：回饋機制
1006：輸出層
1102：訓練資料集
1104：訓練框架
1106：未訓練神經網路
1107：經訓練神經網路
1108：經訓練神經網
1112：新資料
1202：模型平行性
1204：資料平行性
1206：結合的模型及資料平行性
1300：系統單晶片(SOC)
1302：媒體處理器
1304：視覺處理器
1305：晶載記憶體
1306：GPGPU
1308：多核心處理器

1400：多處理器單元
 1402：提取和解碼單元
 1404：分支單元
 1406：暫存器檔案
 1406：執行緒管理器
 1410：單一指令多執行緒單元
 1411A-1418A：ALU
 1411B-1418B：FPU
 1420：電壓和頻率管理器
 1500：混合精確度處理系統
 1501：輸入暫存器
 1502A-1502C：浮點元件
 1503：輸入暫存器
 1504A-1504C：整數元件
 1505：輸入暫存器
 1506A-1506C：整數元件
 1508A：FPU
 1508B：ALU
 1509：計算單元
 1510：運算碼
 1512A-1512C：狀態旗標
 1520：輸出暫存器
 1522A-1522C：結果
 1600：混合精確度處理系統

1602：FP 工作負載
 1604：INT 工作負載
 1606A：FPU
 1606B：ALU
 1607：計算單元
 1608A：FPU
 1608B：ALU
 1609：計算單元
 1700：運算邏輯
 1702：方塊
 1704：方塊
 1706：方塊
 1708：方塊
 1710：方塊
 1800：運算邏輯
 1802：方塊
 1804：方塊
 1805：方塊
 1806：方塊
 1807：方塊
 1808：方塊
 1900：機器學習系統
 1902：未訓練神經網路
 1903：訓練系統

1904：經訓練神經網路
1905：權重資料側寫系統
1906：頻率編碼權重資料
1908：GPGPU 記憶體
1910：編碼側寫
1911：輸入資料
1912：GPGPU 權重解碼器
1913：經解碼權重資料
1914：GPGPU 計算單元
1915：GPGPU 計算結果
2000：邏輯運算
2002：方塊
2004：方塊
2006：方塊
2008：方塊
2010：方塊
2012：方塊
2100：處理系統
2102：處理器
2104：快取記憶體
2106：暫存器檔案
2107：處理器核心
2108：圖形處理器
2109：指令集

2110：處理器匯流排
 2112：外部圖形處理器
 2116：記憶體控制器集線器
 2120：記憶體裝置
 2121：指令
 2122：資料
 2124：資料儲存裝置
 2126：無線收發器
 2128：韌體介面
 2130：輸入輸出(I/O)控制器集線器
 2134：網路控制器
 2140：傳統 I/O 控制器
 2142：通用串列匯流排(USB)控制器
 2144：鍵盤及滑鼠
 2146：音頻控制器
 2200：處理器
 2202A-2202N：處理器核心
 2204A-2204N：快取單元
 2206：共用快取單元
 2208：圖形處理器
 2210：系統代理核心
 2211：顯示控制器
 2212：環狀互連單元
 2213：I/O 鏈路

2214：記憶體控制器
 2216：匯流排控制器單元
 2218：嵌入式記憶體模組
 2300：圖形處理器
 2302：顯示控制器
 2304：區塊影像轉移(BLIT)引擎
 2306：視頻編碼解碼引擎
 2310：圖形處理引擎(GPE)
 2312：3D 管線
 2314：記憶體介面
 2315：3D/媒體子系統
 2316：媒體管線
 2320：顯示裝置
 2403：命令串流器
 2410：圖形處理引擎
 2414：圖形核心陣列
 2418：統一返回緩衝器(URB)
 2420：共用功能邏輯
 2421：取樣器
 2422：數學
 2423：執行緒間通訊(ITC)
 2425：快取
 2500：圖形處理器
 2502：環狀互連

2503：命令串流器

2504：管線前端

2530：視頻品質引擎(VQE)

2533：多格式編碼/解碼(MFX)

2534：視頻前端

2536：幾何管線

2537：媒體引擎

2550A-2550N：第一子核心

2552A-2552N：第一組執行單元

2554A-2554N：媒體/紋理取樣器

2560A-2560N：第二子核心

2562A-2562N：第二組執行單元

2564A-2564N：取樣器

2570A-2570N：共用資源

2580A-2580N：圖形核心

2600：執行邏輯

2602：著色器處理器

2604：執行緒調度器

2606：指令快取

2608A-2608N：執行單元

2610：取樣器

2612：資料快取

2614：資料埠

2700：圖形處理器指令格式

2710：128 位元指令格式
 2712：指令運算碼
 2713：索引欄位
 2714：指令控制欄位
 2716：執行大小欄位
 2718：目的地
 2720：src0
 2722：src1
 2724：SRC2
 2726：存取/位址模式欄位
 2730：64 位元壓縮指令格式
 2740：運算碼解碼
 2742：移動和邏輯運算碼群組
 2744：流程控制指令群組
 2746：雜項指令群組
 2748：平行數學指令群組
 2750：向量數學群組
 2800：圖形處理器
 2802：環狀互連
 2803：命令串流器
 2805：頂點提取器
 2807：頂點著色器
 2811：殼體著色器
 2813：鑲嵌器

2817：領域著色器
2819：幾何著色器
2820：圖形管線
2823：串流輸出單元
2829：截波器
2830：媒體管線
2831：執行緒調度器
2834：視頻前端
2837：媒體引擎
2840：顯示引擎
2841：2D 引擎
2843：顯示控制器
2850：執行邏輯
2851：L1 快取
2852A-2852N：執行單元
2854：媒體取樣器
2856：資料埠
2858：紋理/取樣器快取
2870：渲染輸出管線
2873：柵格化器及深度測試組件
2875：L3 快取
2877：像素操作組件
2878：渲染快取
2879：深度快取

2900：圖形處理器命令格式
 2902：目標客戶
 2904：命令操作碼(運算碼)
 2905：子運算碼
 2906：相關資料
 2908：命令大小
 2910：圖形處理器命令序列
 2912：管線清除命令
 2913：管線選擇命令
 2914：管線控制命令
 2916：返回緩衝器狀態命令
 2920：管線判定
 2922：3D 管線
 2924：媒體管線
 2930：3D 管線狀態
 2932：3D 基元
 2934：執行
 2940：媒體管線狀態
 2942：媒體物件命令
 2944：執行命令
 3000：資料處理系統
 3010：3D 圖形應用程式
 3012：著色器指令
 3014：可執行指令

3016：圖形物件
3020：作業系統
3022：圖形 API
3024：前端著色器編譯器
3026：使用者模式圖形驅動程式
3027：後端著色器編譯器
3028：作業系統內核模式功能
3029：內核模式圖形驅動程式
3030：處理器
3032：圖形處理器
3034：通用處理器核心
3050：系統記憶體
3100：IP 核心開發系統
3110：軟體模擬
3112：模擬模型
3115：暫存器轉移階層(RTL)設計
3120：硬體模型
3130：設計設施
3140：非揮發性記憶體
3150：有線連接
3160：無線連接
3165：製造設施
3200：系統單晶片積體電路
3205：應用處理器

3210：圖形處理器
3215：影像處理器
3220：視頻處理器
3225：USB 控制器
3230：UART 控制器
3235：SPI/SDIO 控制器
3240：I²S/I²C 控制器
3245：顯示裝置
3250：高解析度多媒體介面(HDMI)控制器
3255：行動產業處理器介面(MIPI)顯示介面
3260：快閃記憶體子系統
3265：記憶體控制器
3270：嵌入式安全性引擎
3305：頂點處理器
3310：圖形處理器
3315A-3315N：片段處理器
3320A-3320N：記憶體管理單元(MMU)
3325A-3325N：快取
3330A-3330N：電路互連
3405：核心間工作管理器
3410：圖形處理器
3415A-3415N：著色器核心
3418：填磚單元

【發明申請專利範圍】

【第1項】

一種通用圖形處理單元，包含：

串流多處理器，其具有包含硬體多執行緒的單一指令多執行緒(SIMT)架構，其中該串流多處理器包含：

第一處理區塊，其包含具有第一浮點資料路徑的第一處理核心與具有第一整數資料路徑的第二處理核心，該第一整數資料路徑獨立於該第一浮點資料路徑，而該第一處理區塊額外地包含與該第一處理核心和該第二處理核心耦接的第一暫存器檔案，其中該第一整數資料路徑係用以致使第一指令的執行，而該第一浮點資料路徑係用以致使第二指令的執行，該第一指令將與該第二指令同時執行；

第二處理區塊，其包含具有第二浮點資料路徑的第三處理核心與具有第二整數資料路徑的第四處理核心，該第二整數資料路徑獨立於該第二浮點資料路徑，而該第二處理區塊額外地包含與該第三處理核心和該第四處理核心耦接的第二暫存器檔案，其中該第二整數資料路徑係用以致使第三指令的執行，而該第二浮點資料路徑係用以致使第四指令的執行，該第三指令將與該第四指令同時執行，而該第一暫存器檔案不同於該第二暫存器檔案，其中該第四指令包含四個運算元並且係用以致使該第三處理核心對於兩個16位元來源運算元及一32位元來源運算元進行融合的乘加運算；以及

記憶體，其與該第一處理區塊和該第二處理區塊耦

接。

【第2項】

如申請專利範圍第1項的通用圖形處理單元，其中該記憶體係在該第一處理區塊和該第二處理區塊之間共用。

【第3項】

如申請專利範圍第2項的通用圖形處理單元，其中該記憶體包含至少可由該第一處理核心和該第三處理核心存取的資料快取。

【第4項】

如申請專利範圍第1項的通用圖形處理單元，其中該串流多處理器還包含一或多個硬體排程器以將第一組的執行緒排程到至少該第一處理核心和該第三處理核心。

【第5項】

如申請專利範圍第4項的通用圖形處理單元，該一或多個硬體排程器還將第二組的執行緒排程到至少該第二處理核心和該第四處理核心。

【第6項】

如申請專利範圍第5項的通用圖形處理單元，其中：

該第一組的執行緒與該第一指令相關；以及

該第二組的執行緒與該第二指令相關。

【第7項】

如申請專利範圍第1項的通用圖形處理單元，其中該第一處理核心係用以回應於該第二指令而執行64位元浮點運算。

【第8項】

如申請專利範圍第1項的通用圖形處理單元，其中該第二處理核心係用以執行32位元整數運算，而該第四處理核心係用以執行一或多個8位元整數運算。

【第9項】

一種資料處理系統，包含：

圖形雙倍資料速率記憶體；以及

通用圖形處理器，其經由兩個或多個記憶體控制器與該圖形雙倍資料速率記憶體耦接，該通用圖形處理器包含具有單一指令多執行緒(SIMT)架構的硬體多執行緒計算單元，其中該硬體多執行緒計算單元包含：

第一暫存器檔案和第一硬體排程器，該第一暫存器檔案和該第一硬體排程器各自與第一處理核心和第二處理核心耦接，該第一處理核心具有第一浮點資料路徑，而該第二處理核心具有第一整數資料路徑，該第一整數資料路徑獨立於該第一浮點資料路徑，其中該第一整數資料路徑係用以致使第一指令的執行，而該第一浮點資料路徑係用以致使第二指令的執行，該第一指令將與該第二指令同時執行；

不同於該第一暫存器檔案的第二暫存器檔案和不同於該第一硬體排程器的第二硬體排程器，該第二暫存器檔案和該第二硬體排程器各自與第三處理核心和第四處理核心耦接，該第三處理核心具有第二浮點資料路徑，而該第四處理核心具有第二整數資料路徑，該第二整數資料路徑獨

立於該第二浮點資料路徑，其中該第二整數資料路徑係用以致使第三指令的執行，而該第二浮點資料路徑係用以致使第四指令的執行，該第三指令將與該第四指令同時執行，其中該第四指令包含四個運算元並且係用以致使該第三處理核心對於兩個16位元來源運算元及一32位元來源運算元進行融合的乘加運算；以及

內部記憶體，其與該第一處理核心、該第二處理核心、該第三處理核心和該第四處理核心耦接。

【第10項】

如申請專利範圍第9項的資料處理系統，其中該內部記憶體將在該第一處理核心、該第二處理核心、該第三處理核心和該第四處理核心之間共用，而該圖形雙倍資料速率記憶體係第六版圖形雙倍資料速率(GDDR6)記憶體。

【第11項】

如申請專利範圍第10項的資料處理系統，其中該內部記憶體包含可由該第一處理核心、該第二處理核心、該第三處理核心和該第四處理核心存取的資料快取。

【第12項】

如申請專利範圍第9項的資料處理系統，其中該第一硬體排程器係用以將第一執行緒群組的執行緒排程到該第一處理核心，且將第二執行緒群組的執行緒排程到該第二處理核心，其中該第二硬體排程器係用以將該第一執行緒群組的執行緒排程到該第三處理核心，且將該第二執行緒群組的執行緒排程到該第四處理核心，以及其中該第一執

行緒群組與該第二執行緒群組內的執行緒的執行情境將在執行期間保持晶載。

【第13項】

如申請專利範圍第12項的資料處理系統，其中該第一處理核心包含與該第一浮點資料路徑相關的第一功能單元，而該第二處理核心包含與該第一整數資料路徑相關的第二功能單元。

【第14項】

如申請專利範圍第13項的資料處理系統，該第一功能單元用以執行浮點運算，而該第二功能單元用以獨立於該第一功能單元而執行整數運算。

【第15項】

如申請專利範圍第14項的資料處理系統，其中該第一功能單元係用以對於該等兩個16位元來源運算元執行第一浮點運算並且對於該32位元來源運算元執行第二浮點運算。

【第16項】

如申請專利範圍第14項的資料處理系統，其中該第二功能單元可配置成執行8位元、16位元與32位元整數運算中的一或多個。

【第17項】

一種在通用圖形處理單元上執行的方法，該方法包含：

經由第一硬體排程器將第一組的執行緒排程到第一處

理核心並且經由第二硬體排程器將該第一組的執行緒排程到第三處理核心，該第一組的執行緒用以執行包含融合的乘加運算的一組浮點運算，該融合的乘加運算回應於具有包含兩個16位元來源運算元及一32位元來源運算元的四個運算元的指令進行，該第一處理核心和該第三處理核心包含在該通用圖形處理單元的第二處理區塊的浮點資料路徑內，該通用圖形處理單元包含具有單一指令多執行緒(SIMT)架構的硬體多執行緒計算單元，其中該第一組的執行緒與第一指令相關；

經由該第一硬體排程器將第二組的執行緒排程到第二處理核心並且經由該第二硬體排程器將該第二組的執行緒排程到第四處理核心，該第二硬體排程器不同於該第一硬體排程器，該第二組的執行緒用以執行一組整數運算，該第二處理核心和該第四處理核心包含在該通用圖形處理單元的第二處理區塊的整數資料路徑內，其中該第二組的執行緒與第二指令相關；

使用該浮點資料路徑的處理核心來執行與該第一指令相關的第一執行緒，其中執行該第一執行緒包含分配與該第一處理區塊相關的第一暫存器檔案內的第一暫存器和第三暫存器；以及

使用該整數資料路徑的處理核心來執行與該第二指令相關的第二執行緒，該第二執行緒獨立於該第一執行緒而執行並且與該第一執行緒連續，其中執行該第二執行緒包含分配與該第二處理區塊相關的第二暫存器檔案內的第二

暫存器和第四暫存器，該第一暫存器檔案不同於該第二暫存器檔案。

【第18項】

如申請專利範圍第17項的方法，還包含：

將與該第一指令相關的第一資料以及與該第二指令相關的
第二資料儲存在與該第一處理區塊和該第二處理區塊
耦接的記憶體內；以及

在該第一指令和該第二指令的執行期間，維持該通用
圖形處理單元的硬體內的該第一執行緒和該第二執行緒的
執行環境。

【第19項】

如申請專利範圍第18項的方法，還包含：

分配該第一暫存器以供該第一處理核心使用；

分配該第二暫存器以供該第二處理核心使用；

分配該第三暫存器以供該第三處理核心使用；以及

分配該第四暫存器以供該第四處理核心使用。

【第20項】

一種非暫態機器可讀取媒體，其儲存用以致使一或多
個處理器執行如申請專利範圍第17至19項中任一項的方法
之指令。

【發明圖式】

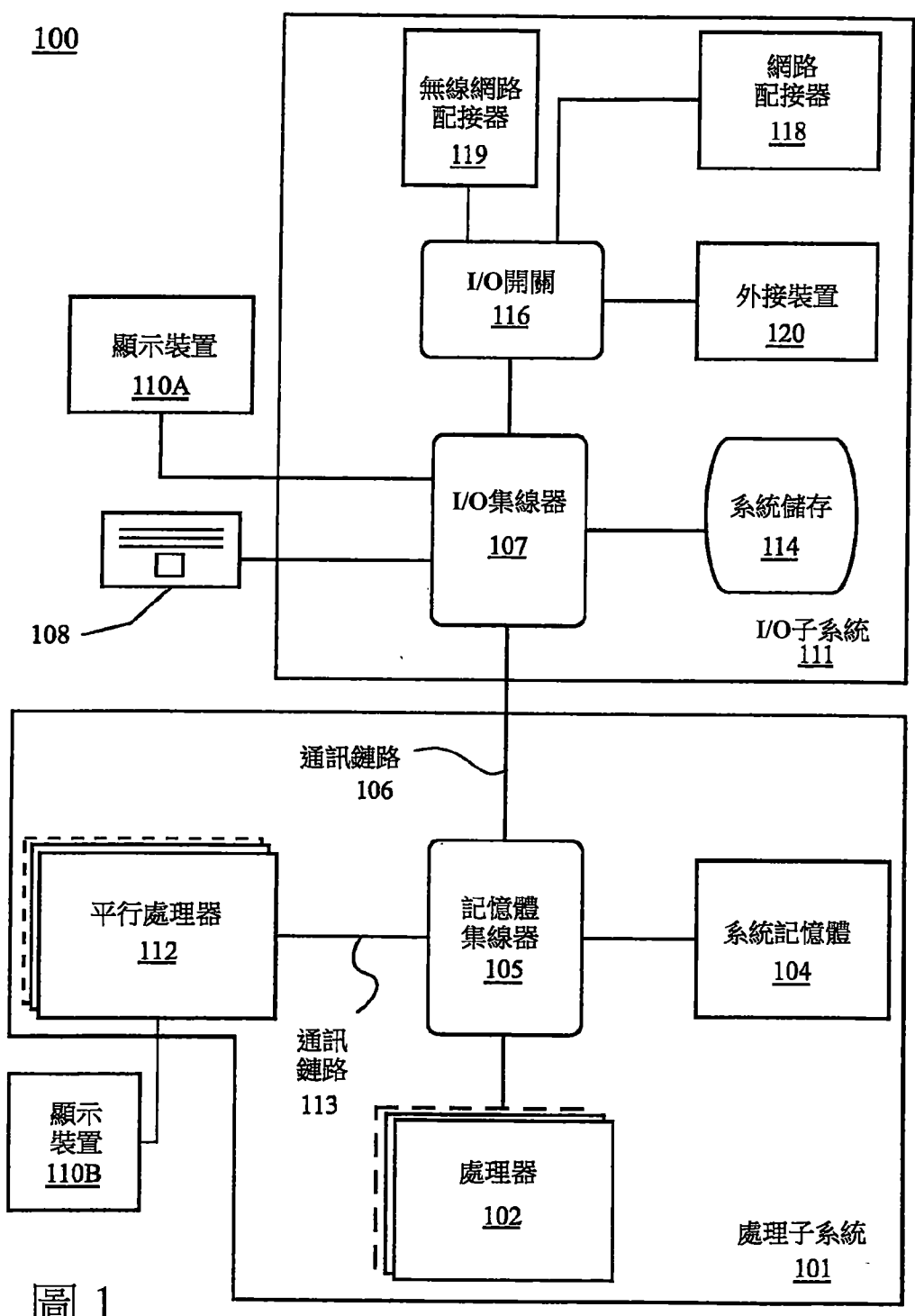


圖 1

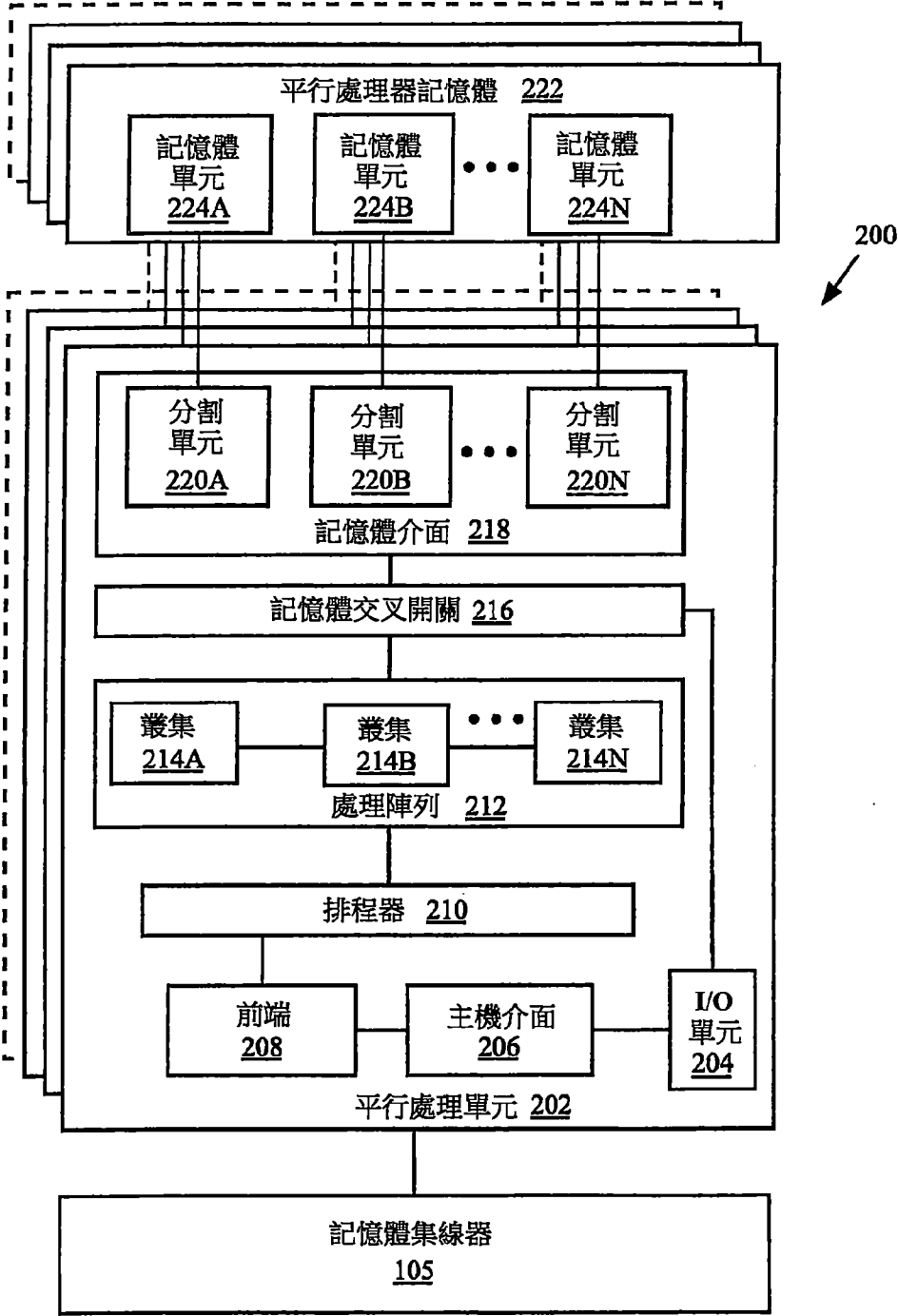


圖 2A

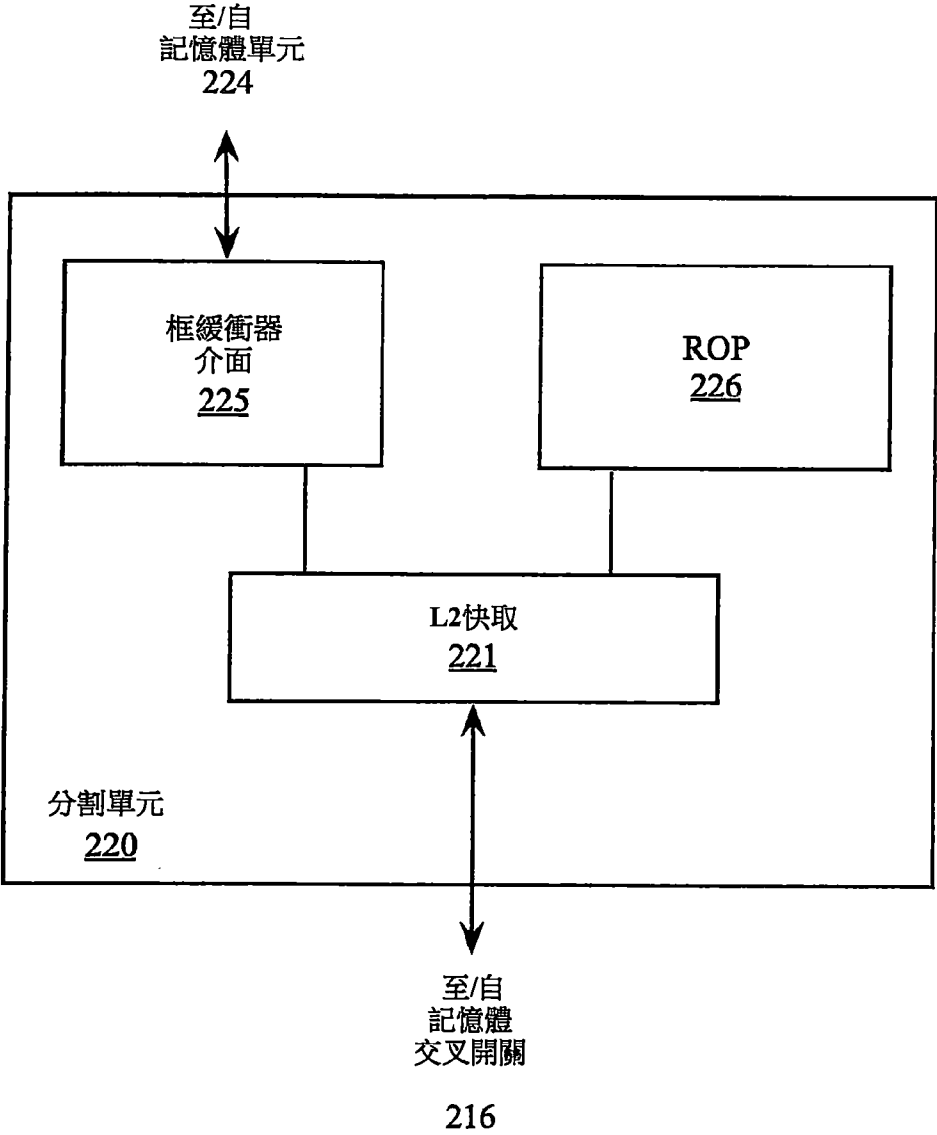


圖 2B

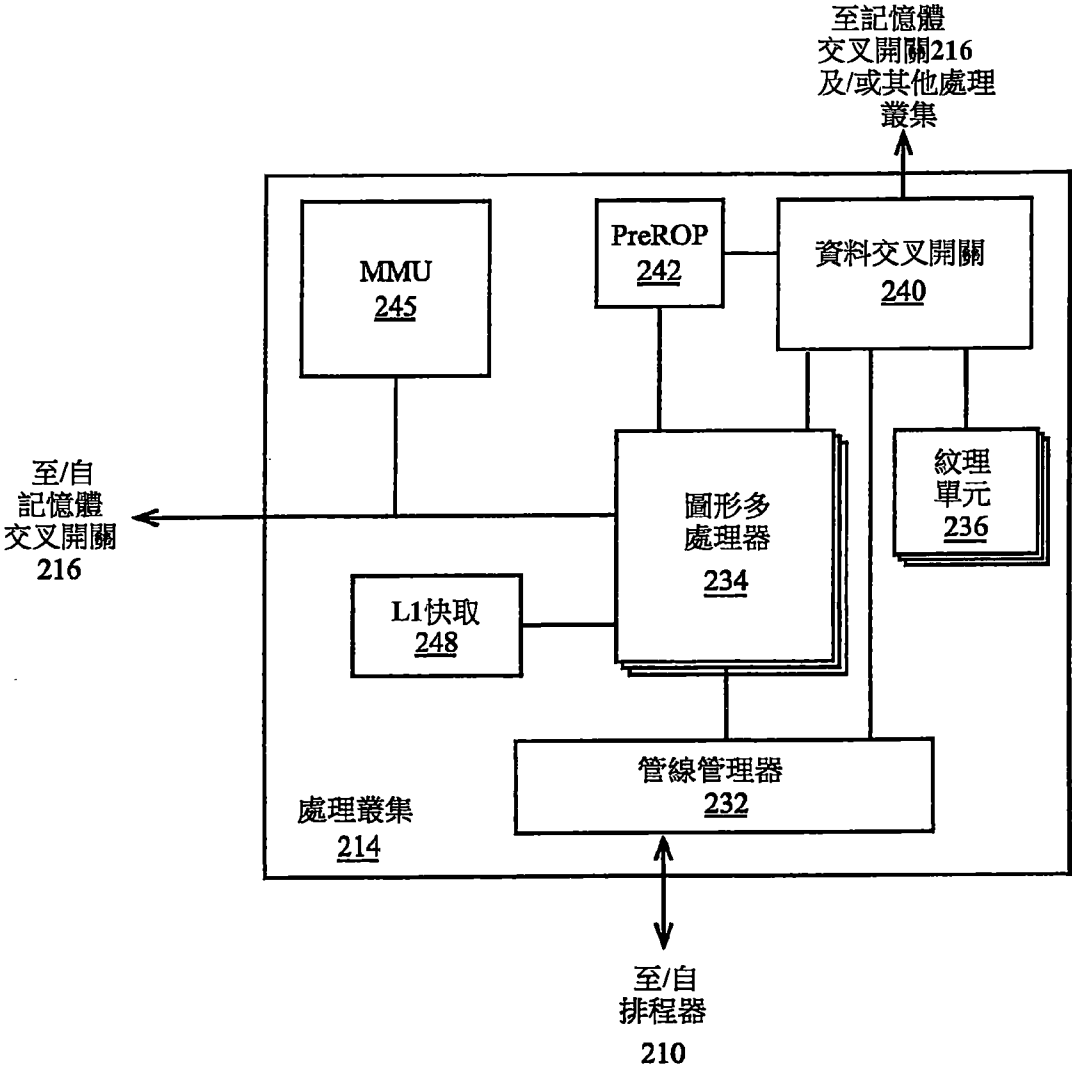


圖 2C

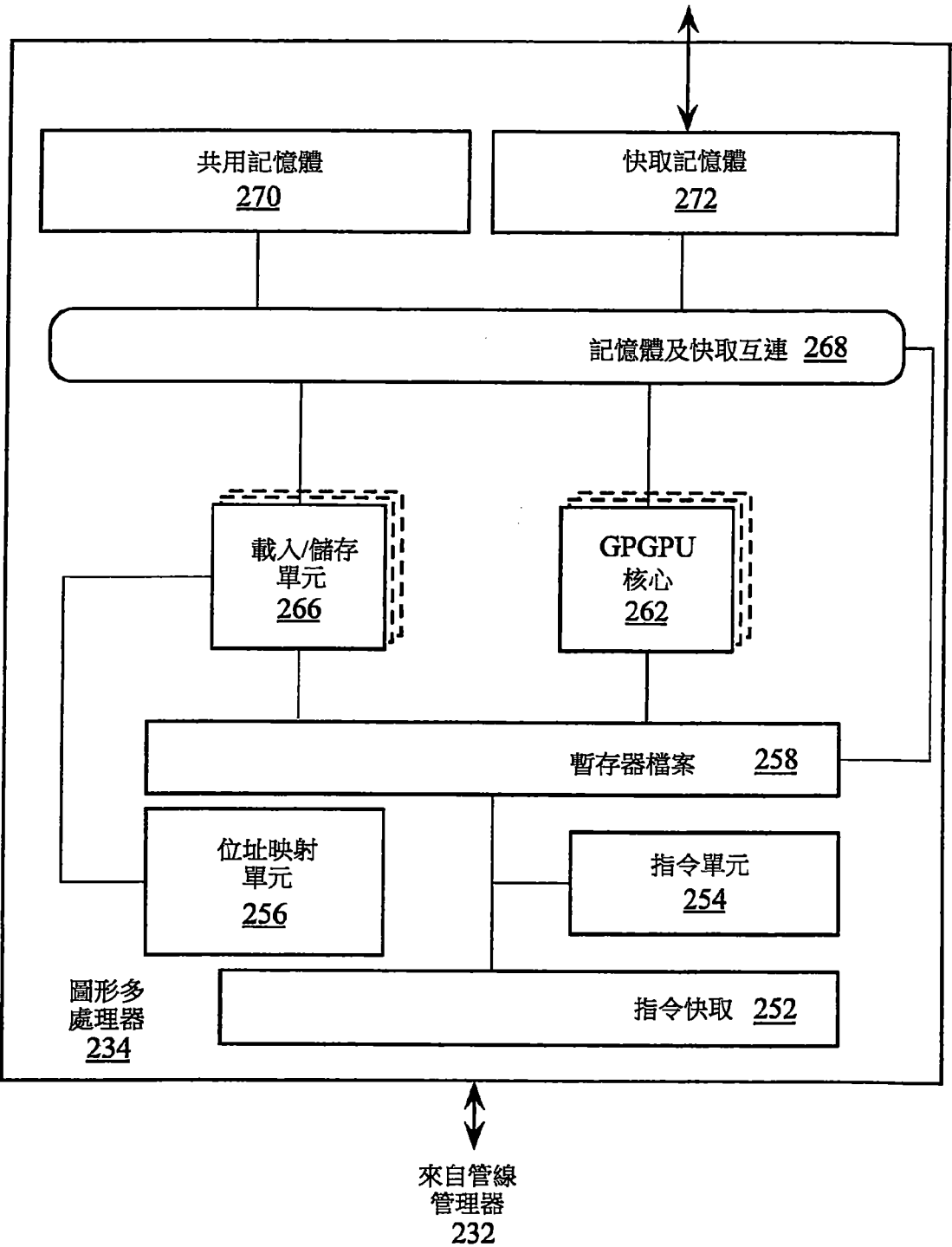


圖 2D

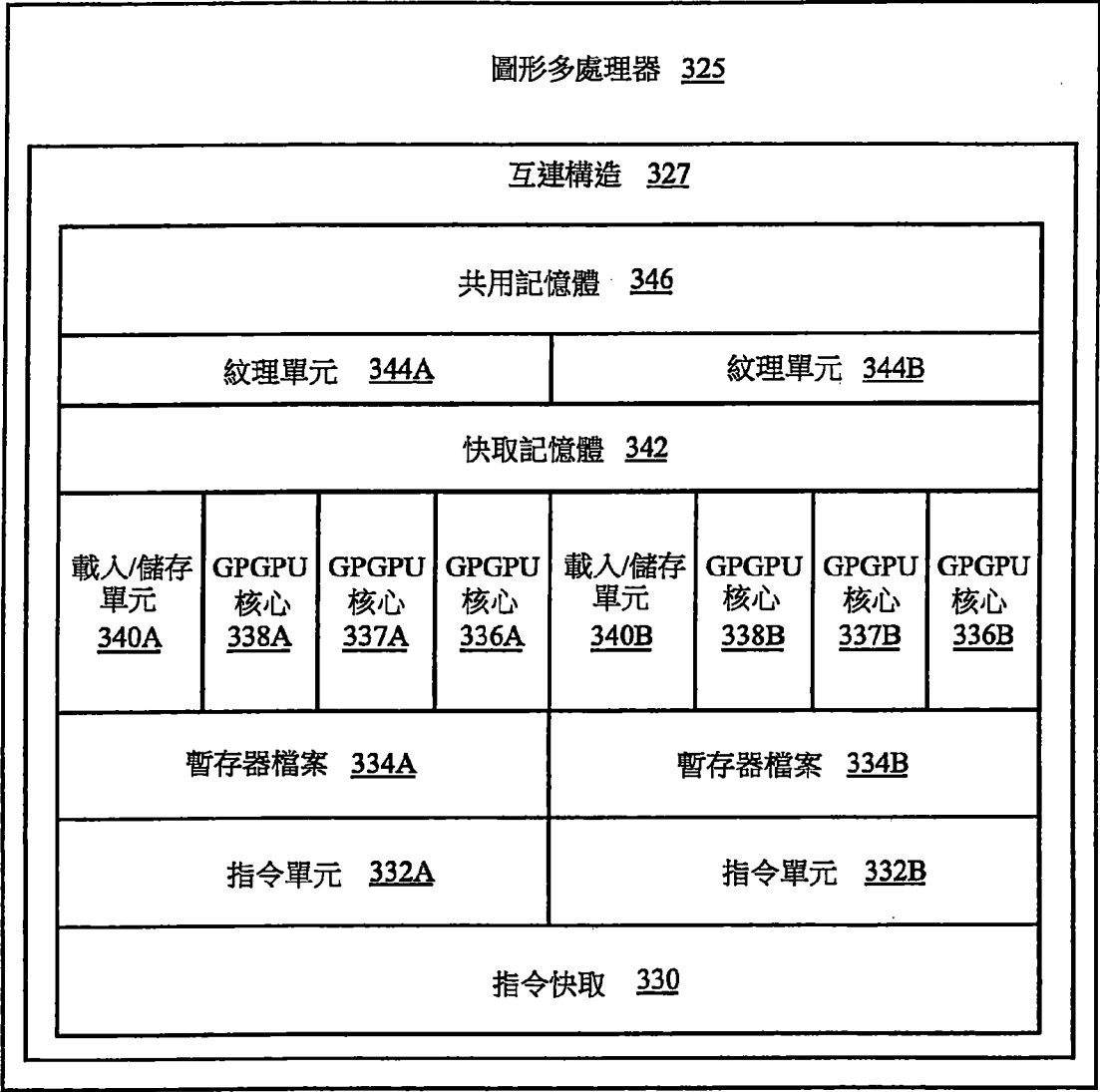


圖 3A

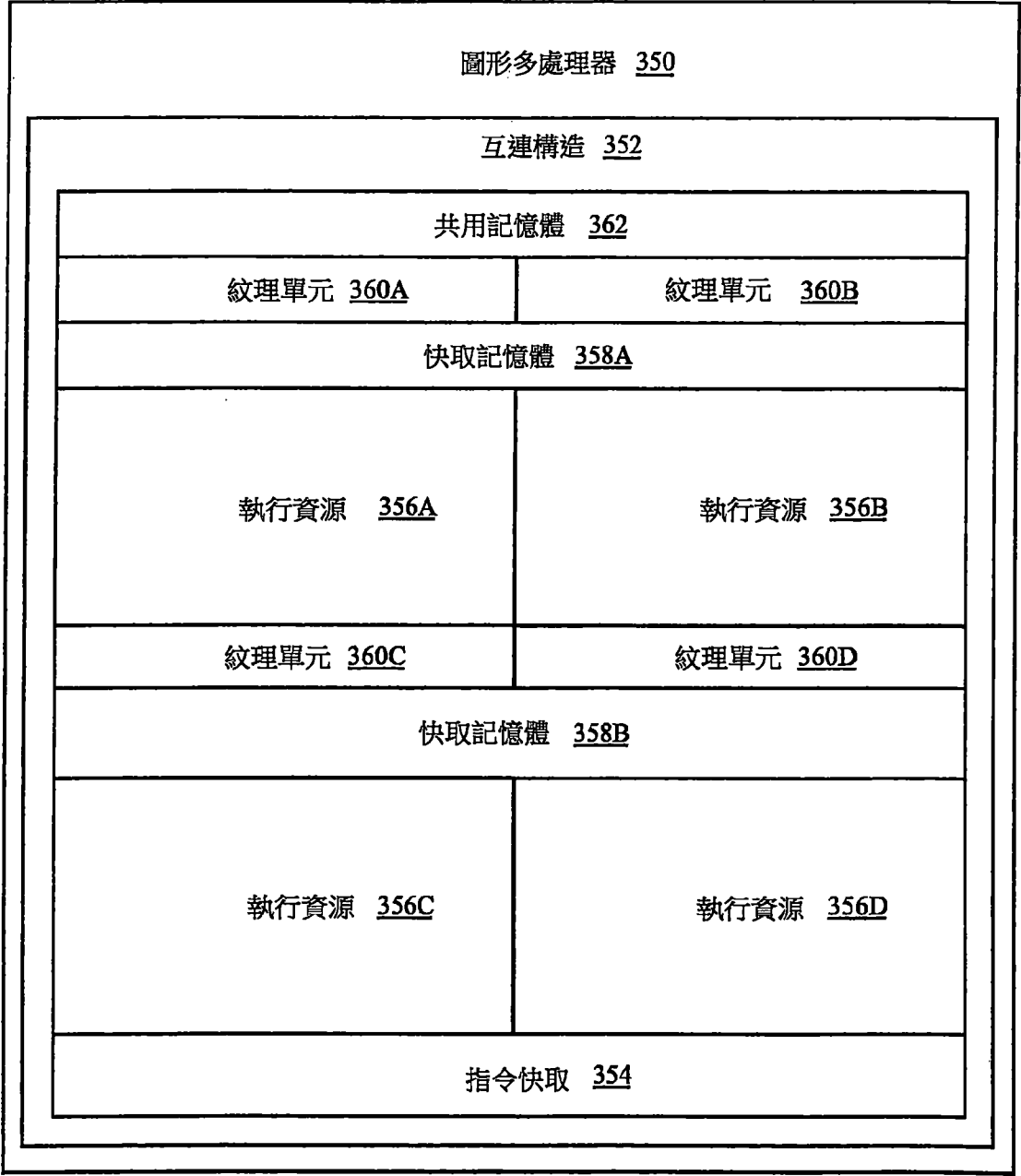


圖 3B

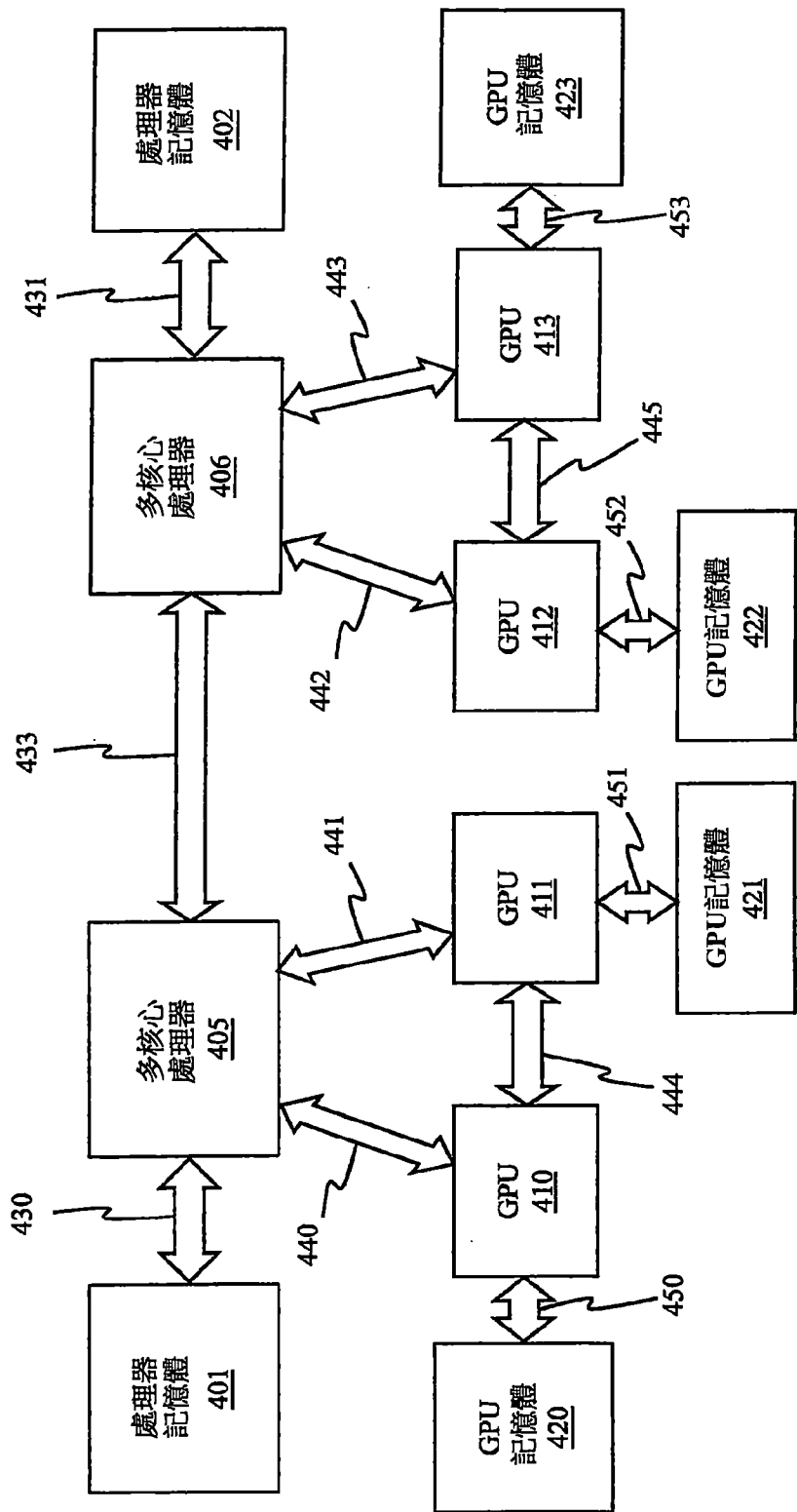


圖 4A

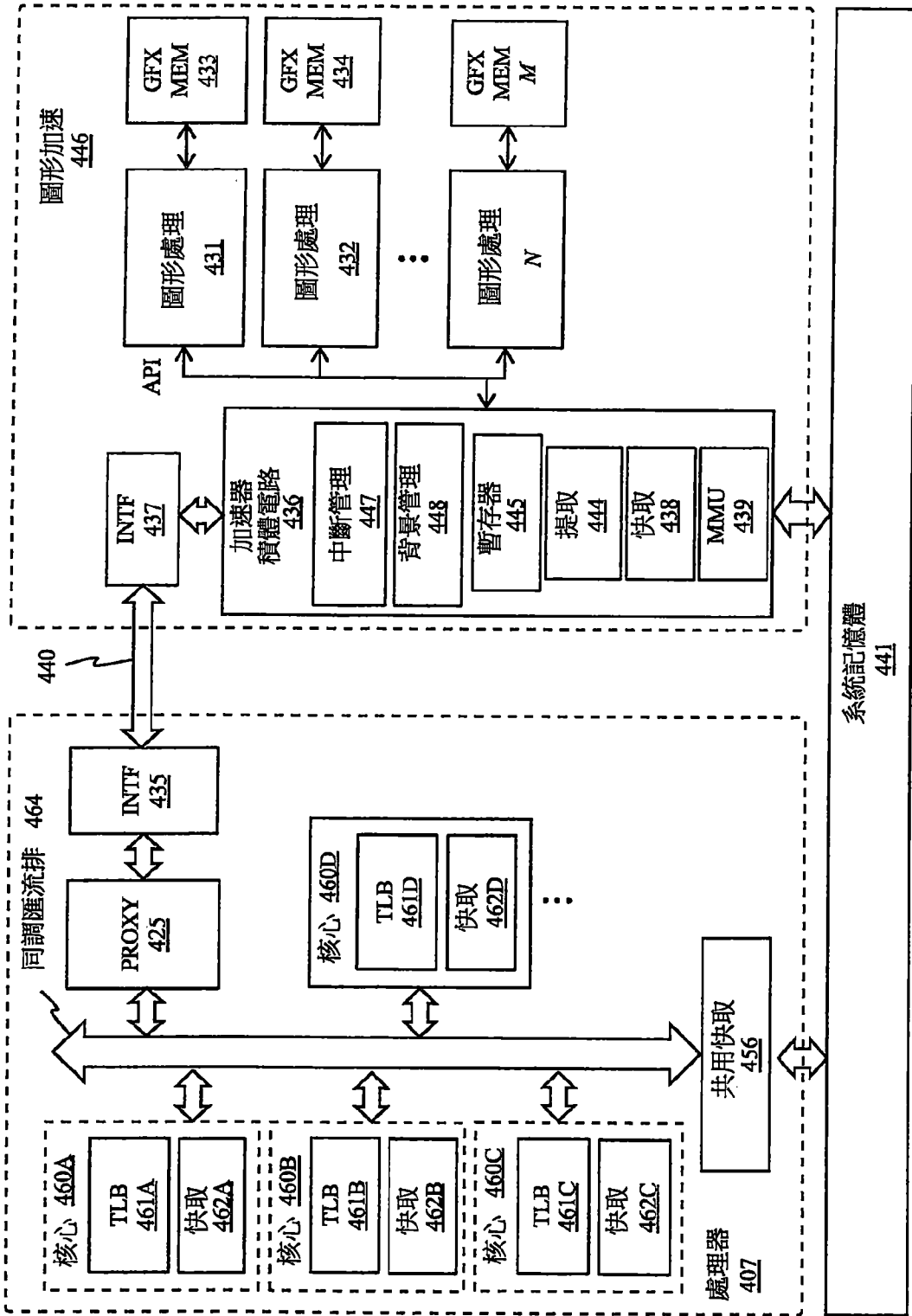


圖 4B

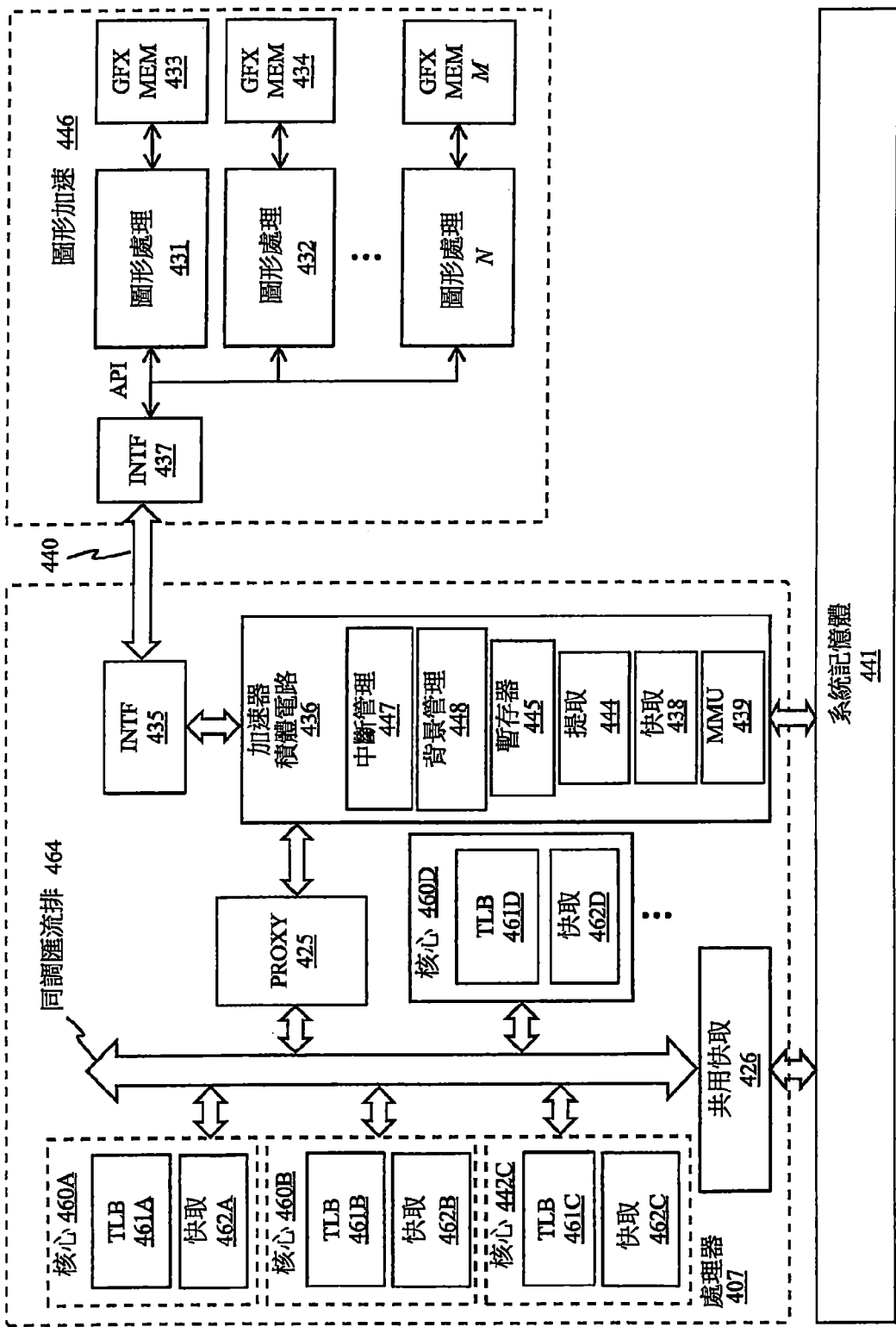


圖 4C

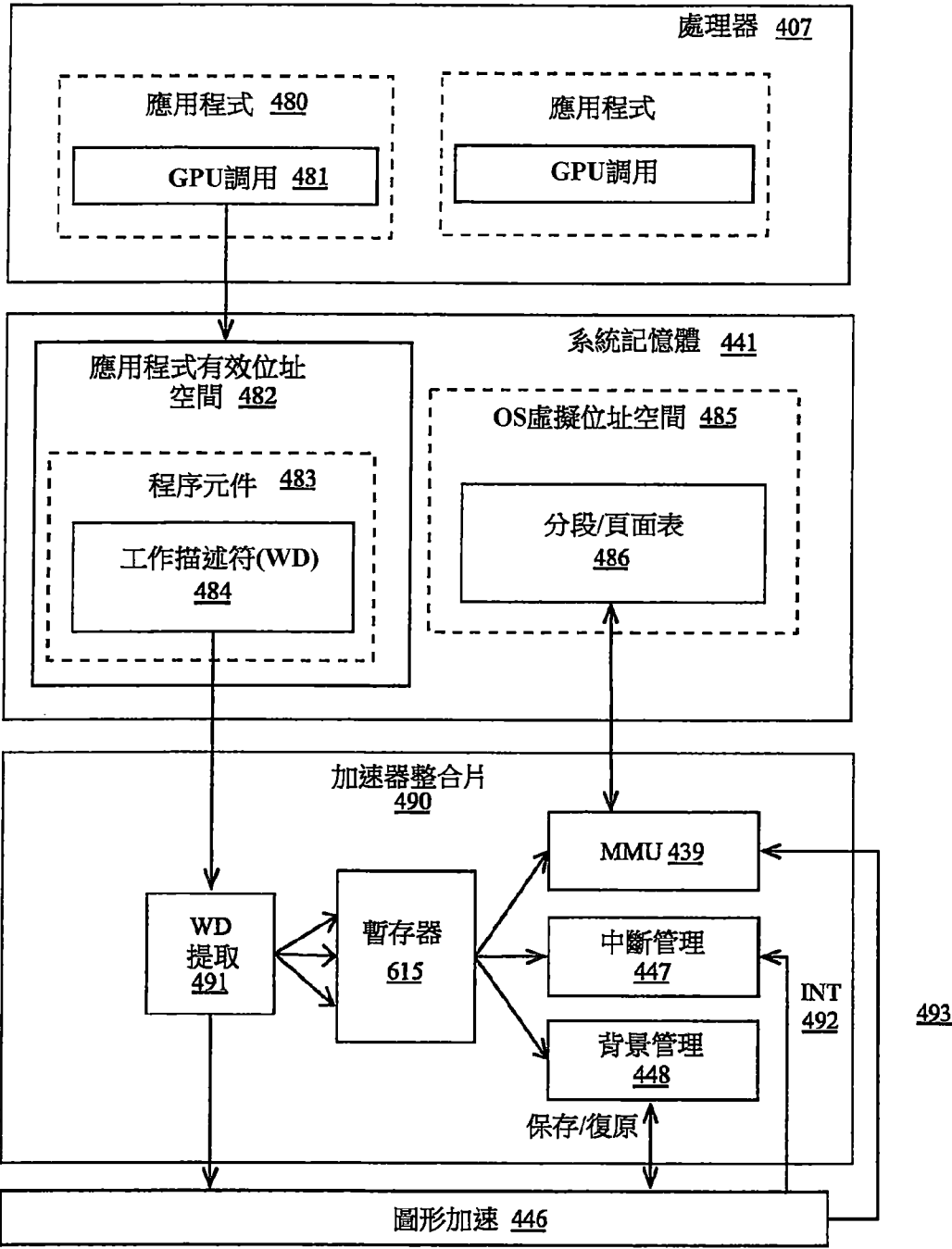


圖 4D

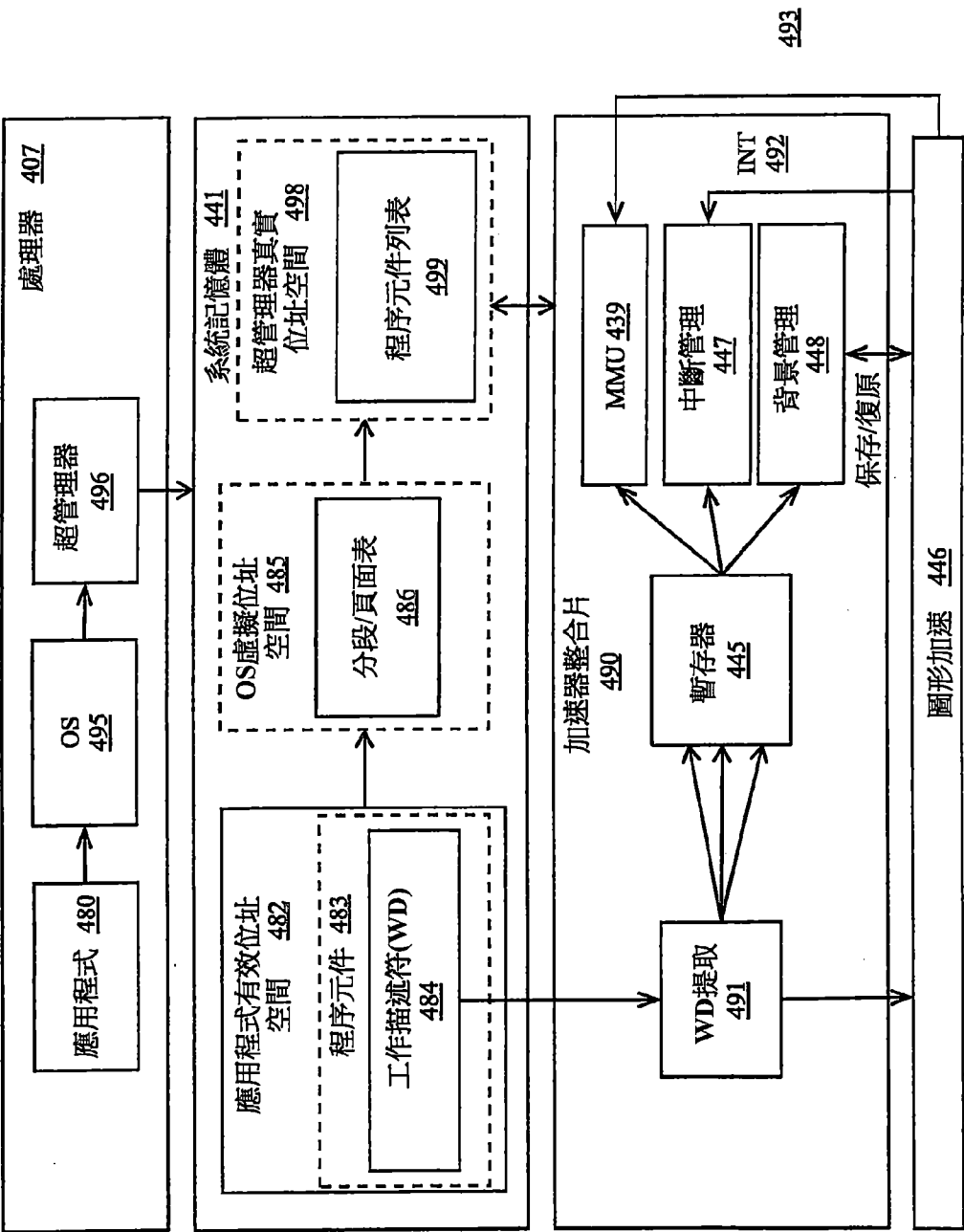


圖 4E

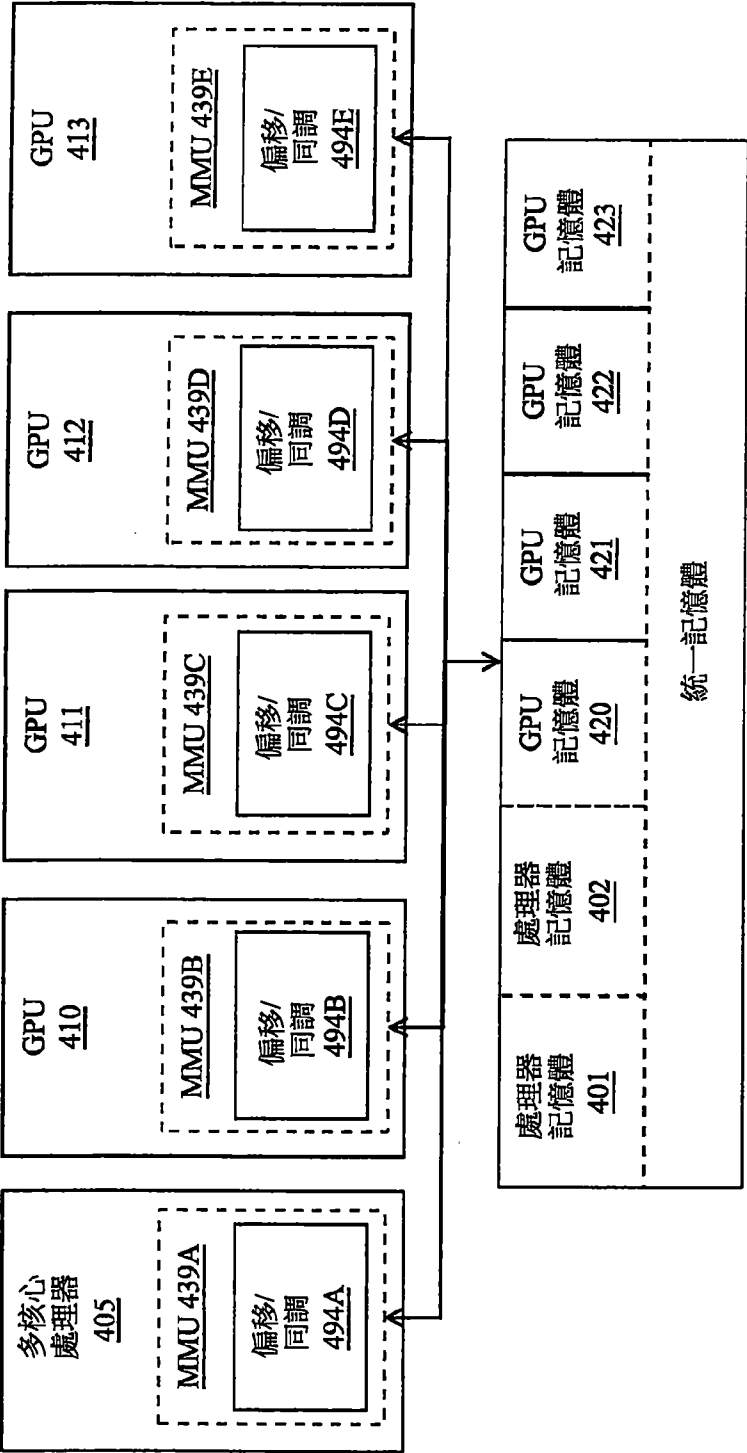


圖 4F

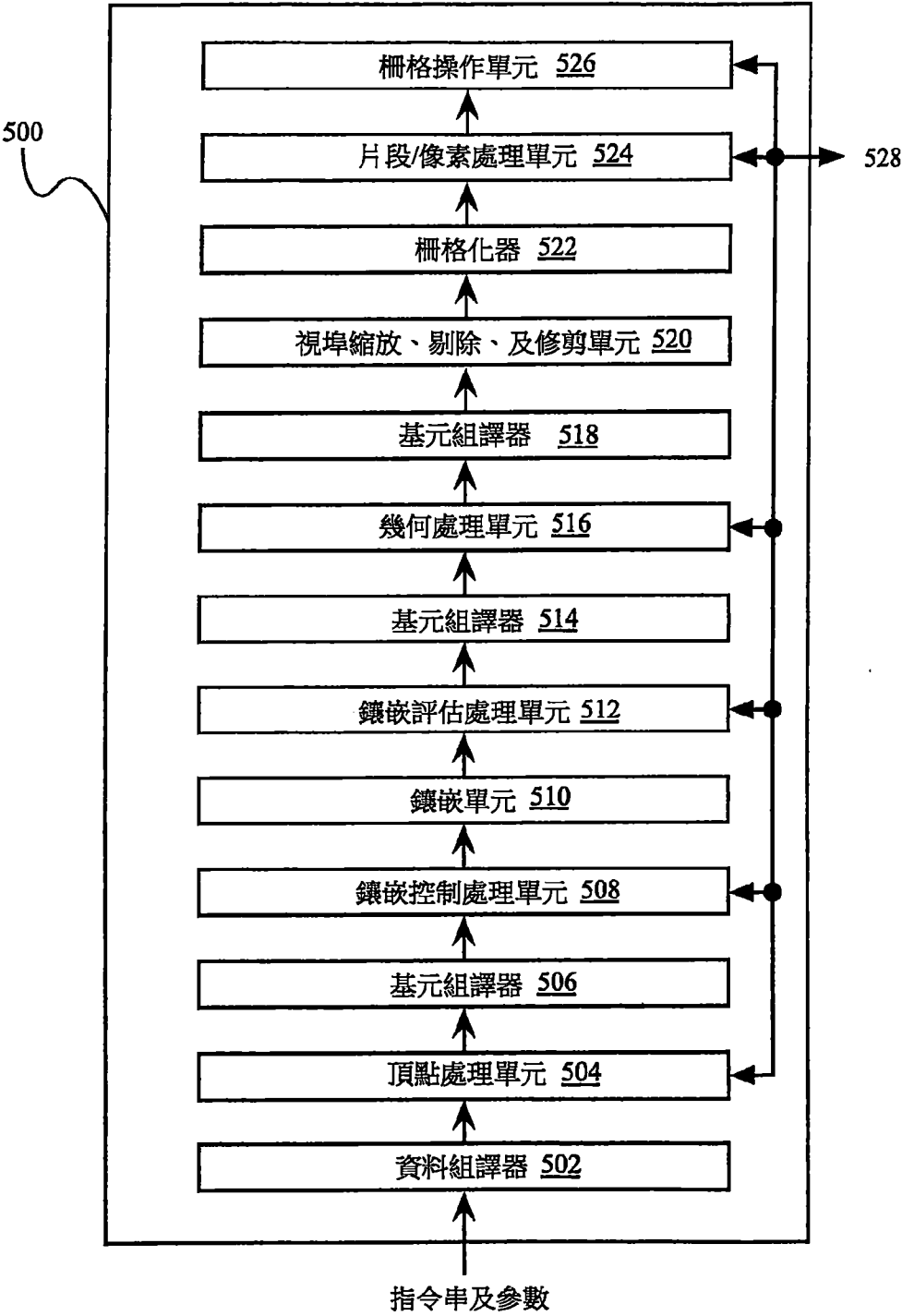


圖 5

600

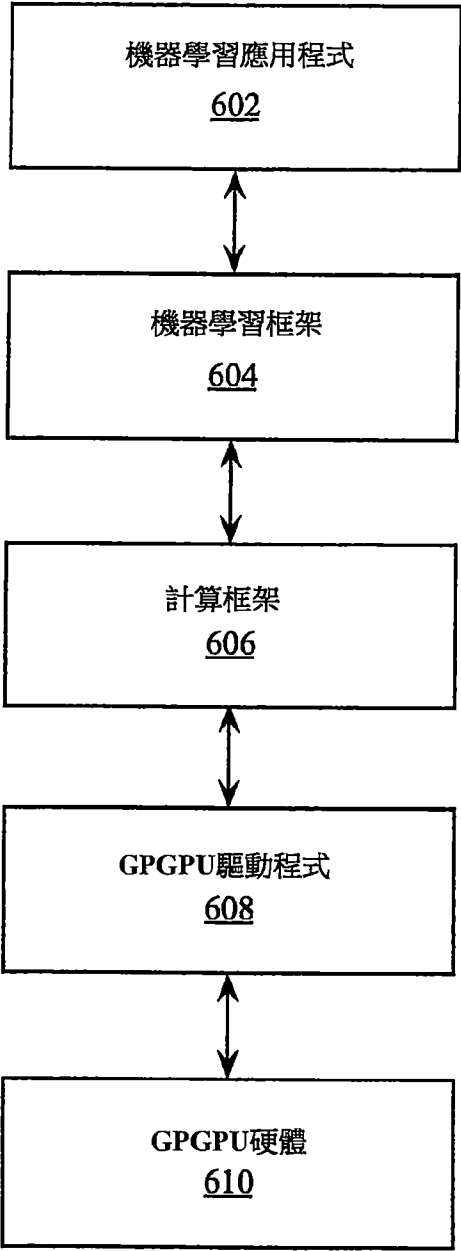


圖 6

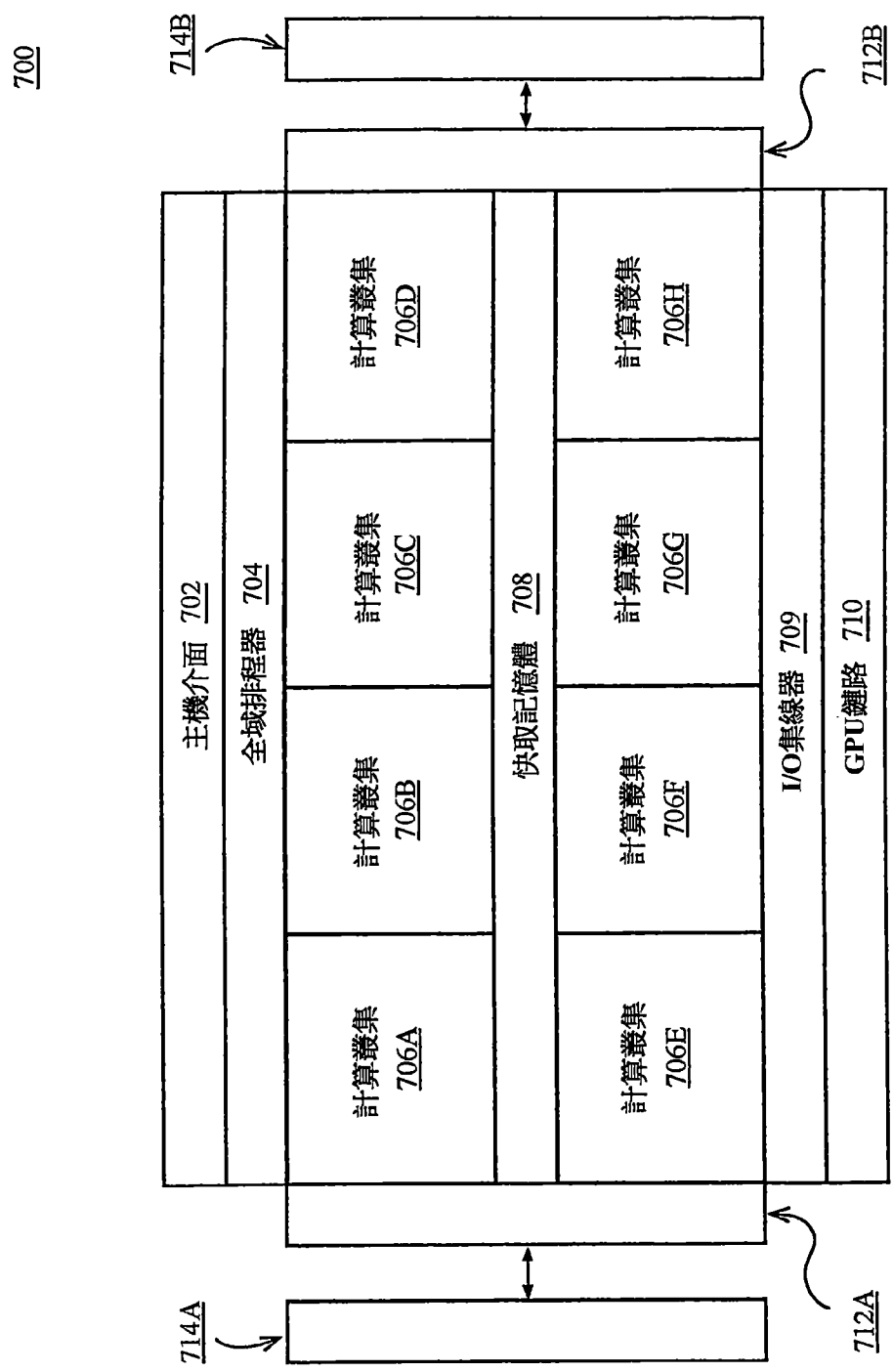


圖 7

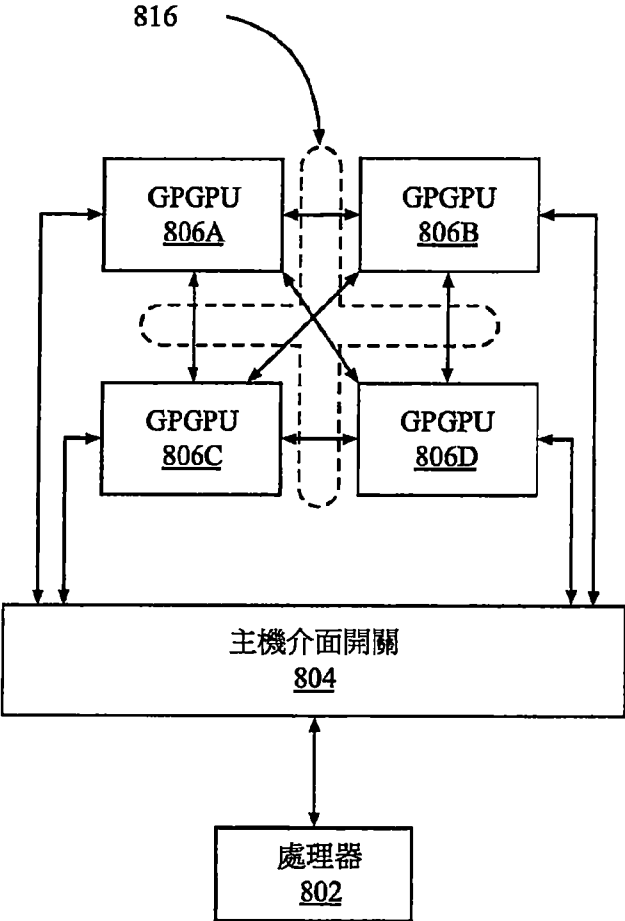


圖 8

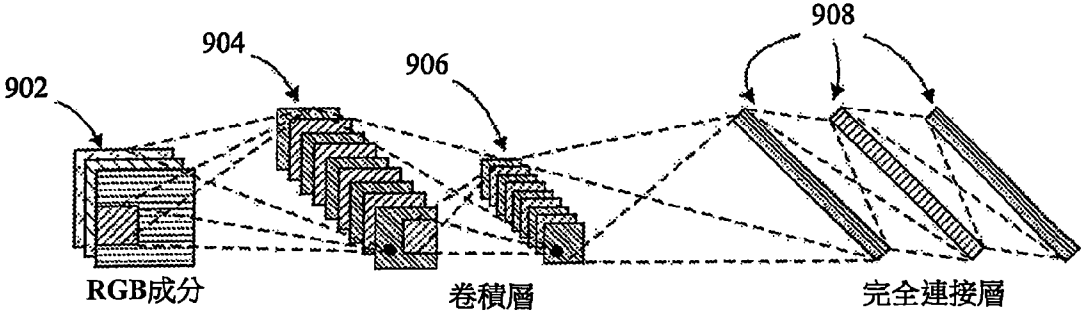


圖 9A

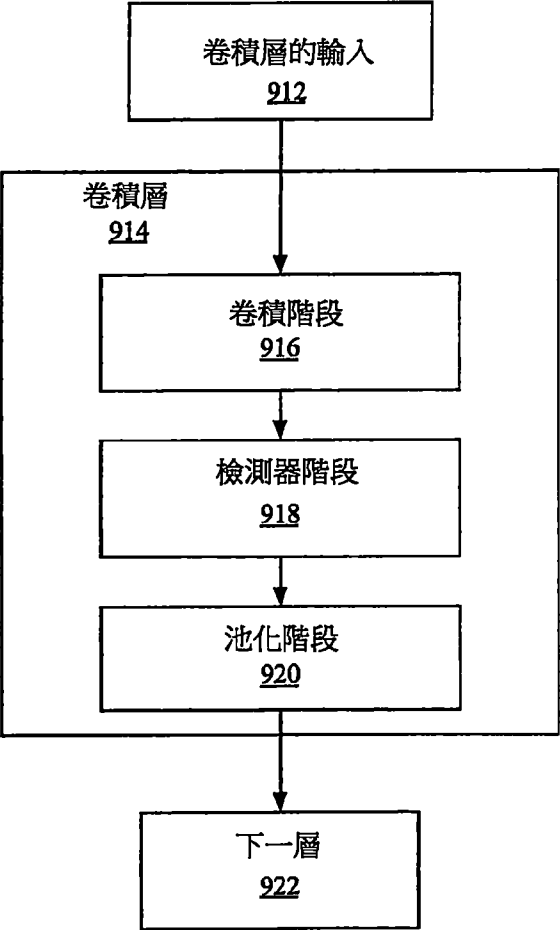


圖 9B

1000

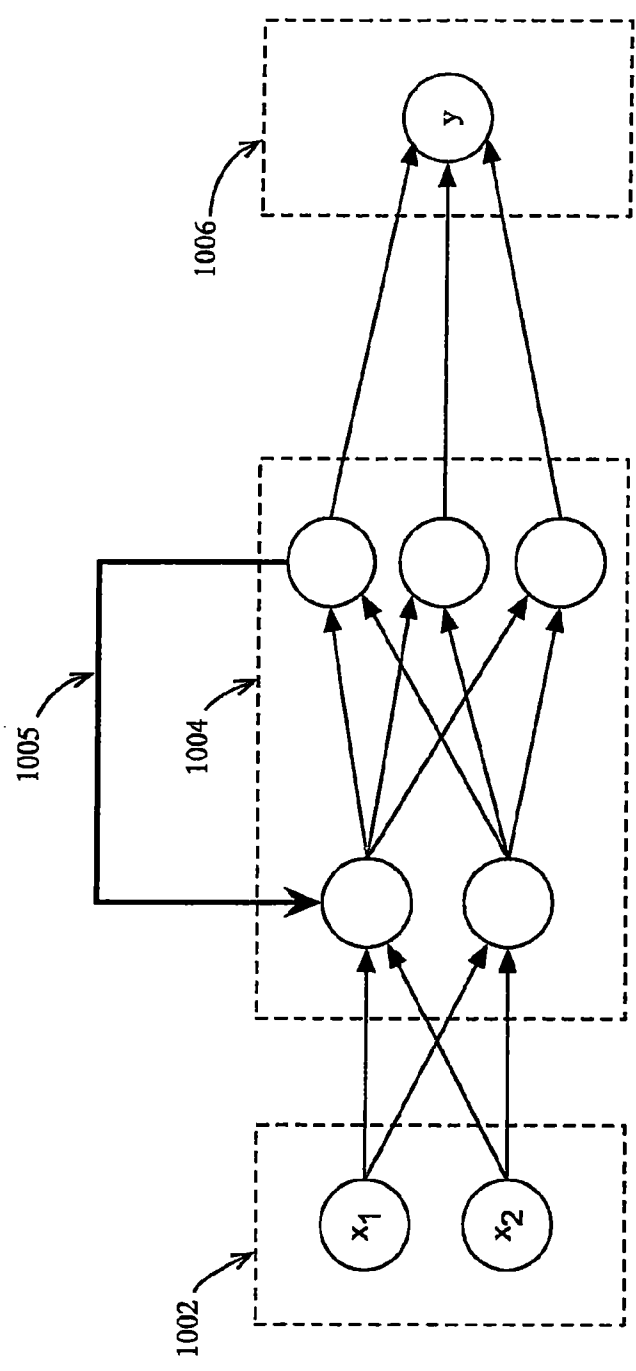


圖 10

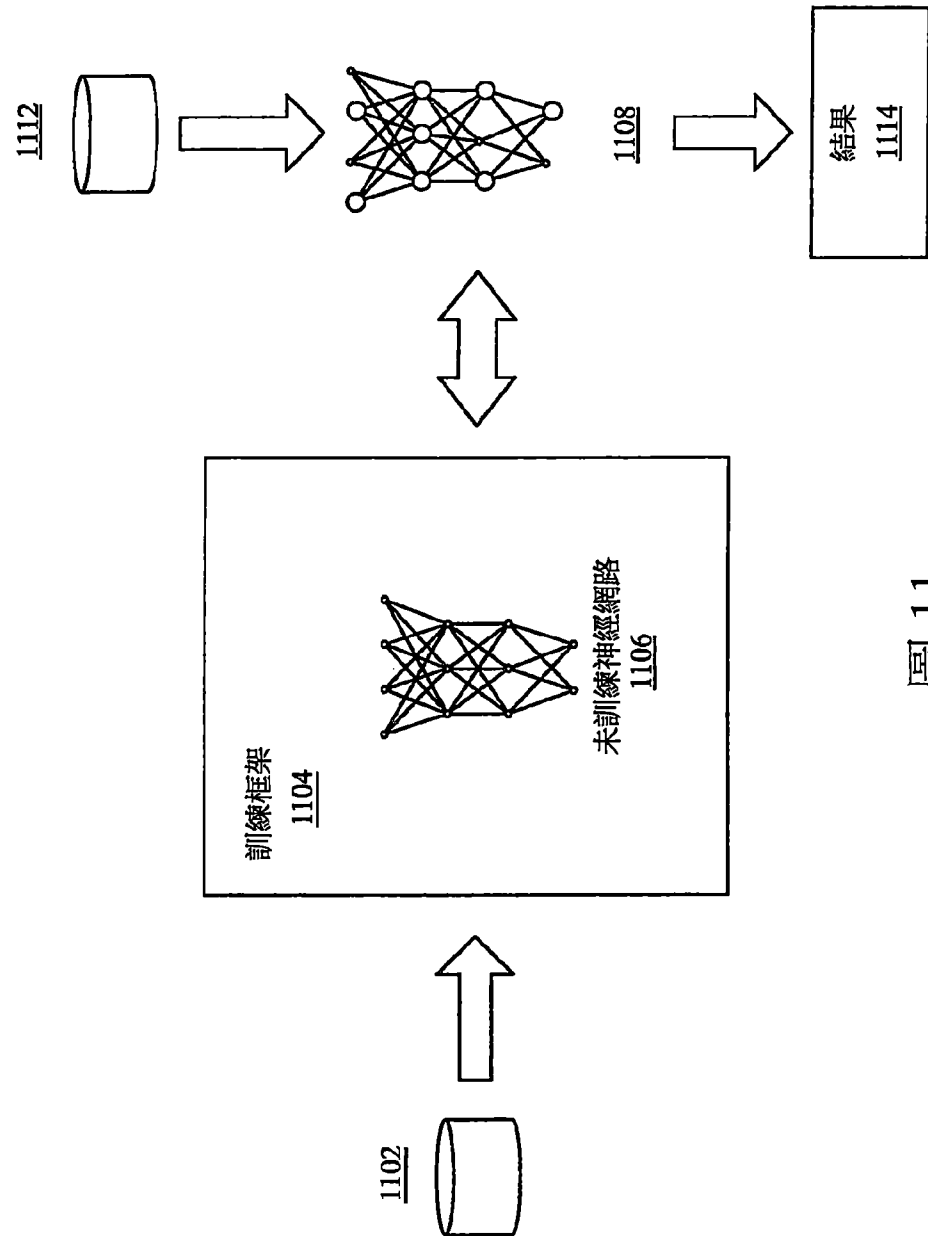


圖 11

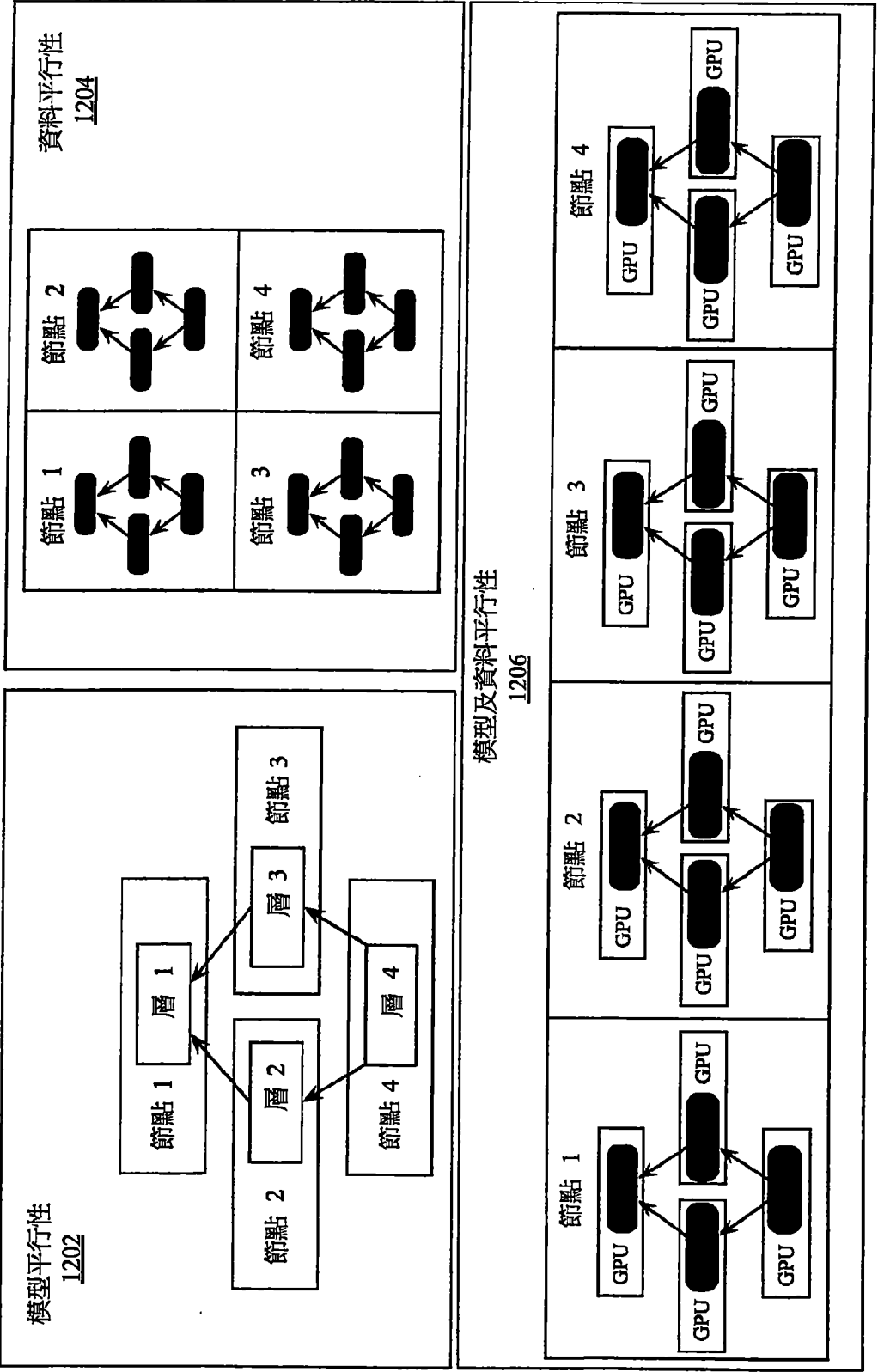


圖 12

1300

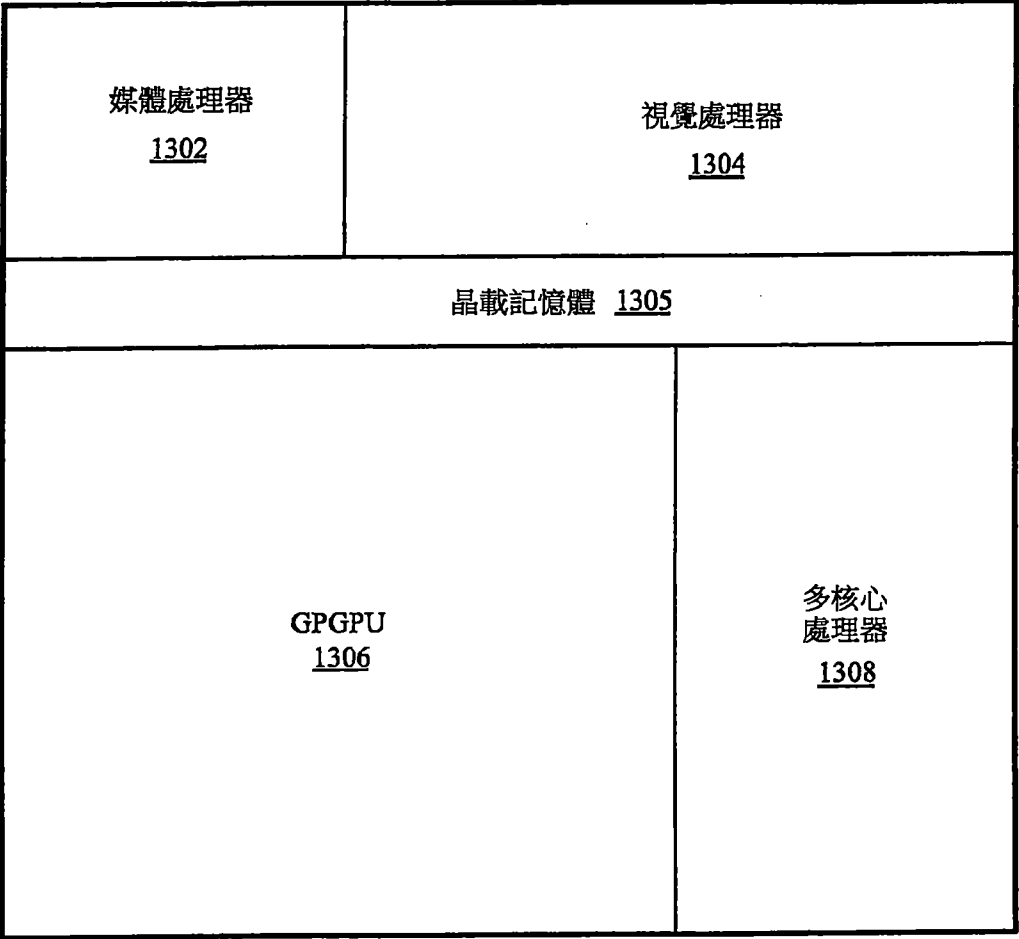


圖 13

1400

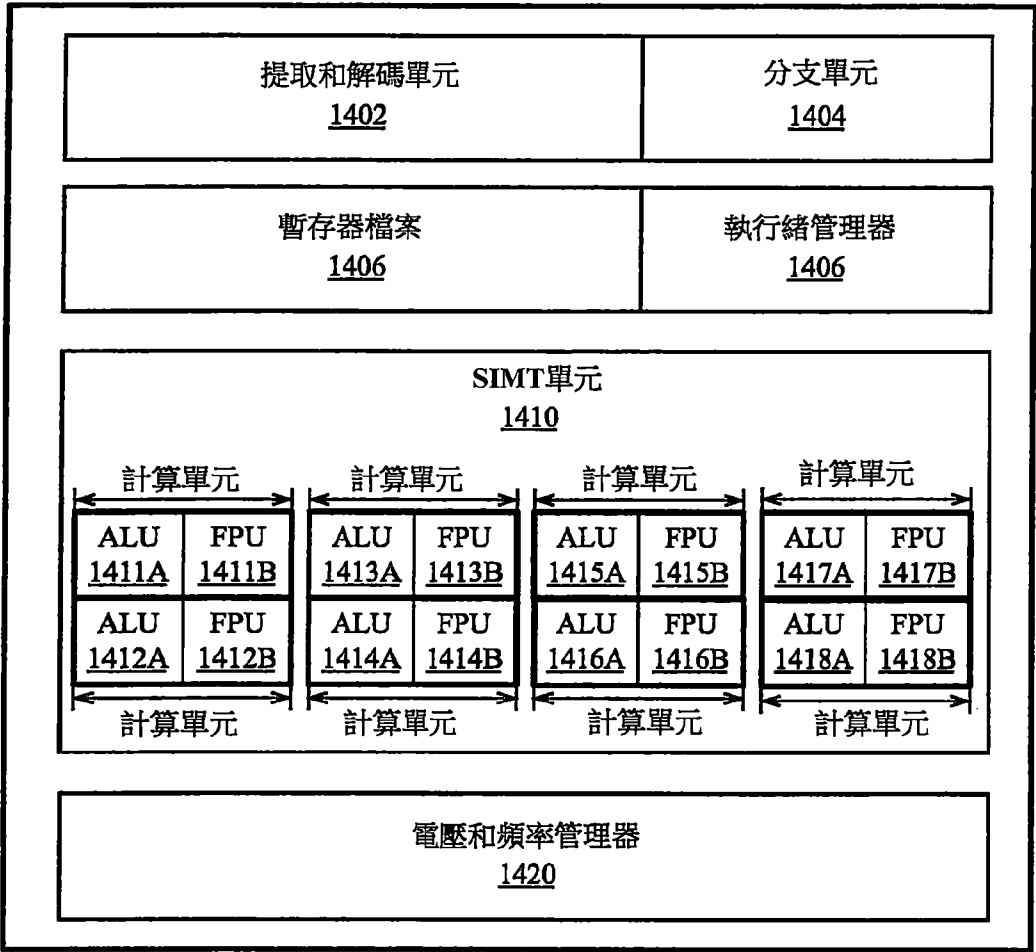


圖 14

1500

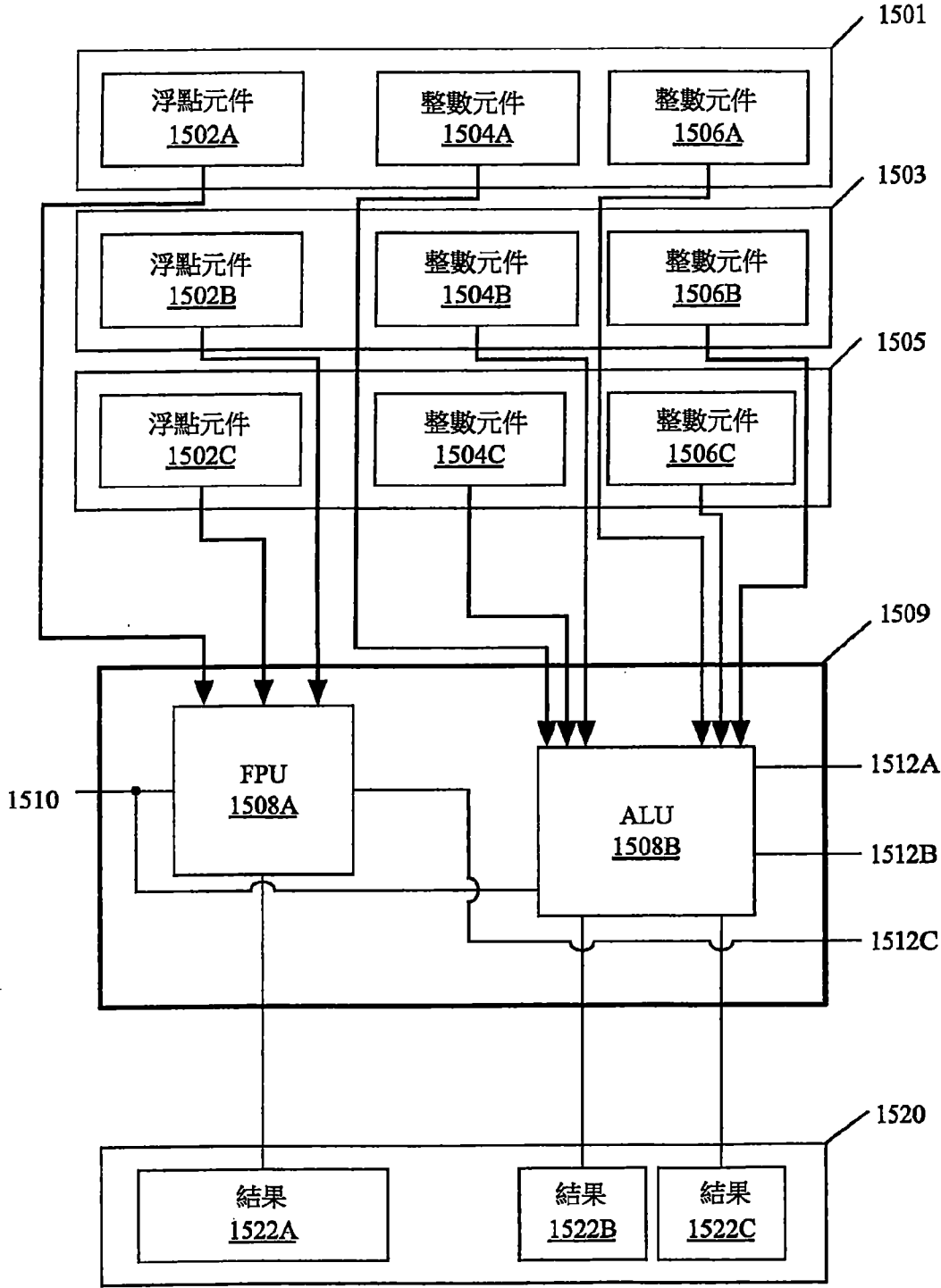


圖 15

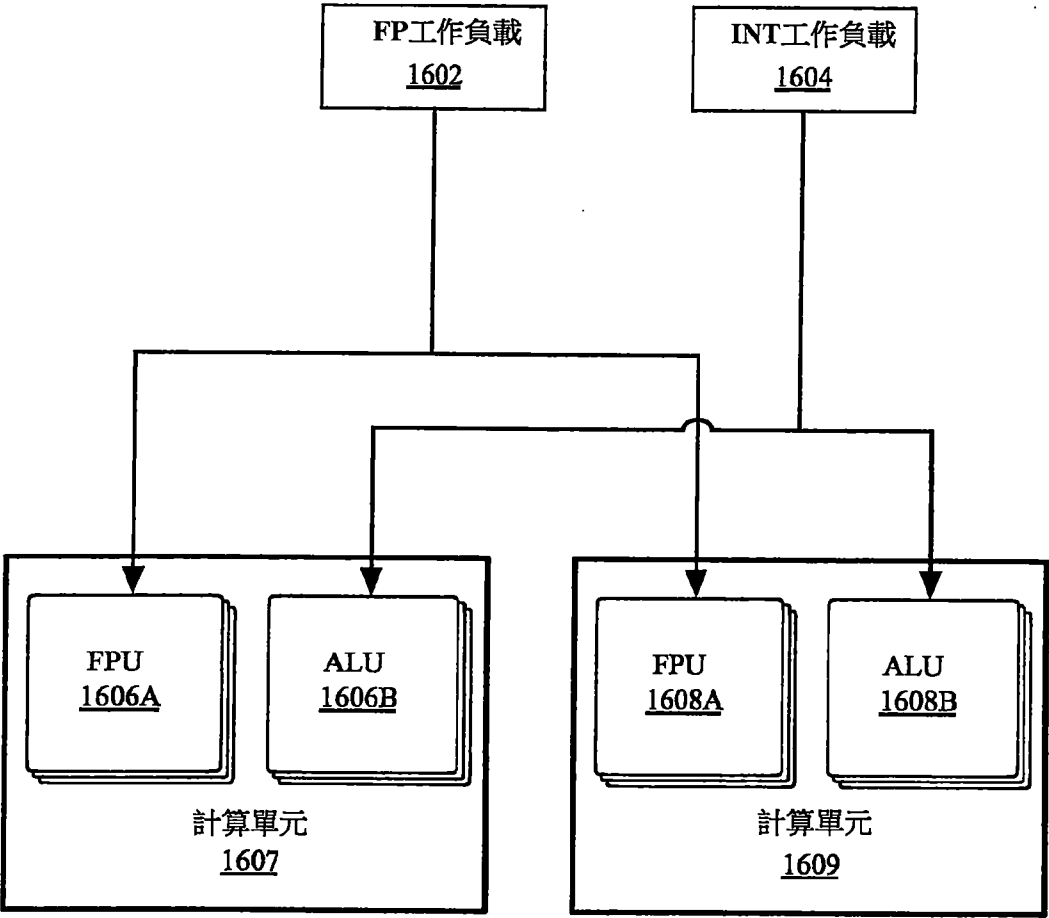


圖 16

1700

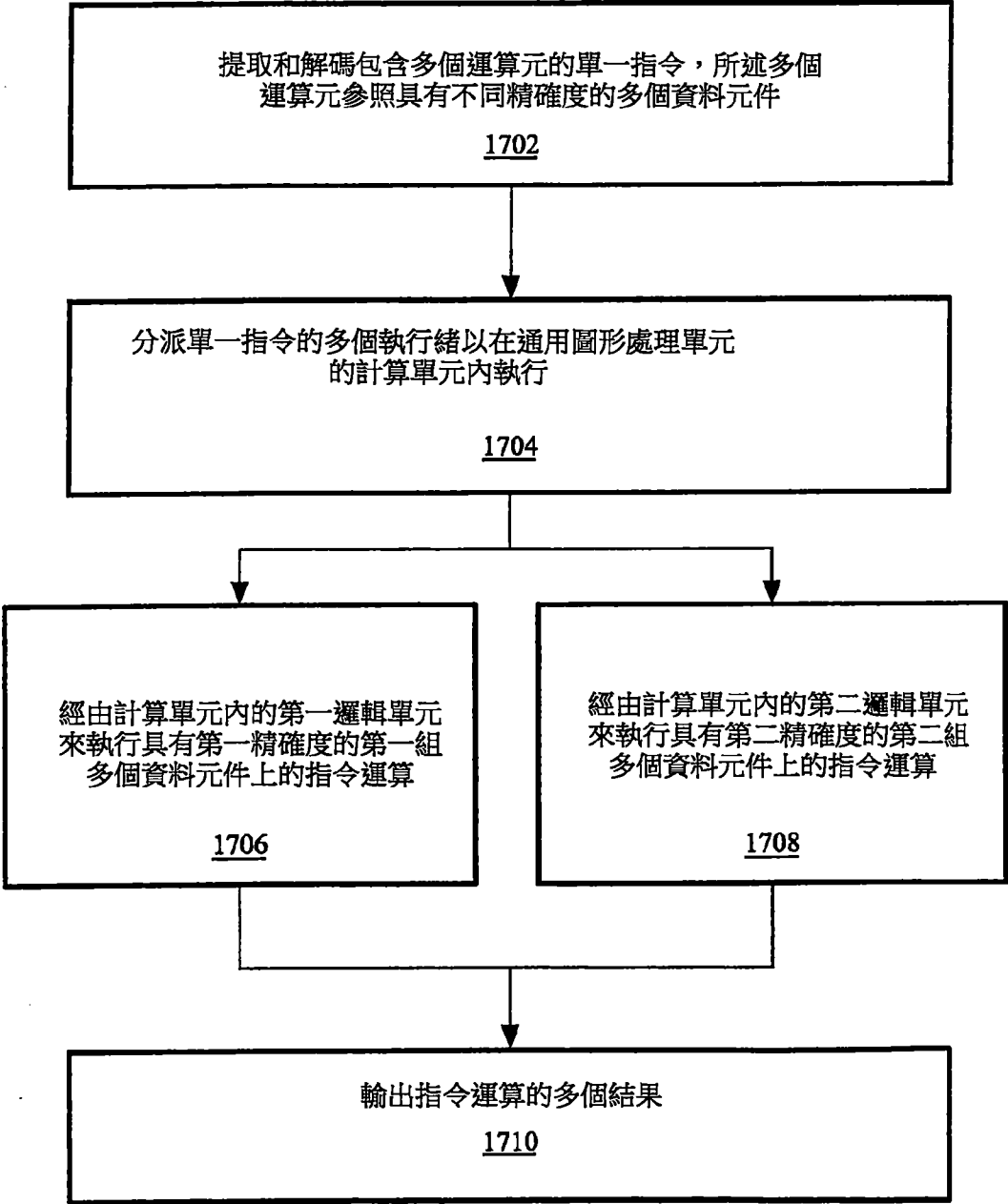


圖 17

1800

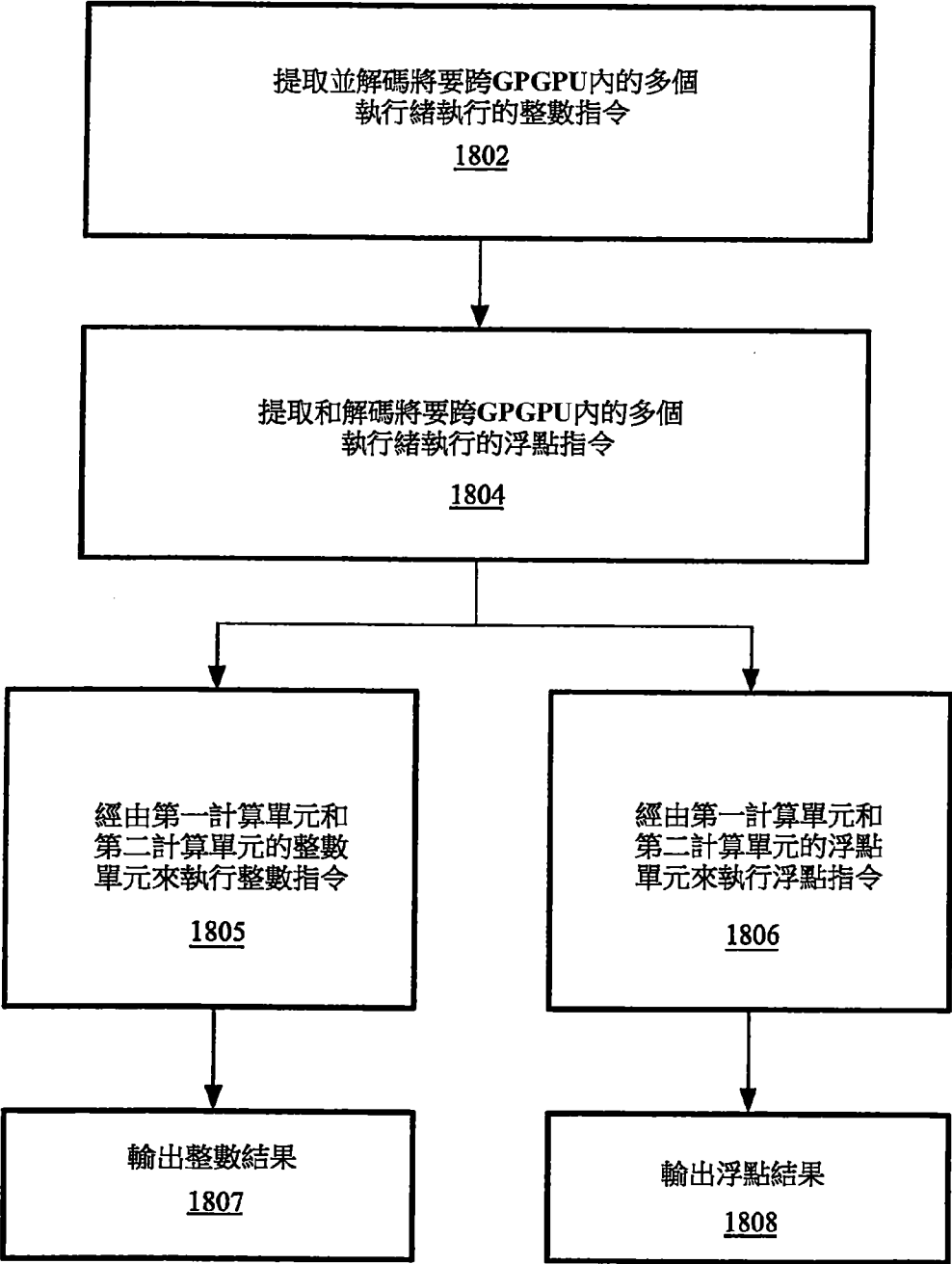


圖 18

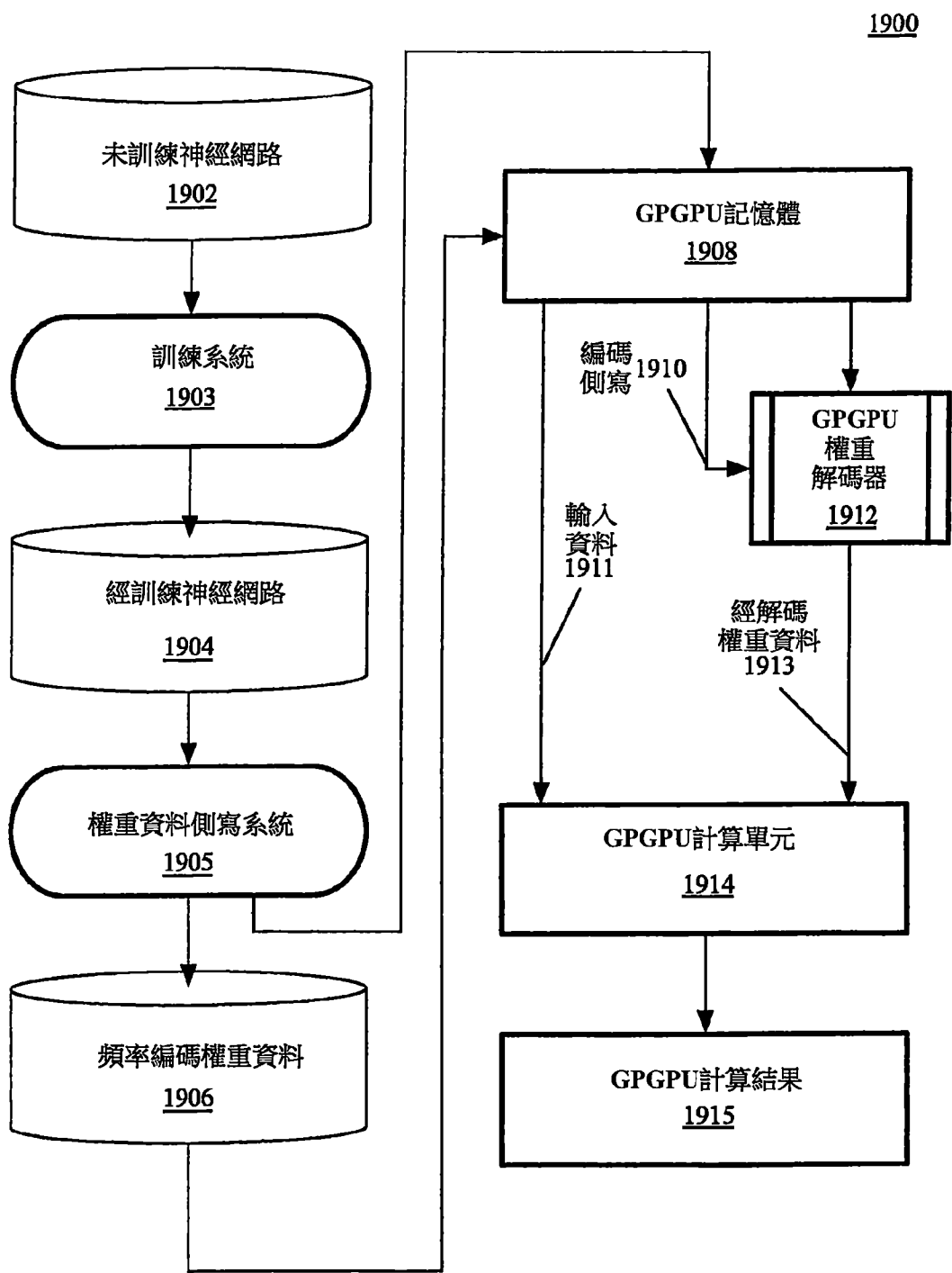


圖 19

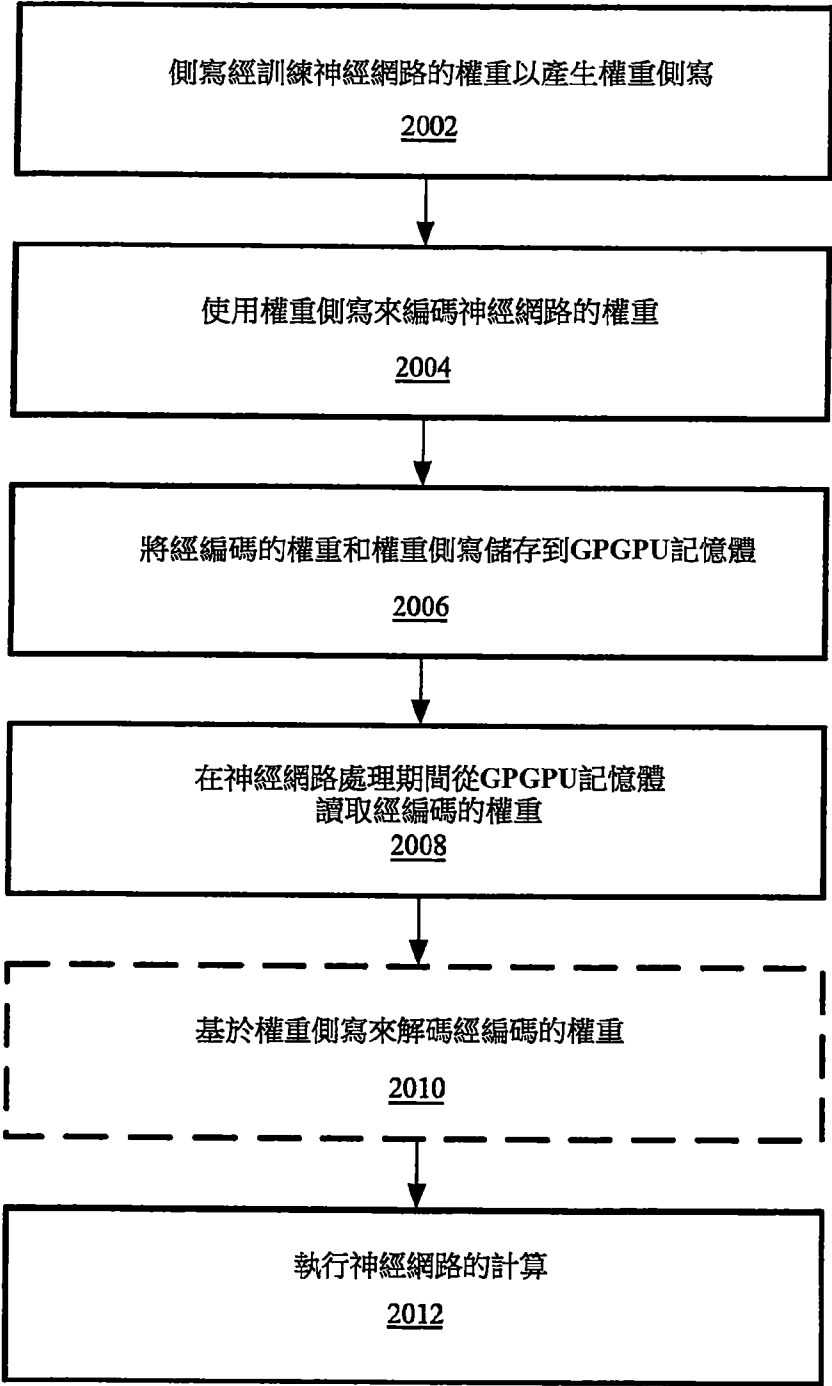


圖 20

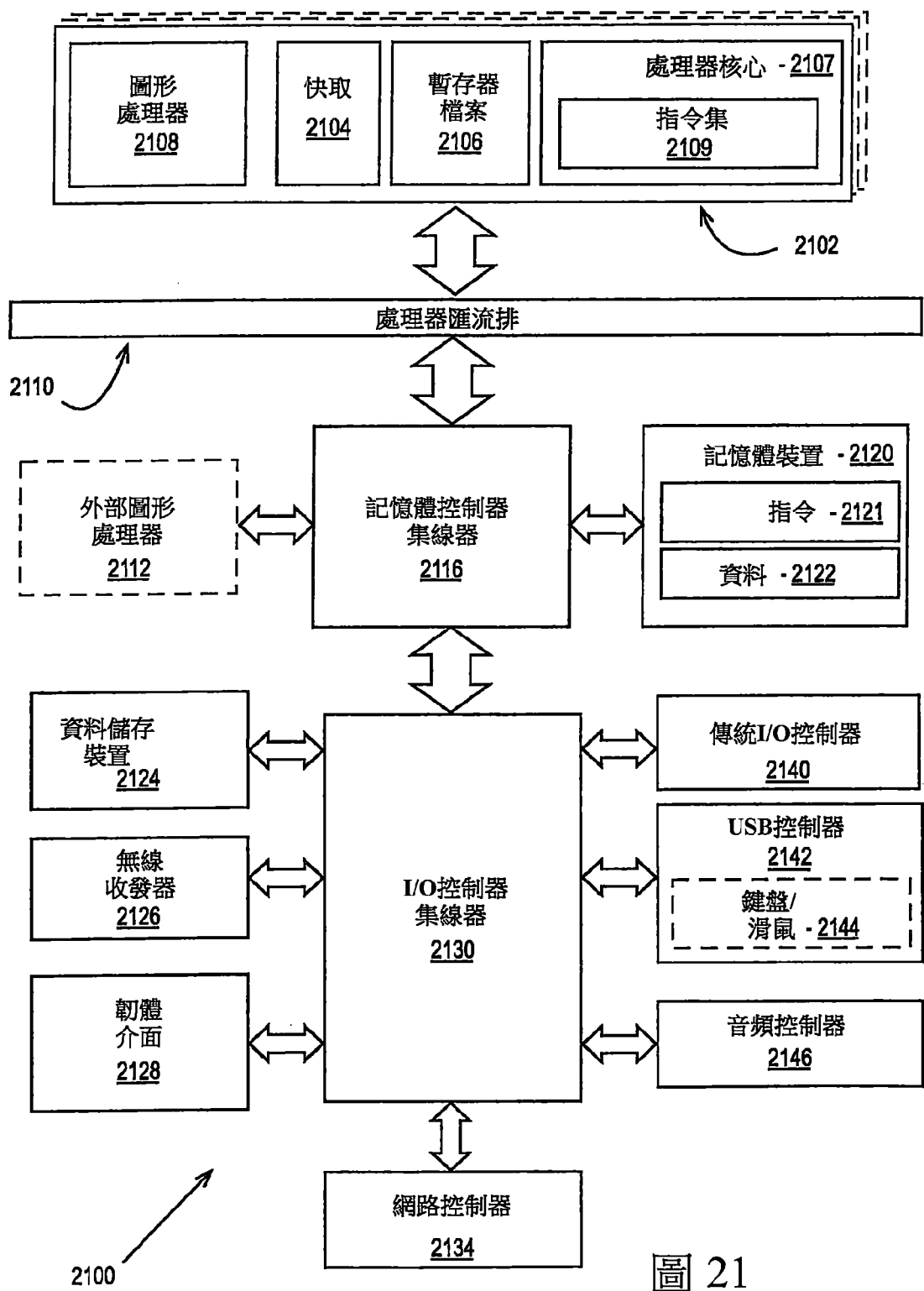


圖 21

2200 ↗

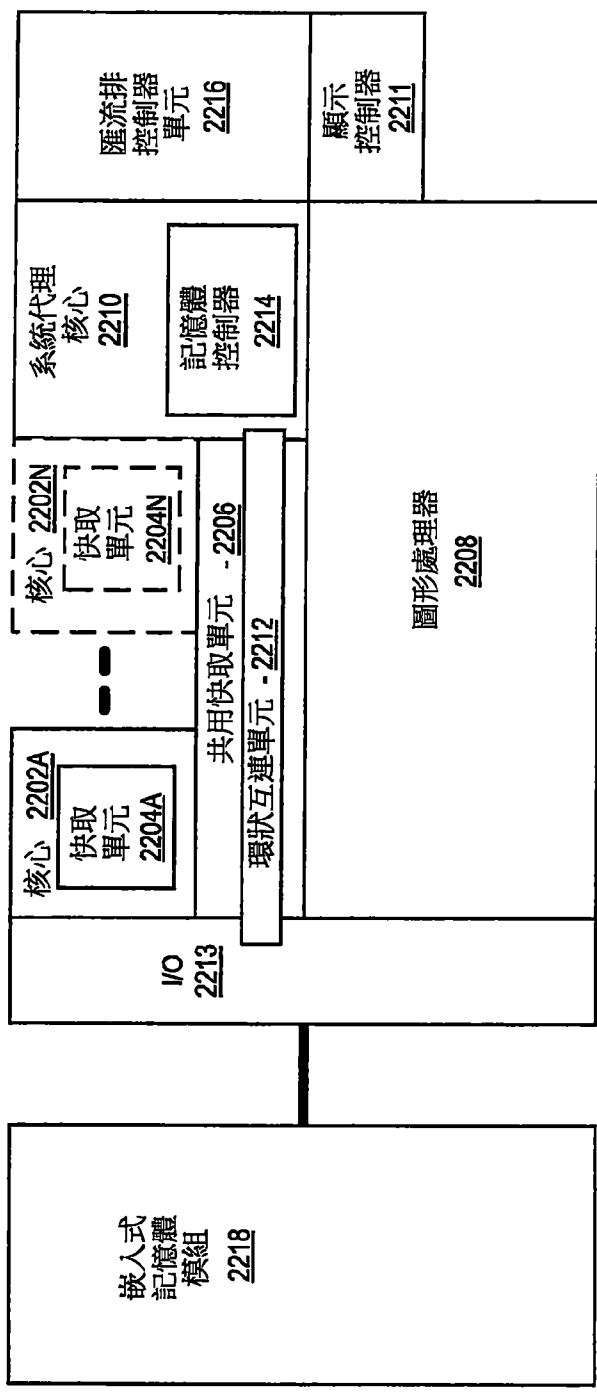


圖 22

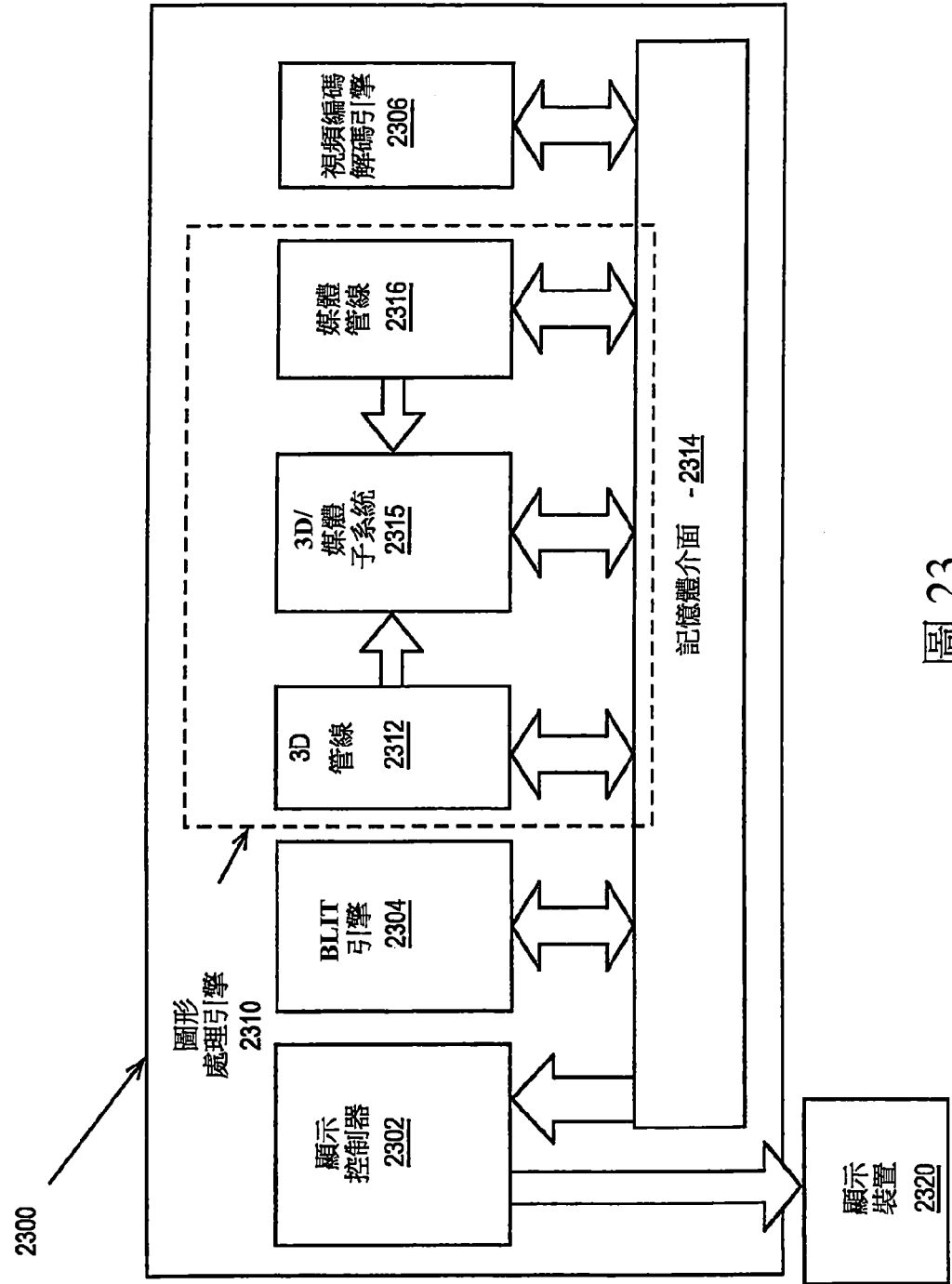


圖 23

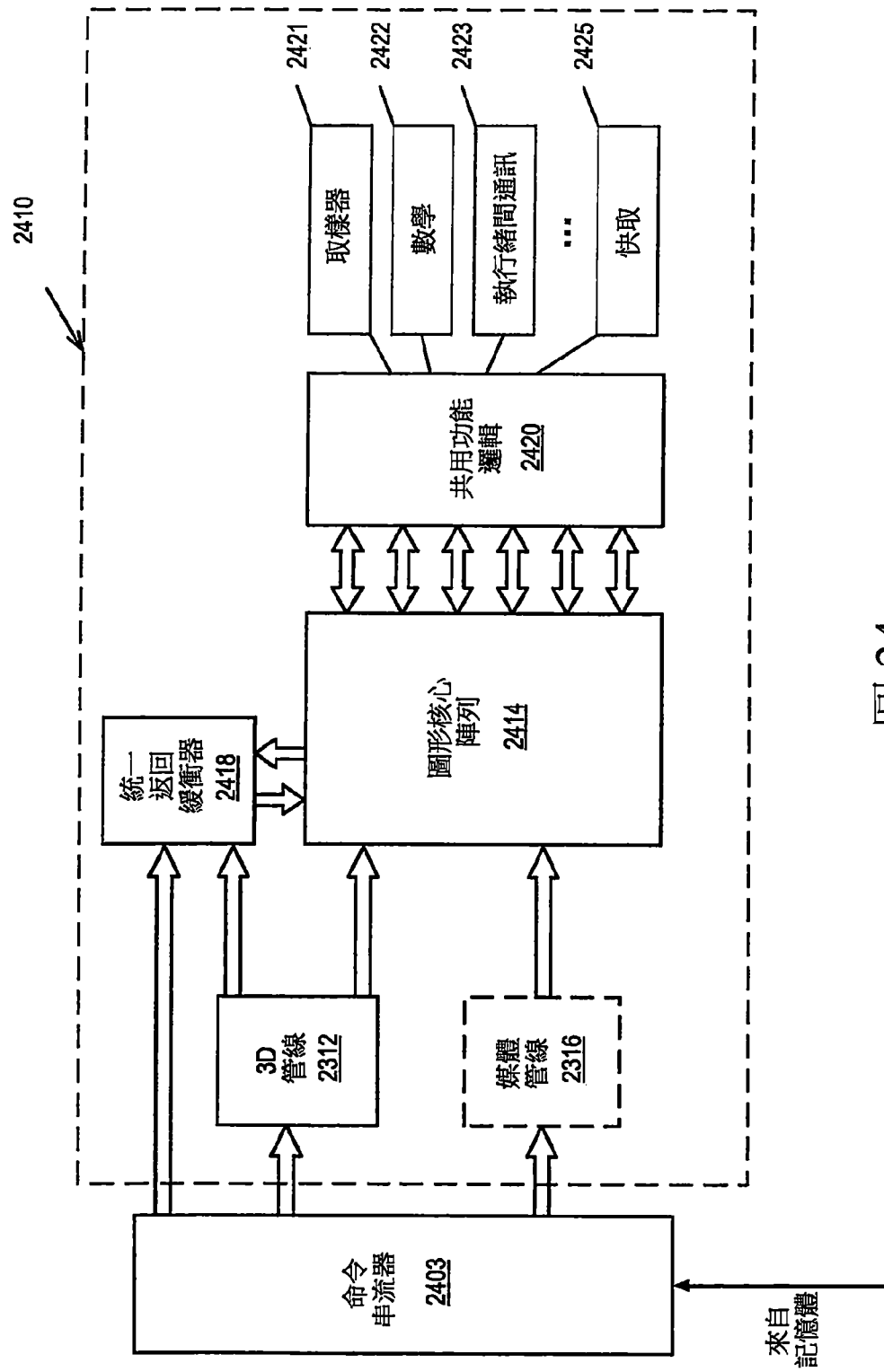


圖 24

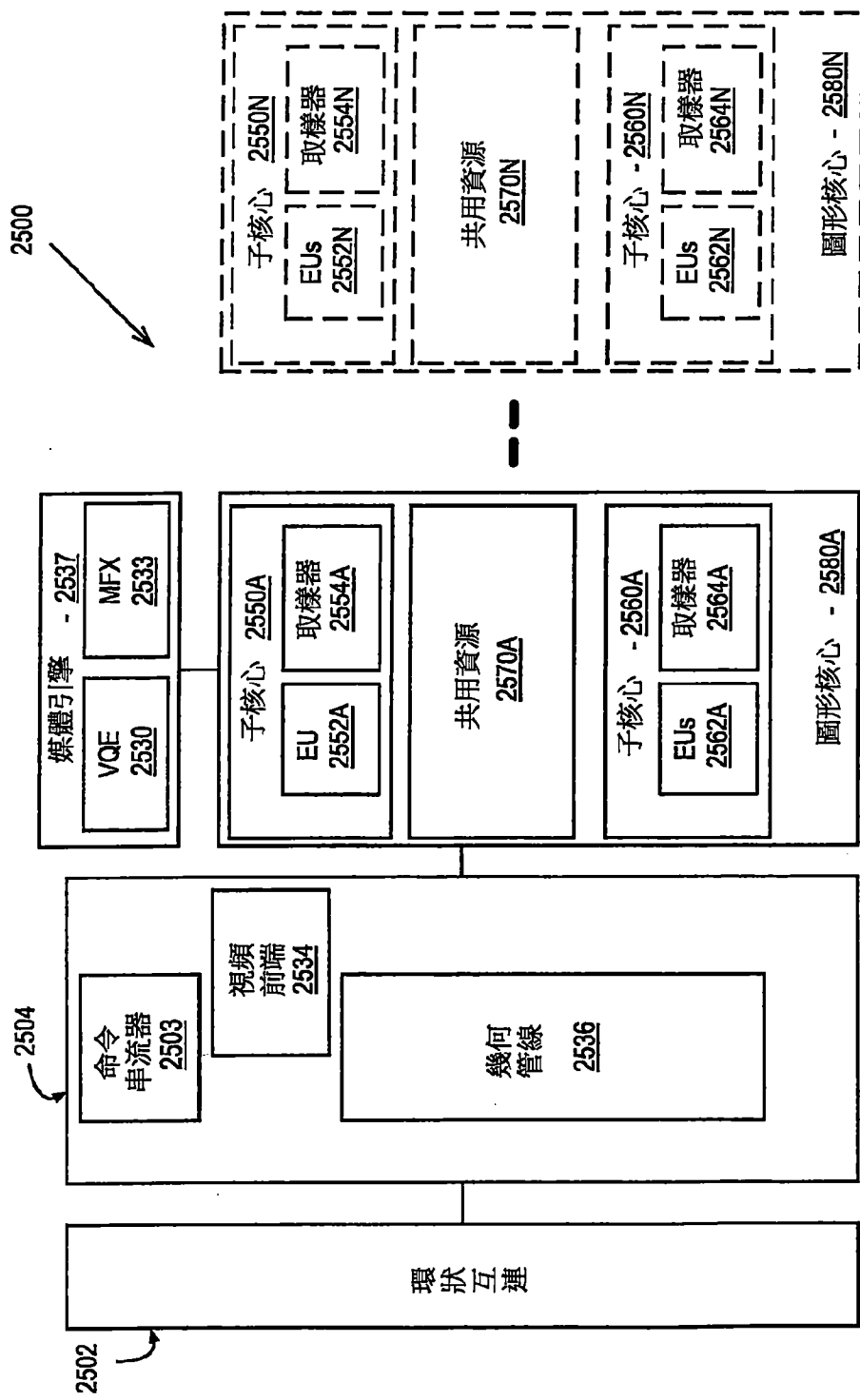


圖 25

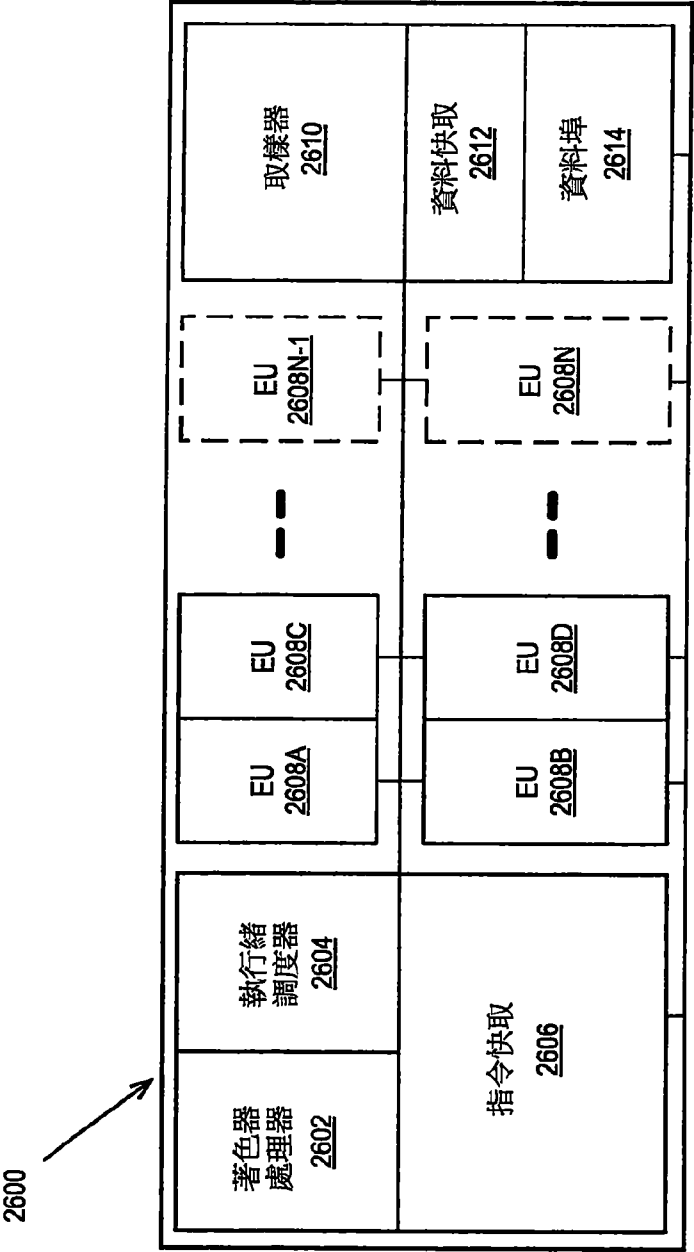


圖 26

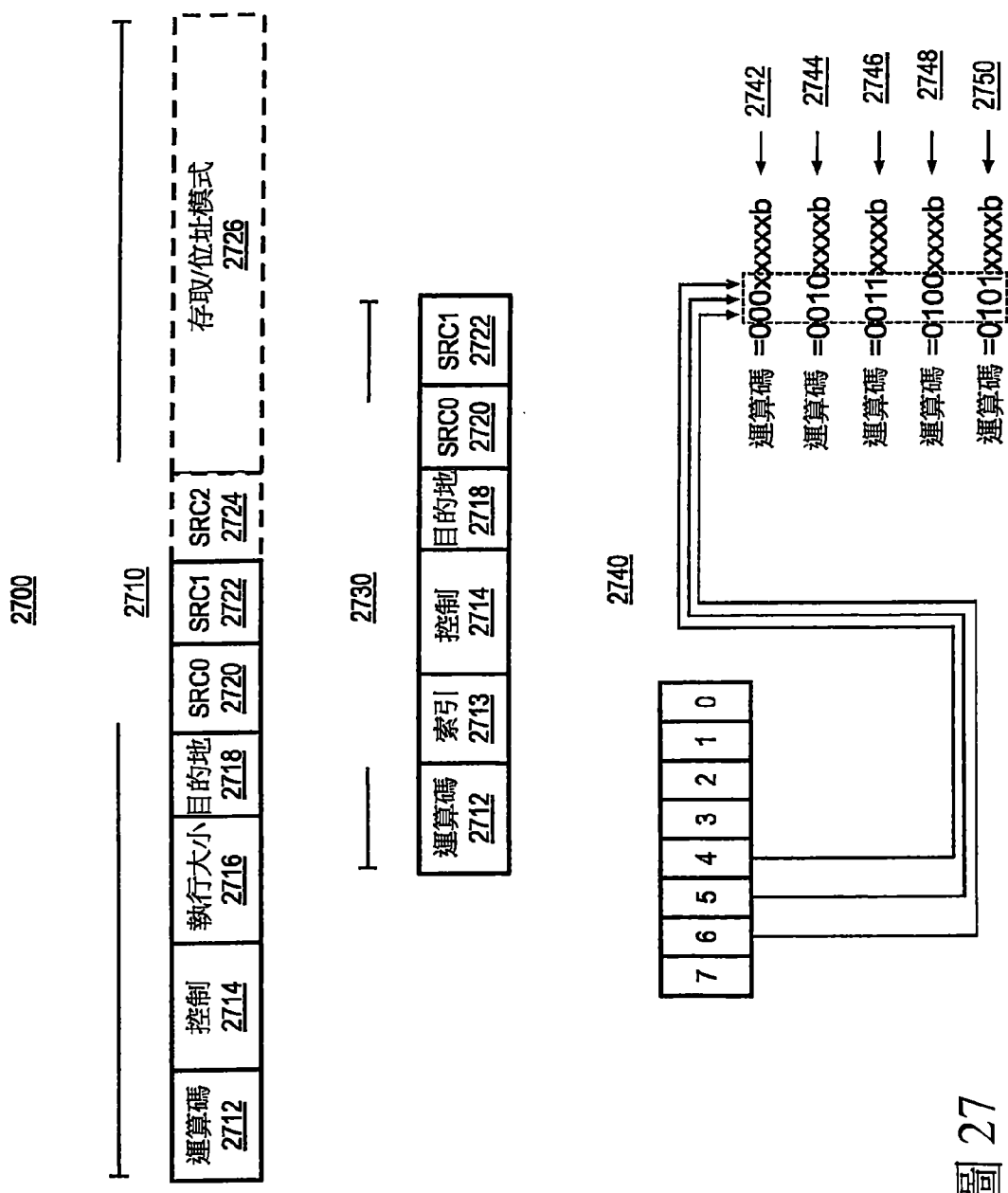


圖 27

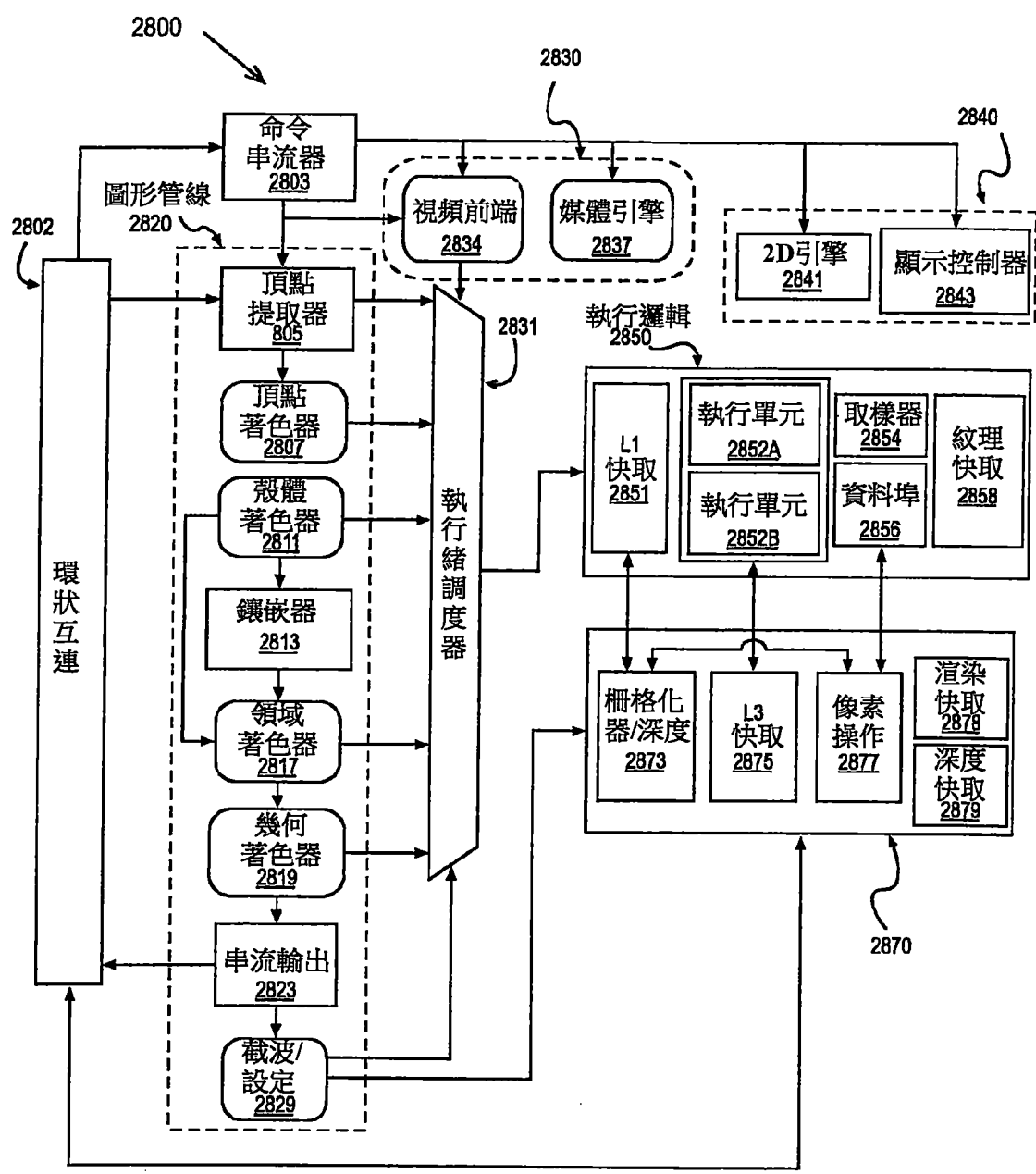


圖 28

圖 29A

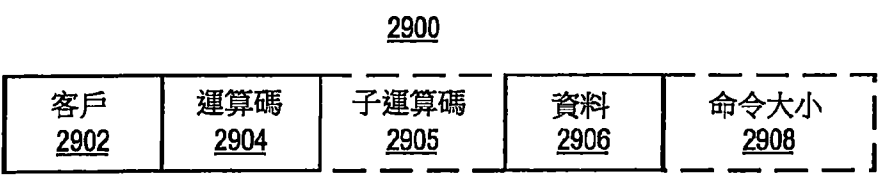
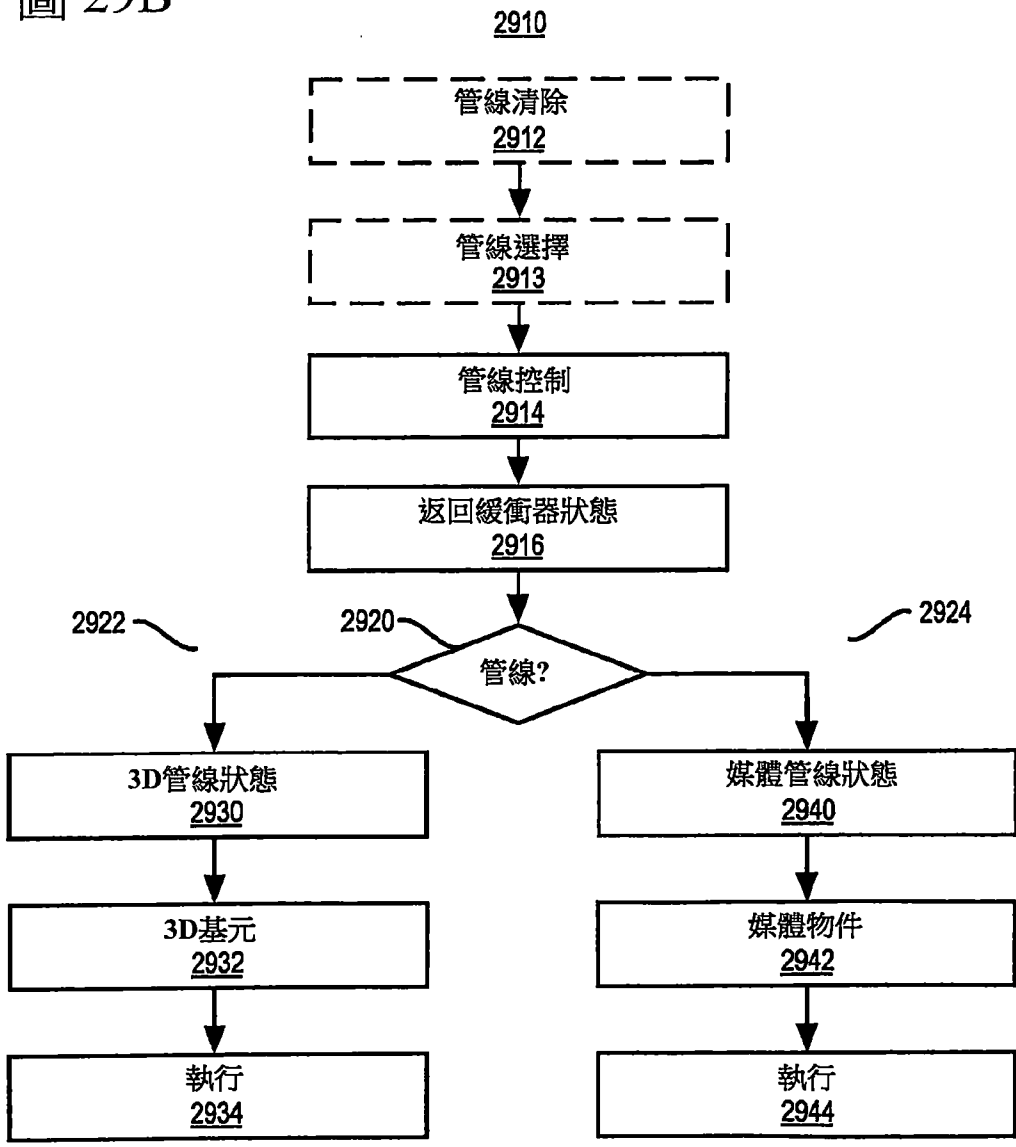


圖 29B



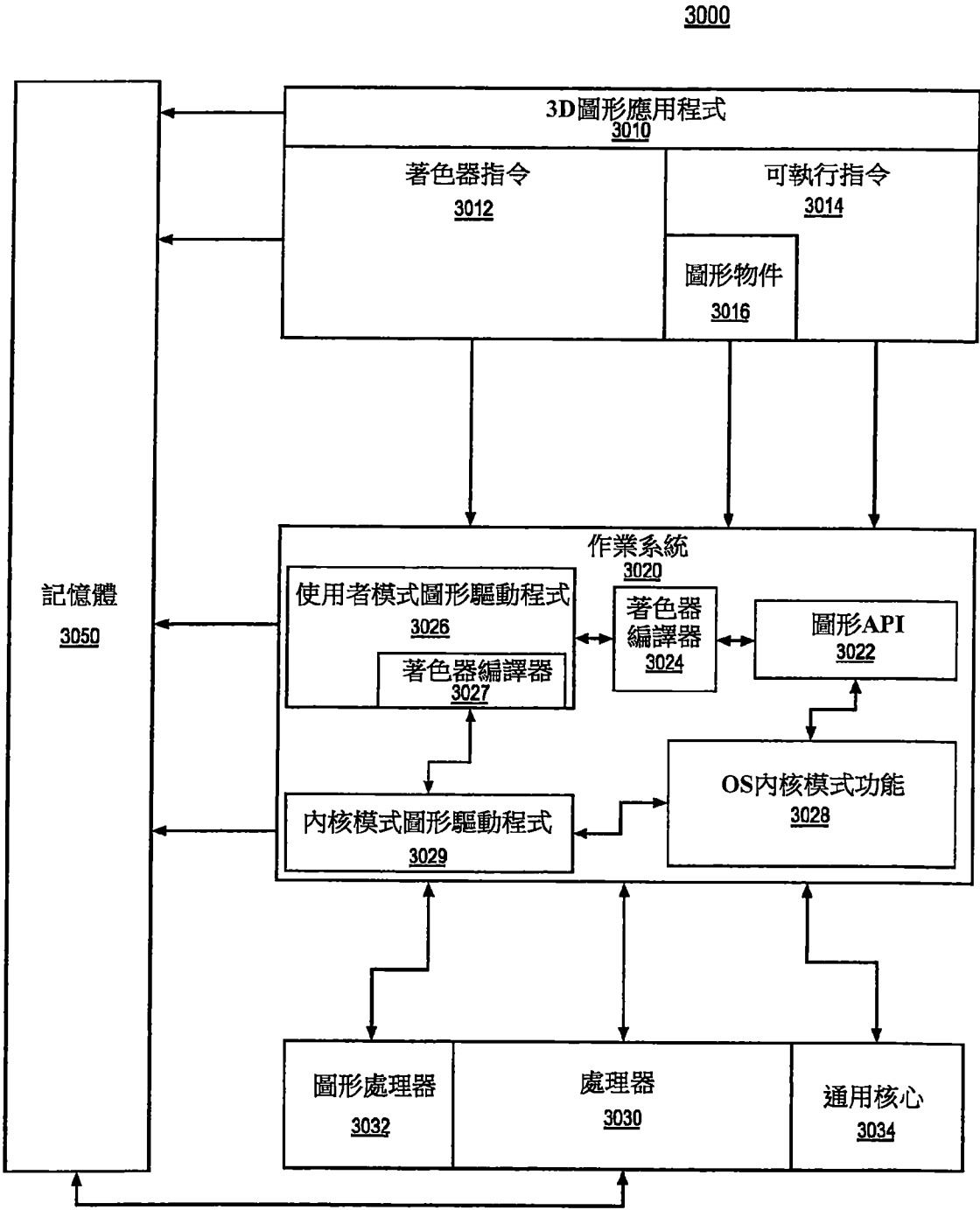


圖 30

3100

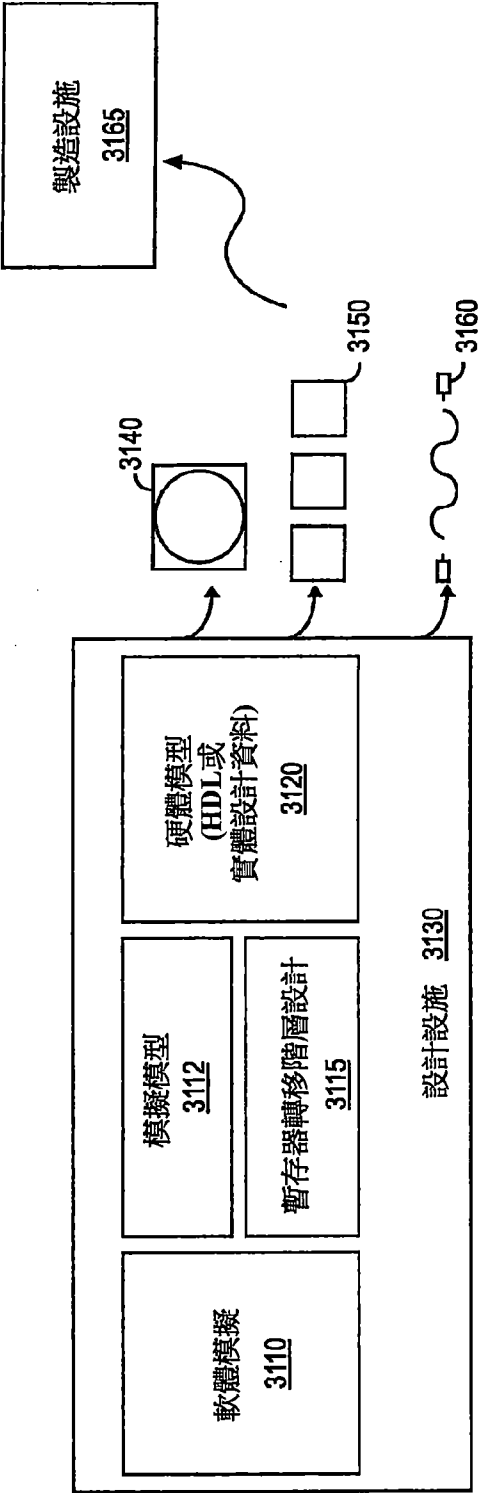


圖 31

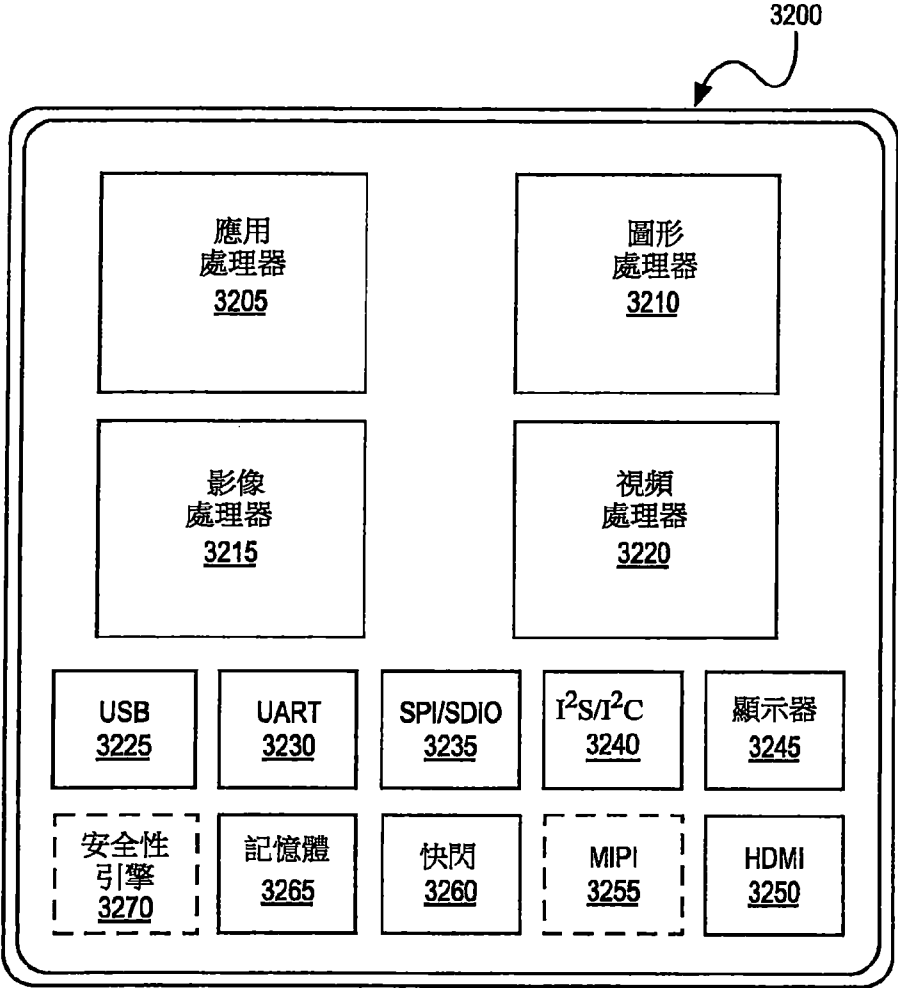


圖 32

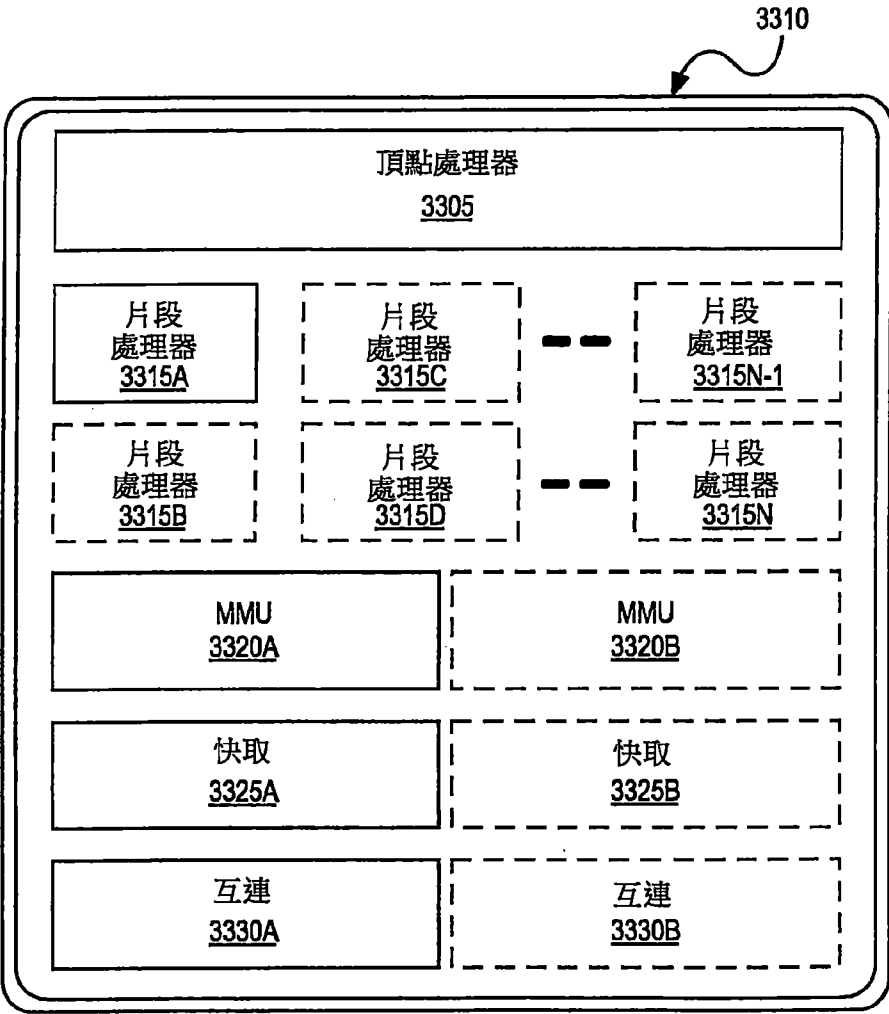


圖 33

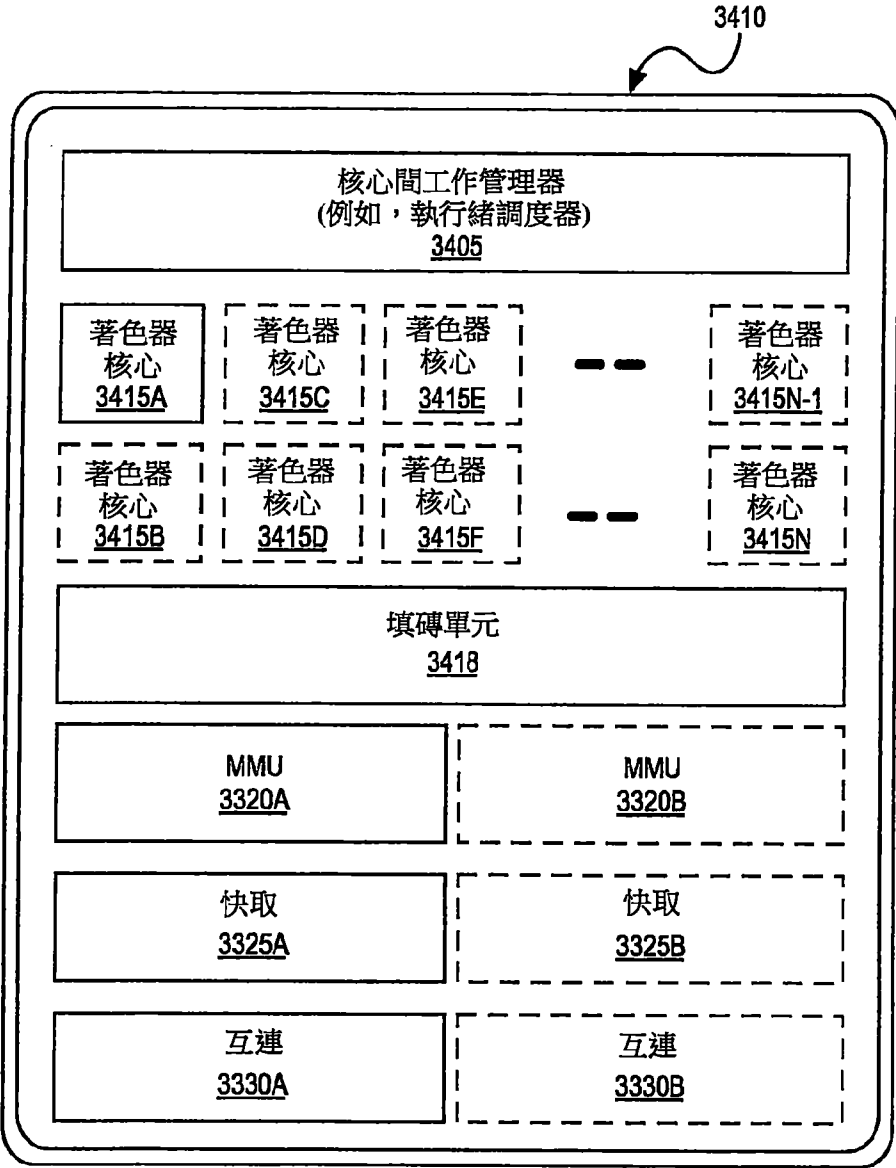


圖 34