



(21)申請案號：101148744 (22)申請日：中華民國 101 (2012) 年 12 月 20 日

(51)Int. Cl. : G06F9/30 (2006.01)

(30)優先權：2011/12/30 世界智慧財產權組織 PCT/US11/68230

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

美國

(72)發明人：賽爾 蘇利曼 SAIR, SULEYMAN (US)；歐德亞麥德維爾 艾蒙斯特阿法 OULD-AHMED-VALL, ELMOUSTAPHA (MR)；楊特 查爾斯 YOUNT, CHARLES

(US)；道西 柯希提傑 DOSHI, KSHITIJ A. (US)；托爾 布萊特 TOLL, BRET (US)

(74)代理人：林志剛

(56)參考文獻：

TW I254260

TW I352512

TW 200935304A

CN 101482810A

US 6781528B1

US 2002/0002666A1

US 2007/0208792A1

審查人員：劉季涵

申請專利範圍項數：20 項 圖式數：15 共 89 頁

(54)名稱

向量頻率壓縮指令

VECTOR FREQUENCY COMPRESS INSTRUCTION

(57)摘要

描述一種處理器核心，其包括用以解碼向量頻率壓縮指令之硬體解碼單元，該向量頻率壓縮指令包括來源運算元和目的地運算元。來源運算元指明一包括複數來源資料元件之來源向量暫存器，該些來源資料元件包括一或更多列相同的資料元件，該些一或更多列相同資料元件之每一者將被壓縮於目的地向量暫存器中而成為一值與列長度對。目的地運算元識別目的地向量暫存器。處理器核心亦包括執行引擎單元，用以執行該解碼的向量頻率壓縮指令，其針對各來源資料元件致使一值被複製入該目的地向量暫存器以指示該來源資料元件之值。一或更多列多來源資料元件同等地被編碼於目的地向量暫存器中而成為預定壓縮值，接續以該列之列長度。

A processor core that includes a hardware decode unit to decode a vector frequency compress instruction that includes a source operand and a destination operand. The source operand specifying a source vector register that includes a plurality of source data elements including one or more runs of identical data elements that are each to be compressed in a destination vector register as a value and run length pair. The destination operand identifies the destination vector register. The processor core also includes an execution engine unit to execute the decoded vector frequency compress instruction which causes, for each source data element, a value to be copied into the destination vector register to indicate that source data element's value. One or more runs of the source data elements equal are encoded in the destination vector register as the predetermined compression value followed by a run length for that run.

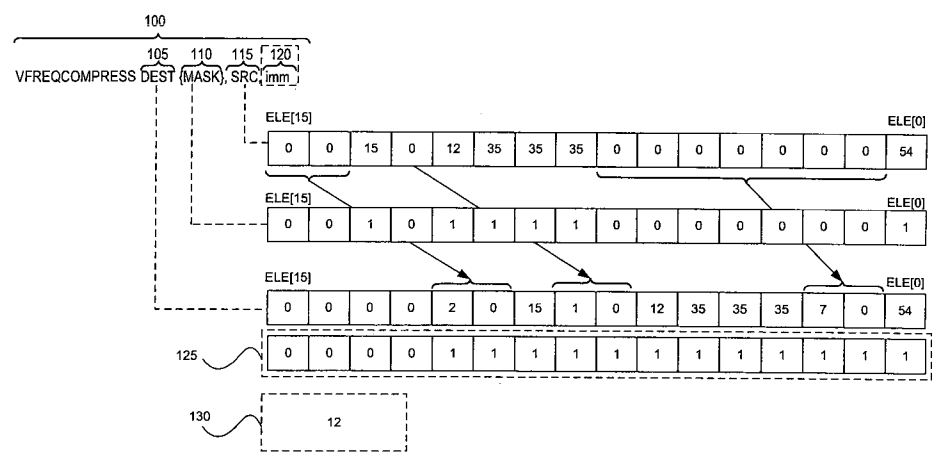


圖1

- 100 . . . 向量頻率壓縮指令
- 105 . . . 目的地運算元
- 110 . . . 向量遮罩 MASK
- 115 . . . 來源運算元
- 120 . . . 選擇性運算元
- 125 . . . 已使用的元件控制遮罩
- 130 . . . 已使用的元件計數

發明專利說明書

(本申請書格式、順序，請勿任意更動，※記號部分請勿填寫)

※申請案號：101148744

※申請日：101年12月20日

※IPC分類：

G06F 9/30

(2006.01)

一、發明名稱：(中文/英文)

向量頻率壓縮指令

Vector frequency compress instruction

二、中文發明摘要：

描述一種處理器核心，其包括用以解碼向量頻率壓縮指令之硬體解碼單元，該向量頻率壓縮指令包括來源運算元和目的地運算元。來源運算元指明一包括複數來源資料元件之來源向量暫存器，該些來源資料元件包括一或更多列相同的資料元件，該些一或更多列相同資料元件之每一者將被壓縮於目的地向量暫存器中而成為一值與列長度對。目的地運算元識別目的地向量暫存器。處理器核心亦包括執行引擎單元，用以執行該解碼的向量頻率壓縮指令，其針對各來源資料元件致使一值被複製入該目的地向量暫存器以指示該來源資料元件之值。一或更多列多來源資料元件同等地被編碼於目的地向量暫存器中而成為預定壓縮值，接續以該列之列長度。

三、英文發明摘要：

A processor core that includes a hardware decode unit to decode a vector frequency compress instruction that includes a source operand and a destination operand. The source operand specifying a source vector register that includes a plurality of source data elements including one or more runs of identical data elements that are each to be compressed in a destination vector register as a value and run length pair. The destination operand identifies the destination vector register. The processor core also includes an execution engine unit to execute the decoded vector frequency compress instruction which causes, for each source data element, a value to be copied into the destination vector register to indicate that source data element's value. One or more runs of the source data elements equal are encoded in the destination vector register as the predetermined compression value followed by a run length for that run.

四、指定代表圖：

(一) 本案指定代表圖為：第(1)圖。

(二) 本代表圖之元件符號簡單說明：

100：向量頻率壓縮指令

105：目的地運算元

110：向量遮罩 MASK

115：來源運算元

120：選擇性運算元

125：已使用的元件控制遮罩

130：已使用的元件計數

五、本案若有化學式時，請揭示最能顯示發明特徵的化學式：無

六、發明說明：

【發明所屬之技術領域】

本發明之領域一般係有關電腦處理器架構，而更明確地，係有關向量頻率壓縮指令。

【先前技術】

指令集，或指令集架構（ISA），為關於編程之電腦架構的部分，並可包括本機資料類型、指令、暫存器架構、定址模式、記憶體架構、中斷和例外處置、及外部輸入和輸出（I/O）。應注意術語「指令」一般於文中指的是巨集指令 - 其為提供給處理器以供執行 - 相對於微指令或微操作（micro-ops） - 其係得自處理器之解碼器解碼巨集指令。指令集架構係不同於微架構，其為實施 ISA 之處理器的內部設計。具有不同微架構之處理器可共用一共同指令集。

指令集包括一或更多指令格式。既定的指令格式係界定各種欄位（位元之數目、位元之位置），以指明（除了別的以外）將履行之操作及該操作所將履行之運算元。既定指令係使用既定指令格式來表達並指明操作及運算元。指令流為指令之一特定序列，其中該序列中之各指令為一指令格式之指令的發生。

科學的、金融的、自動向量化的一般用途；RMS（識別、探勘、及合成），及視覺和多媒體應用（例如，2D/3D 圖形、影像處理、視頻壓縮/解壓縮、聲音識別演算

法及音頻調處) 經常需要對大量資料項目履行相同的操作(稱之為「資料平行化」)。單一指令多重資料(SIMD)指的是一種致使處理器對多重資料項目履行操作之指令的類型。SIMD 技術特別適於其可將暫存器中之位元邏輯地劃分為數個固定大小的資料元件之處理器，每一資料元件代表分離的值。例如，64 位元暫存器中之位元可被指明為來源運算元以便操作為四個分離的 16 位元資料元件，其每個代表一分離的 16 位元值。舉另一例，256 位元暫存器中之位元可被指明為來源運算元以便操作為四個分離的 64 位元緊縮資料元件(四字元(Q)大小資料元件)、八個分離的 32 位元緊縮資料元件(雙字元(D)大小資料元件)、十六個分離的 16 位元緊縮資料元件(字元(W)大小資料元件)、或三十二個分離的 8 位元緊縮資料元件(位元組(B)大小資料元件)。此資料之類型被稱為緊縮資料類型或向量資料類型，而此資料類型之運算元被稱為緊縮資料運算元或向量運算元。換言之，緊縮資料項目或向量指的是緊縮資料元件之序列，而緊縮資料運算元或向量運算元為 SIMD 指令之來源或目的地運算元(亦稱為緊縮資料指令或向量指令)。

舉例而言，SIMD 指令之一類型係指明單一向量操作以供用垂直方式履行於兩來源向量運算元來產生相同大小的目的地向量運算元(亦稱為結果向量運算元)，具有相同的資料元件數，且依相同的資料元件順序。來源向量運算元中之資料元件被稱為來源資料元件，而目的地向量運

算元中之資料元件被稱為目的地或結果資料元件。這些來源向量運算元為相同大小且含有相同寬度的資料元件，而因此其含有相同的資料元件數。兩來源向量運算元中之相同位元位置中的來源資料元件形成資料元件對（亦稱為相應的資料元件；亦即，各來源運算元之資料元件位置 0 中的資料元件相應、各來源運算元之資料元件位置 1 中的資料元件相應，依此類推）。由 SIMD 指令所指明之操作被分離地履行於這些來源資料元件對之每一者，以產生匹配的結果資料元件數，而因此各來源資料元件對具有一相應的結果資料元件。因為操作是垂直的且因為結果向量運算元為相同的大小、具有相同的資料元件數、且結果資料元件依相同的資料元件順序被儲存為來源向量運算元，所以結果資料元件係位於如來源向量運算元中之其相應的來源資料元件對之結果向量運算元的相同位元位置中。除了 SIMD 指令之此範例類型外，有多種 SIMD 指令之其他類型（例如，僅具有一個或具有大於二個來源向量運算元者、以水平方式操作者、產生其為不同大小的結果向量運算元者、具有不同大小的資料元件者、及/或具有不同的資料元件順序者）。應理解術語「目的地向量運算元」（或目的地運算元）被定義為履行由指令所指明之操作的直接結果，包括將該目的地運算元儲存於某一位置（可為一暫存器或者於該指令所指明之記憶體位址上）以致其可由另一指令存取為來源運算元（藉由另一指令之該相同位置的指明）。

某些指令集架構容許多個向量及純量操作平行地完成，並更新指令集架構暫存器集。這些指令集架構可被制衡以實施壓縮/解壓縮指令及演算法，諸如基於運作長度編碼（RLE）之指令。

RLE 是一種無損的資料壓縮形式，其中資料流中之資料的序列被壓縮在當那些序列含有一或更多組連續資料值時。取代將各資料元件儲存於該組連續資料值中，具有該值之單一元件被儲存，接續以一具有連續元件之計數的元件。此壓縮之形式在含有許多此類運行之資料上是最有用的。

例如，零為基的壓縮/解壓縮利用資料流中之經常發生的零。於某些資料類型中，特別是與圖形處理相關的資料，常有資料之顯著部分含有零之值，而（因此）有許多零之運行。基於零之 RLE 常被稱為零為基的壓縮。雖然其他的 RLE 技術可基於零以外的值，假如選擇其他 RLE 值將使壓縮受益的話。

【發明內容及實施方式】

於下列描述中，提出各種特定的細節。然而，應理解本發明之實施例可被實行而無需這些特定細節。於其他例子中，眾所周知的電路、結構及技術未被詳細地顯示以免混淆本描述之理解。

說明書中對於「一實施例」、「實施例」、「範例實施例」等參考係指示所描述之實施例可包括特定的特徵、

結構、或特性，但每一實施例可不一定包括該特定的特徵、結構、或特性。再者，此類用詞不一定指稱相同的實施例。此外，當配合一實施例而描述特定的特徵、結構、或特性時，認定其係落入熟悉本項技術人士之知識內，以影響配合其他實施例之此類特徵、結構、或特性（無論是否明確地描述）。

圖 1 顯示依據一實施例之向量頻率壓縮指令的範例執行。向量頻率壓縮指令係實施 RLE 以致一組來源資料元件係根據該組來源資料元件中所發生之預定值的順序出現而被壓縮。例如，在該組來源資料元件中之一列七個零可被編碼為零接著七以指示零之數目。

向量頻率壓縮指令 100 包括目的地運算元 105 及來源運算元 115。向量頻率壓縮指令 100 屬於指令集架構，且一指令流內之指令 100 的每次「發生」將包括目的地運算元 105 及來源運算元 115 內之值。於此範例中，目的地運算元 105 和來源運算元 115 兩者為向量暫存器（諸如 128、256、512 位元暫存器）。向量暫存器可為具有 16 個 32 位元資料元件之 `zmm` 暫存器，然而，亦可使用其他的資料元件及暫存器大小，諸如 `xmm` 或 `ymm` 暫存器及 16 或 64 位元資料元件。因此，來源運算元 115 及目的地運算元 105 被顯示以 16 個資料元件，針對第一元件標示為 `ele[0]`，使用 0 指標標示法，其中第一元件位於 0 位置。各運算元之最後元件則被標示為 `ele[15]`。

由來源運算元所指明之來源向量暫存器的內容包括多

個來源資料元件。如圖 1 中所示，於指標 0 上之來源資料元件含有值 54。於指標 1-7 上之來源資料元件含有值 0。於指標 8-10 上之來源資料元件含有值 35。於指標 11 上之來源資料元件含有值 12，於指標 12 上之來源資料元件含有值 0，於指標 13 上之來源資料元件含有值 15，及於指標 14 和 15 上之來源資料元件含有值 0。

向量頻率壓縮指令 100 被顯示以壓縮含有值 0 之資料元件的發生。然而，可選擇地，向量頻率壓縮指令 100 可被實施以壓縮含有其他值之資料元件的發生。有選擇性運算元 120 顯示為即刻值，其可被編碼以應被壓縮之值。因此，120 將代表應被壓縮之值。再者，向量頻率壓縮指令 100 被編碼以向量遮罩 MASK 110，其指明哪些資料元件吻合應被壓縮之值以及哪些資料元件不吻合應被壓縮之值。MASK 110 根據指令之操作而含有零或者一。於所示之實施例中，MASK 110 含有一個一在相應於其不吻合應被壓縮之值的來源資料元件之元件中、及一個零在那些吻合應被壓縮之值（如該狀況可為零）的元件中。因此，MASK 110 含有一個零於遮罩元件 1-7、12、14 及 15 中。

向量頻率壓縮指令 100 之操作可被描述以下列方式。當在 MASK 110 中遭遇一連串的零時，表示一系列吻合值出現在來源資料元件中，則指令便計算 MASK 110 中之連續零的數目並將零之值插入其接續著下個位置中之計數的目的地 105 向量之當前位置中。在 MASK 110 中具有相應的一之來源運算元 115 中的值被複製入目的地 105 向量之當

前位置。

如圖 1 中所示，假如來源 115 包含資料元件 15 至 0，含有下列值：0, 0, 15, 0, 12, 35, 35, 35, 0, 0, 0, 0, 0, 0, 0, 54，則目的地 105 向量暫存器將包含下列值於資料元件 11 至 0 中：2, 0, 15, 1, 0, 12, 35, 35, 35, 7, 0, 54。從元件 0 至元件 15，來源 115 向量暫存器之第一元件將遭遇有一個 1 於 MASK 110 中，且 54 將被複製入目的地 105 向量暫存器之當前位置。接著一系列七個零將被計算且值 0 至 7 將被複製入目的地 105 向量暫存器之下兩個位置。之後，值 35, 35, 35, 及 12 將被複製入目的地 105 向量暫存器之接下來位置。現在一系列單 0 被遭遇以致 0 之壓縮實際上需要 2 元件以儲存僅一個單 0。因此，一個 0 及 1 之計數被儲存於目的地 105 向量暫存器之接下來資料元件中。最後，一個 15 接著一個 0 及一個 2 被儲存於目的地 105 向量暫存器之接下來位置中，代表來源 115 向量暫存器之該 15 及最後兩個零。使用範例來源 115 向量暫存器，零為基的 RLE 將向量從 16 資料元件壓縮為 12 資料元件，其產生百分之 25 的節省。

本發明之實施例輸出目的地 105 向量暫存器之哪些元件被用於來源 115 向量暫存器的指示。於一實施例中，此包括設定一已使用的元件控制遮罩 125 中之值。目的地 105 向量暫存器中之各已使用的元件係符合已使用的元件控制遮罩 125 中之一個 1。已使用的元件控制遮罩 125 可為變更用途的遮罩暫存器或特殊包括的遮罩暫存器。指令

100 之進一步實施例容許已使用的元件控制遮罩 125 被選擇並編碼為指令之發生。於圖 1 中，元件 11-0 被設定以一個 1 於已使用的元件控制遮罩 125 中，以指示目的地 105 向量暫存器中之哪些元件被使用。於另一實施例中，一已使用的元件計數 130 被設定以指示目的地 105 向量暫存器中之已使用元件的數目，於此例中係以值 12。如同已使用的元件控制遮罩 125，已使用的元件計數 130 可被設定於特殊暫存器中或者於變更用途的一般暫存器中。再者，指令 100 之實施例容許目的地、記憶體或暫存器被編碼於已使用的元件計數 130。雖然顯示為計算已使用的資料元件，其相反方式亦可被使用於指令之其他實施例中。明確地，明確地，已使用的元件控制遮罩及已使用的元件計數可為未使用的元件控制遮罩及未使用的元件計數。

圖 2 顯示依據一實施例之向量頻率壓縮指令的範例執行。向量頻率壓縮指令 200 被顯示為壓縮零以外的值。如上所述，一實施例包括一被編碼入指令 200 之即刻值 220，用以指示應被壓縮之值。指令之另一實施例使用 MASK 210 以判斷一值，其將藉由讀取相應於一或更多具有一 0 之 MASK 210 的來源 215 向量暫存器中所儲存之值而被壓縮。藉由設計，來源 215 向量暫存器之值可被判定且 MASK 210 被計算以符合針對一既定輸入之最有效率的編碼。於圖 2 中，值 35 被壓縮，因為其將產生最長的列以便使來源 215 向量暫存器之壓縮最佳化。來源 215 向量暫存器之資料元件 15 至 0 含有下列值：0, 0, 15, 0, 12,

35, 35, 35, 0, 7, 0, 98, 44, 0, 0, 及 54。這些值係使用值 35 之 RLE 而被壓縮於目的地 205 向量暫存器中而成爲資料元件 14-0 如下列值：0, 0, 15, 0, 12, 3, 35, 0, 7, 0, 98, 44, 0, 0, 及 54。如圖 2 中所示，3 個 35 之列被壓縮爲 35 接續於 3 在目的地 205 向量暫存器中。於圖 2 中，已使用的元件控制遮罩 225 之資料元件 14-0 被設爲 1 而已使用的元件計數 230 被設爲 15，以指示目的地 205 向量暫存器之哪些元件包含已壓縮的來源 215 向量暫存器。

存在有其無法由指令所完整地處置之輸入來源向量暫存器的情況。例如，可能有一些情況，其中來源向量暫存器含有某些值以致已壓縮的目的地向量暫存器將大於來源向量暫存器。諸如當來源含有應被壓縮之值的一或更多單一發生時，因爲各單一值被翻譯爲目的地之兩資料元件：值和計數。此情況可藉由計算應被壓縮之值的發生次數及計算該已計算的控制遮罩中之零的數目而由軟體所輕易地檢測。軟體可接著將壓縮劃分爲兩個循環以考量此情境或嘗試使用不同的 RLE 技術以使該輸入最佳化。此輸入情境之另一選擇是使該指令在每當此情況發生時提出一特殊例外並容許軟體被觸發以處置該情況。

圖 3 爲一流程圖，其顯示依據一實施例藉由執行處理器中之向量頻率壓縮指令以將來自來源向量暫存器之值壓縮至目的地向量暫存器的範例操作。於操作 310，向量頻率壓縮指令係由處理器所提取（例如，藉由處理器之提取單元）。向量頻率壓縮指令包括（至少）來源運算元及目

的地運算元。來源運算元指明一組應被壓縮入目的地運算元之資料元件（例如，xmm, ymm 或 zmm 暫存器）。於該指令之至少一次發生中，來源運算元包含其將被 RLE 處理成一值之一或更多列的值及該值的計數。

流程從操作 310 移至操作 315，其中處理解碼向量頻率壓縮指令。例如，於某些實施例中，處理器包括硬體解碼單元，其被提供指令（亦即，藉由處理器之提取單元）。多種不同之眾所周知的解碼單元可被用於該解碼單元。例如，解碼單元可將該向量頻率壓縮指令解碼為單一寬微指令。如另一範例，解碼單元可將該向量頻率壓縮指令解碼為多個寬微指令。如特別適於失序處理器管線之另一範例，解碼單元可將該向量頻率壓縮指令解碼為一或更多微操作，其中每一微操作可被失序地發送及執行。同時，可被實施與一或更多解碼器，且各解碼器可被實施為可編程邏輯陣列（PLA），如本技術中眾所周知者。舉例而言，一既定的解碼單元可：1）具有引導邏輯以將不同的微指令指引至不同的解碼器；2）第一解碼器，其可解碼指令集之一子集（其較第二、第三、及第四解碼器更多）並同時地產生兩個微操作；3）第二、第三、及第四解碼器，其可各解碼整個指令集之僅一子集並同時地產生僅一微操作；4）微排程器 ROM，其可解碼整個指令集之僅一子集並同時地產生四個微操作；及 5）多工邏輯，其係由該些解碼器及該微排程器 ROM 所饋送以判斷何者之輸出被提供至一微操作佇列。解碼單元之其他實施例可具

有更多或更少解碼器，其解碼更多或更少指令及指令子集。例如，一實施例可具有第二、第三、及第四解碼器，其可各同時地產生兩個微操作；並可包括一微排程器 ROM，其同時地產生八個微操作。

流程接著移至操作 320，其中處理器執行向量頻率壓縮指令，針對每一來源資料元件致使一值被複製入目的地暫存器以指示該來源資料元件之值。來源資料運算元之至少一子集被編碼入目的地向量暫存器而成爲一值及計數，藉此將來源資料元件壓縮入較少的目的地資料元件。

範例指令格式

文中所述之指令的實施例可被實施以不同格式。此外，範例系統、架構、及管線被詳述於下。指令之實施例可被執行於此等系統、架構、及管線之上，但不限定於那些詳述者。

VEX 指令格式

VEX 編碼容許指令具有多於二運算元，並容許 SIMD 向量暫存器長於 128 位元。VEX 前綴之使用提供三運算元（或更多）語法。例如，先前的二運算元指令係履行諸如 $A=A + B$ （其覆寫來源運算元）等操作。VEX 前綴之使用致能運算元履行諸如 $A=B + C$ 等非破壞性操作。

圖 4A 說明包括 VEX 前綴 402、真實運算碼欄位 430、Mod R/M 位元組 440、SIB 位元組 450、置換欄位

462、及 IMM8 472 之範例 AVX 指令格式。圖 4B 說明來自圖 4A 之哪些欄位組成全運算碼欄位 474 及基礎操作欄位 442。圖 4C 說明來自圖 4A 之哪些欄位組成暫存器指標欄位 444。

VEX 前綴（位元組 0-2）402 被編碼以三位元組形式。第一位元組為格式欄位 440（VEX 位元組 0，位元 [7:0]），其含有明確 C4 位元組值（用於分辨 C4 指令格式之獨特值）。第二-第三位元組（VEX 位元組 1-2）包括提供特定能力之數個位元欄位。明確地，REX 欄位 405（VEX 位元組 1，位元 [7-5]）包括 VEX.R 位元欄位（VEX 位元組 1，位元 [7]-R）、VEX.X 位元欄位（VEX 位元組 1，位元 [6]-X）、及 VEX.B 位元欄位（VEX 位元組 1，位元 [5]-B）。指令之其他欄位係編碼暫存器指標之較低三個位元，如本技術中已知者（rrr、xxx、及 bbb），以致 Rrrr、Xxxx、及 Bbbb 可藉由將 VEX.R、VEX.X、及 VEX.B 相加而形成。運算元映圖欄位 415（VEX 位元組 1，位元 [4:0]-mmmm）包括用以編碼暗示的領先運算元位元組之內容。W 欄位 464（VEX 位元組 2，位元 [7]-W）係由標號 VEX.W 所表示，並根據指令而提供不同功能。VEX.vvvv 420（VEX 位元組 2，位元 [6:3]-vvvv）之角色可包括以下：1）VEX.vvvv 編碼第一來源暫存器運算元、以反相（1s 補數）形式指明並可用於具有 2 或更多來源運算元之指令；2）VEX.vvvv 編碼目的地暫存器運算元、針對某些向量位移以 1s 補數形式指明；或 3）VEX.vvvv 不

編碼任何運算元、該欄位被保留並應含有 1111b。假如 VEX.L 468 大小欄位 (VEX 位元組 2, 位元 [2]-L) = 0, 則指示 128 位元向量; 假如 VEX.L=1, 則指示 256 位元向量。前綴編碼欄位 425 (VEX 位元組 2, 位元 [1:0]-pp) 提供額外位元給基礎操作欄位。

真實運算碼欄位 430 (位元組 3) 亦已知為運算碼位元組。運算碼之部分被指明於此欄位中。

Mod R/M 欄位 440 (位元組 4) 包括 MOD 欄位 442 (位元 [7-6])、Reg 欄位 444 (位元 [5-3])、及 R/M 欄位 446 (位元 [2-0])。Reg 欄位 444 之角色可包括下列: 編碼目的地暫存器運算元或來源暫存器運算元之任一者 (Rrrr 之 rrr)、或者被視為運算碼擴充且不被用於編碼任何指令運算元。R/M 欄位 446 之角色可包括下列: 編碼其參照記憶體位址之指令運算元、或者編碼目的地暫存器運算元或來源暫存器運算元之任一者。

比率、指標、基礎 (SIB) - 比率欄位 450 (位元組 5) 之內容包括 SS 452 (位元 [7-6]), 其被用於記憶體位址產生。SIB.xxx 454 (位元 [5-3]) 及 SIB.bbb 456 (位元 [2-0]) 之內容先前已針對暫存器指標 Xxxx 及 Bbbb 而被提及。

置換欄位 462 及即刻欄位 (IMM8) 472 含有位址資料。

範例編碼為 VEX

一般性向量友善指令格式

向量友善指令格式是一種適於向量指令之指令格式（例如，有專屬於向量操作之某些欄位）。雖然描述了其中向量和純量操作兩者均透過向量友善指令格式而被支援的實施例，但其他實施例僅使用向量操作於向量友善指令格式。

圖 5A-5B 為方塊圖，其說明依據本發明之實施例的一般性向量友善指令格式及其指令模板。圖 5A 為說明依據本發明之實施例的一般性向量友善指令格式及其類別 A 指令模板之方塊圖；而圖 5B 為說明依據本發明之實施例的一般性向量友善指令格式及其類別 B 指令模板之方塊圖。明確地，一般性向量友善指令格式 500 係定義類別 A 及類別 B 指令模板，其兩者包括無記憶體存取 505 指令模板和記憶體存取 520 指令模板。在向量友善指令格式之背景下的術語「一般性」指的是未連結任何特定指令集之指令格式。

雖然本發明之實施例將描述其中該向量友善指令格式支援下列：具有 32 位元（4 位元組）或 64 位元（8 位元組）資料元件寬度（或大小）之 64 位元組向量運算元長度（或大小）（而因此，64 位元組係由 16 個雙字元大小的元件或替代地 8 個四字元大小的元件所構成）；具有 16 位元（2 位元組）或 8 位元（1 位元組）資料元件寬度（或大小）之 64 位元組向量運算元長度（或大小）；具有 32 位元（4 位元組）、64 位元（8 位元組）、16 位元

(2 位元組) 、或 8 位元 (1 位元組) 資料元件寬度 (或大小) 之 32 位元組向量運算元長度 (或大小) ；及具有 32 位元 (4 位元組) 、64 位元 (8 位元組) 、16 位元 (2 位元組) 、或 8 位元 (1 位元組) 資料元件寬度 (或大小) 之 16 位元組向量運算元長度 (或大小) ；但是替代實施例可支援具有更多、更少、或不同資料元件寬度 (例如，128 位元 (16 位元組) 資料元件寬度) 之更多、更少及 / 或不同向量運算元大小 (例如，256 位元組向量運算元) 。

圖 5A 中之類別 A 指令模板包括：1) 於無記憶體存取 505 指令模板內顯示有無記憶體存取、全捨入 (full round) 控制類型操作 510 指令模板及無記憶體存取、資料轉變類型操作 515 指令模板；以及 2) 於記憶體存取 520 指令模板內顯示有記憶體存取、暫時 525 指令模板及記憶體存取、非暫時 530 指令模板。圖 5B 中之類別 B 指令模板包括：1) 於無記憶體存取 505 指令模板內顯示有無記憶體存取、寫入遮罩控制、部分捨入控制類型操作 512 指令模板及無記憶體存取、寫入遮罩控制、vsize 類型操作 517 指令模板；以及 2) 於記憶體存取 520 指令模板內顯示有記憶體存取、寫入遮罩控制 527 指令模板。

一般性向量友善指令格式 500 包括依圖 5A-5B 中所示之順序所列出於下的如下欄位。

格式欄位 540-此欄位中之特定值 (指令格式識別符值) 獨特地識別向量友善指令格式，而因此識別指令流中

之向量友善指令格式的指令之發生。如此一來，此欄位是選擇性的，因為其對於僅具有一般性向量友善指令格式之指令集是不需要的。

基礎操作欄位 542-其內容係分辨不同的基礎操作。

暫存器指標欄位 544-其內容（直接地或透過位址產生）指明來源及目的地運算元之位置，任其於暫存器中或記憶體中。這些包括足夠的位元數以從 $P \times Q$ （例如， 32×512 、 16×128 、 32×1024 、 64×1024 ）暫存器檔案選擇 N 暫存器。雖然於一實施例中， N 可高達三個來源及一個目的地暫存器，但替代實施例可支援更多或更少來源及目的地暫存器（例如，可支援高達兩個來源，其中這些來源之一亦作用為目的地；可支援高達三個來源，其中這些來源之一亦作用為目的地；可支援高達兩個來源及一個目的地）。

修飾符欄位 546-其內容係從那些不指明記憶體存取者分辨其指明記憶體存取之一般性向量指令格式中的指令之發生；亦即，介於無記憶體存取 505 指令模板與記憶體存取 520 指令模板之間。記憶體存取操作係讀取及/或寫入至記憶體階層（於某些情況下使用暫存器中之值以指明來源及/或目的地位址），而無記憶體存取操作則不（例如，來源及目的地為暫存器）。雖然於一實施例中，此欄位亦於三個不同方式之間選擇以履行記憶體位址計算，但替代實施例可支援更多、更少、或不同方式以履行記憶體位址計算。

擴增 (augmentation) 操作欄位 550-其內容係分辨除了基礎操作之外的多種不同操作之何者應被履行。此欄位是背景特定的。於本發明之一實施例中，此欄位被劃分為類別欄位 568、阿爾發欄位 552、及貝他欄位 554。擴增操作欄位 550 容許共同族群的操作被履行於單一指令而非 2、3 或 4 個指令。

比率欄位 560-其內容容許指標欄位之內容的定標 (scaling) 以供記憶體位址產生 (例如，用於使用 $2^{\text{scale}} * \text{index} + \text{base}$ 之位址產生) 。

置換欄位 562A-其內容被使用為記憶體位址產生之部分 (例如，用於使用 $2^{\text{scale}} * \text{index} + \text{base} + \text{displacement}$ 之位址產生) 。

置換因數欄位 562B (注意其直接於置換因數欄位 562B 上方的置換欄位 562A 之並列指示一者或另一者被使用) -其內容被使用為位址產生之部分；其指明將由記憶體存取之大小 (N) 所定標的置換因數 -其中 N 為記憶體存取中之位元組數 (例如，用於使用 $2^{\text{scale}} * \text{index} + \text{base} + \text{scaled displacement}$ 之位址產生) 。多餘的低階位元被忽略，因此，置換因數欄位之內容被乘以記憶體運算元總大小 (N) 以產生最終置換來被用於計算有效位址。N 之值係根據全運算碼欄位 574 (文中稍後所描述) 及資料調處欄位 554C 而由處理器硬體判定於運行時間。置換欄位 562A 及置換因數欄位 562B 是選擇性的，因為其並未用於無記憶體存取 505 指令模板及 / 或不同的實施例可僅實施

兩者之一或無。

資料元件寬度欄位 564-其內容係分辨數個資料元件寬度之何者應被使用（於某些實施例用於所有指令；於其他實施例中僅用於部分指令）。此欄位是選擇性的，因為其是不需要的假如僅有一資料元件寬度被支援及/或資料元件寬度係使用運算碼之某形態而被支援。

寫入遮罩欄位 570-其內容控制，以每資料元件位置為基，目的地向量運算元中之資料元件位置是否反應基礎操作及擴增操作之結果。類別 A 指令模板支援合併-寫入遮蔽，而類別 B 指令模板支援合併-和歸零-寫入遮蔽兩者。當合併時，向量遮罩容許目的地中之任一組元件被保護不被更新於任何操作之執行期間（由基礎操作及擴增操作所指明）；於另一實施例中，保存目的地之各元件的舊值，其中相應的遮罩位元具有 0。反之，當歸零時，向量遮罩容許目的地中之任一組元件被歸零於任何操作之執行期間（由基礎操作及擴增操作所指明）；於另一實施例中，當相應的遮罩位元具有 0 值時目的地之一元件被設為 0。此功能之一子集為控制其正履行中之操作的向量長度之能力（亦即，元件之跨距被修改，從第一至最後者）；然而，當被修改之元件為連續時則其為不需要的。因此，寫入遮罩欄位 570 容許部分向量操作，包括載入、儲存、算術、邏輯，等等。雖然本發明之實施例係描述其中寫入遮罩欄位 570 之內容選擇含有待使用之寫入遮罩的數個寫入遮罩暫存器之一（而因此寫入遮罩欄位 570 之內容間接地識別

其應履行之遮蔽)，但替代實施例取代地或額外地容許寫入遮罩欄位 570 之內容直接地指明應履行之遮蔽。

即刻欄位 572-其內容容許一即刻之指明。此欄位是選擇性的，因為在不支援即刻之一般性向量友善指令格式的實施中其並不存在以及在不使用即刻之指令中其並不存在。

類別欄位 568-其內容係分辨於不同類別的指令之間。參考圖 5A-B，此欄位之內容係選擇於類別 A 與類別 B 指令之間。於圖 5A-B 中，圓角的方塊係用以指示一特定值出現在一欄位中（例如，個別於圖 5A-B 中的類別欄位 568 之類別 A 568A 及類別 B 568B）。

類別 A 之指令模板

於類別 A 之無記憶體存取 505 指令模板的情況下，阿爾發欄位 552 被解讀為 RS 欄位 552A，其內容係分辨不同擴增操作類型之何者應被履行（例如，捨入 552A.1 及資料轉變 552A.2 被個別地指明給無記憶體存取、捨入類型操作 510 及無記憶體存取、資料轉變類型操作 515 指令模板），而貝他欄位 554 係分辨已指明類型之操作的何者應被履行。於無記憶體存取 505 指令模板中，比率欄位 560、置換欄位 562A、及置換比率欄位 562B 並未出現。

無記憶體存取指令模板-全捨入控制類型操作

於無記憶體存取全捨入控制類型操作 510 指令模板

中，貝他欄位 554 被解讀為捨入控制欄位 554A，其內容提供靜態捨入。雖然於本發明之已描述實施例中捨入控制欄位 554A 包括一抑制所有浮點例外（SAE）欄位 556 及一捨入操作控制欄位 558，但替代實施例可支援將這些觀念編碼入相同欄位中或者僅具有這些觀念/欄位之一或另一（例如，可僅具有捨入操作控制欄位 558）。

SAE 欄位 556-其內容係分辨是否使例外事件報告失效；當 SAE 欄位 556 之內容指示抑制啓用時，則一既定指令不會報告任何種類的浮點例外旗標且不會提出任何浮點例外處置器。

捨入操作控制欄位 558-其內容係分辨捨入操作之族群的何者應履行（例如，捨進、捨去、朝零捨入及捨入至最接近）。因此，捨入操作控制欄位 558 容許以每指令為基之捨入模式的改變。於其中處理器包括一用以指明捨入模式之控制暫存器的本發明之一實施例中，捨入操作控制欄位 550 之內容係置換該暫存器值。

無記憶體存取指令模板-資料轉變類型操作

於無記憶體存取資料轉變類型操作 515 指令模板中，貝他欄位 554 被解讀為資料轉變欄位 554B，其內容係分辨數個資料轉變之何者應被履行（例如，無資料轉變、拌和、廣播）。

於類別 A 之記憶體存取 520 指令模板的情況下，阿爾發欄位 552 被解讀為逐出（eviction）暗示欄位 552B，其

內容係分辨逐出暗示之何者應被使用（於圖 5A 中，暫時 552B.1 及非暫時 552B.2 被個別地指明給記憶體存取、暫時 525 指令模板及記憶體存取、非暫時 530 指令模板），而貝他欄位 554 被解讀為資料調處欄位 554C，其內容係分辨數個資料調處操作（亦已知為基元）之何者應被履行（例如，無調處；廣播；來源之上轉換；及目的地之下轉換）。記憶體存取 520 指令模板包括比率欄位 560、及選擇性地置換欄位 562A 或置換比率欄位 562B。

向量記憶體指令履行向量載入自及向量儲存至記憶體，具有轉換支援。如同普通向量指令，向量記憶體指令以資料元件式方式將資料轉移自/至記憶體，其中被實際地轉移之元件係由其被選擇為寫入遮罩之向量遮罩的內容所支配。

記憶體存取指令模板 - 暫時

暫時資料為可能夠快地被再使用而受益自快取的資料。然而，此為暗示，且不同處理器可用不同方式來實施之，包括完全忽略暗示。

記憶體存取指令模板 - 非暫時

非暫時資料為不太可能夠快地被再使用而受益自第一階快取中之快取且應被給定逐出之優先權的資料。然而，此為暗示，且不同處理器可用不同方式來實施之，包括完全忽略暗示。

類別 B 之指令模板

於類別 B 之指令模板的情況下，阿爾發欄位 552 被解讀為寫入遮罩控制 (Z) 欄位 552C，其內容係分辨其由寫入遮罩欄位 570 所控制之寫入遮蔽是否應為合併或歸零。

於類別 B 之無記憶體存取 505 指令模板的情況下，貝他欄位 554 之部分被解讀為 RL 欄位 557A，其內容係分辨不同擴增操作類型之何者應被履行（例如，捨入 557A.1 及向量長度 (VSIZE) 557A.2 被個別地指明給無記憶體存取、寫入遮罩控制、部分捨入控制類型操作 512 指令模板及無記憶體存取、寫入遮罩控制、VSIZE 類型操作 517 指令模板），而貝他欄位 554 之剩餘者係分辨已指明類型之操作的何者應被履行。於無記憶體存取 505 指令模板中，比率欄位 560、置換欄位 562A、及置換比率欄位 562B 並未出現。

於無記憶體存取、寫入遮罩控制、部分捨入控制類型操作 510 指令模板中，貝他欄位 554 之剩餘者被解讀為捨入操作欄位 559A 並使例外事件報告失效（一既定指令不會報告任何種類的浮點例外旗標且不會提出任何浮點例外處置器）。

捨入操作控制欄位 559A-正如同捨入操作控制欄位 558，其內容係分辨捨入操作之族群的何者應履行（例如，捨進、捨去、朝零捨入及捨入至最接近）。因此，捨入操作控制欄位 559A 容許以每指令為基之捨入模式的改

變。於其中處理器包括一用以指明捨入模式之控制暫存器的本發明之一實施例中，捨入操作控制欄位 550 之內容係置換該暫存器值。

於無記憶體存取、寫入遮罩控制、VSIZE 類型操作 517 指令模板中，貝他欄位 554 被解讀為向量長度欄位 559B，其內容係分辨數個資料向量長度之何者應被履行（例如，128、256、或 512 位元組）。

於類別 B 之記憶體存取 520 指令模板的情況下，貝他欄位 554 之部分被解讀為廣播欄位 557B，其內容係分辨廣播類型資料調處操作是否應被履行，而貝他欄位 554 之剩餘者被解讀為向量長度欄位 559B。記憶體存取 520 指令模板包括比率欄位 560、及選擇性的置換欄位 562A、或置換比率欄位 562B。

針對一般性向量友善指令格式 500，一全運算碼欄位 574 顯示為包括格式欄位 540、基礎操作欄位 542、及資料元件寬度欄位 564。雖然一實施例係顯示全運算碼欄位 574 包括所有這些欄位，但於其不支援這些所有的實施例中全運算碼欄位 574 可包括少於所有這些欄位。全運算碼欄位 574 提供運算碼（opcode）。

擴增操作欄位 550、資料元件寬度欄位 564、及寫入遮罩欄位 570 容許這些特徵以每指令為基被指明於一般性向量友善指令格式中。

寫入遮罩欄位與資料元件寬度欄位之組合產生定型的指令，由於其容許遮罩根據不同資料元件寬度而被應用。

於類別 A 和類別 B 中所發現的各種指令模板於不同情況下是有利的。於本發明之某些實施例中，不同處理器或一處理器內之不同核心可支援唯獨類別 A、唯獨類別 B、或兩個類別。例如，用於通用計算之高性能通用失序核心可支援唯獨類別 B；主要用於圖形及/或科學（通量）計算之核心可支援唯獨類別 A；而用於上述兩者之核心可支援兩類別（當然，具有來自兩類別之模板與指令的某種混合但非來自兩類別之所有模板和指令的核心仍於本發明之範圍內）。同時，單一處理器可包括多個核心，其所有均支援相同類別或者其中不同核心支援不同類別。例如，於具有分離的圖形和通用核心之處理器中，主要用於圖形及/或科學計算的圖形核心之一可支援唯獨類別 A，而通用核心之一或更多可為高性能通用核心，其具有用於支援唯獨類別 B 之通用計算的失序執行和暫存器重新命名。不具有分離圖形核心之另一處理器可包括其支援類別 A 與類別 B 兩者之一或更多通用依序或失序核心。當然，來自一類別之特徵亦可被實施於本發明之不同實施例中的其他類別中。以高階語言所寫的程式將被輸入（例如，僅於時間編譯或靜態編譯）多種不同的可執行形式，包括：1）僅具有由用於執行之目標處理器所支援之類別的指令；或 2）具有使用所有類別之指令的不同組合所寫的替代常式並具有其根據由目前正執行碼之處理器所支援的指令以選擇供執行之常式的控制流程碼的形式。

範例特定向量友善指令格式

圖 6A 為方塊圖，其說明依據本發明之實施例的範例特定向量友善指令格式。圖 6A 顯示一特定向量友善指令格式 600，其係由於指明欄位之位置、大小、解讀、和順序、以及那些欄位之部分的值而為特定的。特定向量友善指令格式 600 可用以擴充 x86 指令集，而因此某些欄位係類似於或相同於現有的 x86 指令集及其擴充（例如，AVX）中所使用的那些欄位。此格式保持為與具有擴充之現有的 x86 指令集之前綴編碼欄位、真實運算碼位元組欄位、MOD R/M 欄位、SIB 欄位、置換欄位、及即刻欄位一致。映射入來自圖 6A 之欄位的來自圖 5 之欄位被顯示。

應理解：雖然本發明係參考於一般性向量友善指令格式 500 之背景下的特定向量友善指令格式 600 來描述以利說明之目的，但本發明除了所請求的範圍之外並不限於特定向量友善指令格式 600。例如，一般性向量友善指令格式 500 係考量針對各種欄位之多種可能的大小，而特定向量友善指令格式 600 則顯示為具有特定大小的欄位。藉由特定範例，雖然資料元件寬度欄位 564 被顯示為特定向量友善指令格式 600 中之一位元欄位，但本發明並未如此受限（亦即，一般性向量友善指令格式 500 係考量資料元件寬度欄位 564 之其他大小）。

一般性向量友善指令格式 500 包括依圖 6A 所示之順序的如下欄位。

EVEX 前綴（位元組 0-3）602-被編碼以四位元組之形

式。

格式欄位 540 (EVEX 位元組 0, 位元 [7:0]) - 第一位元組 (EVEX 位元組 0) 為格式欄位 540 且其含有 0x62 (用於分辨本發明之一實施例中的向量友善指令格式之獨特值)。

第二-第四位元組 (EVEX 位元組 1-3) 包括提供特定能力之數個位元欄位。

REX 欄位 605 (EVEX 位元組 1, 位元 [7-5]) - 由 EVEX.R 位元欄位 (EVEX 位元組 1, 位元 [7]-R)、EVEX.X 位元欄位 (EVEX 位元組 1, 位元 [6]-X)、及 557BEX 位元組 1, 位元 [5]-B 所組成。EVEX.R、EVEX.X 及 EVEX.B 位元欄位係提供如相應 VEX 位元欄位之相同的功能, 且係使用 1 補數形式來編碼, 亦即, ZMM0 被編碼為 1111B; ZMM15 被編碼為 0000B。指令之其他欄位將暫存器指標之較低三個位元編碼, 如本技術中所已知者 (rrr, xxx 及 bbb), 以致 Rrrr、Xxxx、及 Bbbb 可藉由加入 EVEX.R、EVEX.X 及 EVEX.B 而形成。

REX' 欄位 510 - 此為 REX' 欄位 510 之第一部分且為用以將擴充的 32 暫存器集之上 16 或下 16 個編碼的 EVEX.R' 位元欄位 (EVEX 位元組 1, 位元 [4]-R')。於本發明之一實施例中, 此位元 (連同以下所指出之其他位元) 被儲存以位元反轉格式來分辨 (以眾所周知的 x86 32 位元模式) 自 BOUND 指令, 其真實運算碼位元組為 62, 但於 MOD R/M 欄位 (如下所述) 中並未接受 MOD 欄位中

之 11 的值；本發明之替代實施例並未以反轉格式儲存此及以下所指示的位元。1 之值被用以編碼下 16 暫存器。換言之，R'Rrrr 係藉由組合 EVEX.R'、EVEX.R、及來自其他欄位之其他 RRR 而形成。

運算碼映圖欄位 615 (EVEX 位元組 1, 位元 [3:0]-mmmm) -其內容係編碼一隱含的前導 (leading) 運算碼位元組 (0F、0F 38 或 0F 3)。

資料元件寬度欄位 564 (EVEX 位元組 2, 位元 [7]-W) -係由記法 EVEX.W 所表示。EVEX.W 係用以定義資料位元組 (32 位元資料元件或 64 位元資料元件) 之粒度 (大小)。

EVEX.vvvv 620 (EVEX 位元組 2, 位元 [6:3]-vvvv) - EVEX.vvvv 之角色可包括下列：1) EVEX.vvvv 編碼第一來源暫存器運算元，以反轉 (1 補數) 形式指明且針對具有二或更多來源運算元之指令是有效的；2) EVEX.vvvv 編碼目的地暫存器運算元，以 1 補數形式指明於某些向量位移；或 3) EVEX.vvvv 未編碼任何運算元，該欄位被保留且應含有 1111b。因此，EVEX.vvvv 620 編碼其以反轉 (1 補數) 形式所儲存之第一來源暫存器指明符的 4 個低階位元。根據該指令，一額外的不同 EVEX 位元欄位被用以擴充指明符大小至 32 暫存器。

EVEX.U 568 類別欄位 (EVEX 位元組 2, 位元 [2]-U) -假如 EVEX.U=0, 其指示類別 A 或 EVEX.U0；假如 EVEX.U=1, 其指示類別 B 或 EVEX.U1。

前綴編碼欄位 625 (EVEX 位元組 2, 位元 [1:0]-pp) - 提供基礎操作欄位之額外位元。除了提供 EVEX 前綴格式之傳統 SSE 指令的支援以外, 此亦具有壓縮 SIMD 前綴之優點 (取代需要一位元組來表達 SIMD 前綴, EVEX 前綴僅需要 2 位元)。於一實施例中, 爲了支援其使用 SIMD 前綴 (66H, F2H, F3H) 之傳統 SSE 指令於傳統格式和 EVEX 前綴格式兩者, 這些傳統 SIMD 前綴被編碼入 SIMD 前綴編碼欄位; 且於運行時間被擴充爲傳統 SIMD 前綴, 在被提供至解碼器之 PLA 以前 (因此 PLA 可執行這些傳統指令之傳統和 EVEX 格式而無修改)。雖然較新的指令可使用 EVEX 前綴編碼欄位之內容爲運算碼擴充, 但某些實施例以類似方式擴充一致性而容許由這些傳統 SIMD 前綴指明不同意義。一替代實施例可重新設計 PLA 以支援 2 位元 SIMD 前綴編碼, 而因此不需要擴充。

阿爾發欄位 552 (EVEX 位元組 3, 位元 [7]-EH; 亦已知爲 EVEX.EH、EVEX.rs、EVEX.RL、EVEX.寫入遮罩控制、及 EVEX.N; 亦以 α 顯示) -如先前所述, 此欄位爲背景特定的。

貝他欄位 554 (EVEX 位元組 3, 位元 [6:4]-SSS, 亦已知爲 EVEX.s₂₋₀、EVEX.r₂₋₀、EVEX.rr1、EVEX.LL0、EVEX.LLB; 亦以 $\beta \beta \beta$ 顯示) -如先前所述, 此欄位爲背景特定的。

REX'欄位 510-此爲 REX'欄位之剩餘者且爲可用以將擴充的 32 暫存器集之上 16 或下 16 個編碼的 EVEX.V'位

元欄位 (EVEX 位元組 3, 位元 [3]-V')。此位元被儲存以位元反轉格式。1 之值被用以編碼下 16 暫存器。換言之, V'VVVV 係藉由組合 EVEX.V'、EVEX.vvvv 而形成。

寫入遮罩欄位 570 (EVEX 位元組 3, 位元 [2:0]-kkk) - 其內容係指明寫入遮罩暫存器中之暫存器的指標, 如先前所述者。於本發明之一實施例中, 特定值 EVEX.kkk=000 具有一特殊行為, 其隱含無寫入遮罩被用於特定指令 (此可被實施於多種方式, 包括使用固線至所有電路之寫入遮罩或者其旁通遮蔽硬體之硬體)。

真實運算碼欄位 630 (位元組 4) 亦已知為運算碼位元組。運算碼之部分被指明於此欄位中。

MOD R/M 欄位 640 (位元組 5) 包括 MOD 欄位 642、Reg 欄位 644、及 R/M 欄位 646。如先前所述, MOD 欄位 642 之內容係分辨於記憶體存取與非記憶體存取操作之間。Reg 欄位 644 之角色可被概述為兩種情況: 編碼目的地暫存器運算元或來源暫存器運算元之任一者、或者被視為運算碼擴充且不被用於編碼任何指令運算元。R/M 欄位 646 之角色可包括下列: 編碼其參照記憶體位址之指令運算元、或者編碼目的地暫存器運算元或來源暫存器運算元之任一者。

比率、指標、基礎 (SIB) 位元組 (位元組 6) - 如先前所述, 比率欄位 550 之內容被用於記憶體位址產生。SIB.xxx 654 及 SIB.bbb 656-這些欄位之內容先前已針對暫存器指標 Xxxx 及 Bbbb 而被提及。

置換欄位 562A (位元組 7-10) -當 MOD 欄位 642 含有 10 時，位元組 7-10 為置換欄位 562A，且其工作相同於傳統 32 位元置換 (disp32) 且工作於位元組粒度。

置換因數欄位 562B (位元組 7) -當 MOD 欄位 642 含有 01 時，位元組 7 為置換因數欄位 562B。此欄位之位置係相同於傳統 x86 指令集 8 位元置換 (disp8)，工作於位元組粒度。因為 disp8 為符號擴充，所以其僅可定址於 -128 與 127 位元組偏移之間；針對 64 位元組快取線，disp8 使用其僅可被設為四個實際有用值 -128、-64、0、及 64 之 8 個位元；因為常需要較大的範圍，所以 disp32 被使用；然而，disp32 需要 4 個位元組。相反於 disp8 及 disp32，置換因數欄位 562B 為 disp8 之再解讀；當使用置換因數欄位 562B 時，實際置換係由置換因數欄位乘以記憶體運算元存取之大小 (N) 的內容所決定。此類型的置換被稱為 $\text{disp8} * N$ 。此係減少平均指令長度 (用於置換但具有大得多的範圍之單一位元組)。此壓縮的置換係基於假設其有效置換為記憶體存取之粒度的倍數，而因此，位址偏移之多餘的低階位元無須被編碼。換言之，置換因數欄位 562B 取代傳統 x86 指令集 8 位元置換。因此，置換因數欄位 562B 被編碼以如 x86 指令集 8 位元置換之相同方式 (因此 ModRM/SIB 編碼規則並無改變)，唯一例外為 disp8 被超載至 $\text{disp8} * N$ 。換言之，編碼規則或編碼長度並無改變，而僅於藉由硬體之置換值的解讀 (其需由記憶體運算元之大小定標該置換以獲得位元組式的位址偏

移)。

即刻欄位 572 係操作如先前所述。

全運算碼欄位

圖 6B 為方塊圖，其說明組成全運算碼欄位 574 之特定向量友善指令格式 600 的欄位，依據本發明之一實施例。明確地，全運算碼欄位 574 包括格式欄位 540、基礎操作欄位 542、及資料元件寬度 (W) 欄位 564。基礎操作欄位 542 包括前綴編碼欄位 625、運算碼映圖欄位 615、及真實運算碼欄位 630。

暫存器指標欄位

圖 6C 為方塊圖，其說明組成暫存器指標欄位 544 之特定向量友善指令格式 600 的欄位，依據本發明之一實施例。明確地，暫存器指標欄位 544 包括 REX 欄位 605、REX' 欄位 610、MODR/M.reg 欄位 644、MODR/M.r/m 欄位 646、VVVV 欄位 620、xxx 欄位 654、及 bbb 欄位 656。

擴增操作欄位

圖 6D 為方塊圖，其說明組成擴增操作欄位 550 之特定向量友善指令格式 600 的欄位，依據本發明之一實施例。當類別 (U) 欄位 568 含有 0 時，其表示 EVEX.U0 (類別 A 568A)；當其含有 1 時，其表示 EVEX.U1 (類別 B 568B)。當 U=0 且 MOD 欄位 642 含有 11 (表示無

記憶體存取操作)時，阿爾發欄位 552 (EVEX 位元組 3, 位元 [7]-EH) 被解讀為 rs 欄位 552A。當 rs 欄位 552A 含有一個 1 (捨入 552A.1) 時，貝他欄位 554 (EVEX 位元組 3, 位元 [6:4]-SSS) 被解讀為捨入控制欄位 554A。捨入控制欄位 554A 包括一位元 SAE 欄位 556 及二位元捨入操作欄位 558。當 rs 欄位 552A 含有 0 (資料轉變 552A.2) 時，貝他欄位 554 (EVEX 位元組 3, 位元 [6:4]-SSS) 被解讀為三位元資料轉變欄位 554B。當 U=0 且 MOD 欄位 642 含有 00、01、或 10 (表示記憶體存取操作) 時，阿爾發欄位 552 (EVEX 位元組 3, 位元 [7]-EH) 被解讀為逐出暗示 (EH) 欄位 552B 而貝他欄位 554 (EVEX 位元組 3, 位元 [6:4]-SSS) 被解讀為三位元資料調處欄位 554C。

當 U=1 時，阿爾發欄位 552 (EVEX 位元組 3, 位元 [7]-EH) 被解讀為寫入遮罩控制 (Z) 欄位 552C。當 U=1 且 MOD 欄位 642 含有 11 (表示無記憶體存取操作) 時，貝他欄位 554 之部分 (EVEX 位元組 3, 位元 [4]-S₀) 被解讀為 RL 欄位 557A；當其含有 1 (捨入 557A.1) 時，貝他欄位 554 之剩餘者 (EVEX 位元組 3, 位元 [6-5]-S₂₋₁) 被解讀為捨入操作欄位 559A；而當 RL 欄位 557A 含有 0 (VSIZE 557.A2) 時，貝他欄位 554 之剩餘者 (EVEX 位元組 3, 位元 [6-5]-S₂₋₁) 被解讀為向量長度欄位 559B (EVEX 位元組 3, 位元 [6-5]-L₁₋₀)。當 U=1 且 MOD 欄位 642 含有 00、01、或 10 (表示記憶體存取操作) 時，貝他欄位 554 (EVEX 位元組 3, 位元 [6:4]-SSS) 被解讀為向量

長度欄位 559B (EVEX 位元組 3, 位元 [6-5]-L₁₋₀) 及廣播欄位 557B (EVEX 位元組 3, 位元 [4]-B)。

範例暫存器架構

圖 7 為依據本發明之一實施例的暫存器架構 700 之方塊圖。於所示之實施例中，有 32 個 512 位元寬之向量暫存器 710；這些暫存器被稱為 zmm0 至 zmm31。較低的 16 個 zmm 暫存器之較低階 256 位元被疊置在暫存器 ymm0-16 之上。較低的 16 個 zmm 暫存器之較低階 128 位元 (ymm 暫存器之較低階 128 位元) 被疊置在暫存器 xmm0-15 之上。特定向量友善指令格式 600 係操作於這些疊置的暫存器檔案上，如下表所示。

可調整向量長度	類別	操作	暫存器
不包括向量長度欄位 559B 之指令模板	A (圖 5A; U=0)	510、515、 525、530	zmm 暫存器 (向量長度為 64 位元組)
	B (圖 5B; U=1)	512	zmm 暫存器 (向量長度為 64 位元組)
包括向量長度欄位 559B 之指令模板	B (圖 5B; U=1)	517、527	zmm、ymm、或 xmm 暫存器 (向量長度為 64 位元組、 32 位元組、或 16 位元組)， 根據向量長度欄位 559B

換言之，向量長度欄位 559B 選擇於最大長度與一或更多其他較短長度之間，其中每一此較短長度為先前長度之長度的一半；而無向量長度欄位 559B 之指令模板係操作於最大向量長度上。此外，於一實施例中，特定向量友

善指令格式 600 之類別 B 指令模板係操作於緊縮或純量單/雙精確浮點資料上以及緊縮或純量整數資料上。純量操作為履行在 `zmm/ymm/xmm` 暫存器中之較低階資料元件位置上的操作；較高階資料元件位置係根據實施例而被保留如執行前之相同者或被歸零。

寫入遮罩暫存器 715-於所示之實施例中，有 8 個寫入遮罩暫存器（`k0` 至 `k7`），大小各為 64 位元。於一替代實施例中，寫入遮罩暫存器 715 之大小為 16 位元。如先前所述，於本發明之一實施例中，向量遮罩暫存器 `k0` 無法被使用為寫入遮罩；當其通常將指示 `k0` 之編碼被用於寫入遮罩時，其選擇 `0xFFFF` 之固線式（`hardwired`）寫入遮罩，有效地除能該指令之寫入遮蔽。

通用暫存器 725-於所示之實施例中，有十六個 64 位元的通用暫存器，其係配合現有的 `x86` 定址模式而使用以定址記憶體運算元。這些暫存器被稱為下列名稱：`RAX`、`RBX`、`RCX`、`RDX`、`RBP`、`RSI`、`RDI`、`RSP`、及 `R8` 至 `R15`。

純量（`scalar`）浮點堆疊暫存器檔案（`x87` 堆疊）745，於其上係混疊 `MMX` 緊縮整數平坦暫存器檔案 750-於所示之實施例中，`x87` 堆疊為八元件的堆疊，用以對其使用 `x87` 指令集擴充之 32/64/80 位元的浮點資料履行純量浮點操作；而 `MMX` 暫存器被用以對 64 位元的緊縮整數資料履行操作，以及保留運算元給某些於 `MMX` 與 `XMM` 暫存器之間所履行的操作。

本發明之替代實施例可使用較寬的或較窄的暫存器。此外，本發明之替代實施例可使用更多的、更少的、或不同的暫存器檔案及暫存器。

範例核心架構、處理器、及電腦架構

處理器核心可被實施以不同方式、用於不同目的、以及於不同處理器中。例如，此類核心之實施可包括：1) 用於通用計算之通用依序核心；2) 用於通用計算之高性能通用失序核心；3) 主要用於圖形及/或科學（通量）計算之特殊用途核心。不同處理器之實施可包括：1) 包括一或更多用於通用計算之通用依序核心及/或一或更多用於通用計算之通用失序核心的 CPU；及 2) 包括一或更多主要用於圖形及/或科學（通量）之特殊用途核心。此類不同處理器導致不同的電腦系統架構，其可包括：1) 與 CPU 分離之晶片上的共處理器；2) 與 CPU 相同之封裝中之分離晶粒上的共處理器；3) 與 CPU 相同之晶粒上的共處理器（於此情況下，此一共處理器有時稱為特殊用途邏輯，諸如集成圖形及/或科學（通量）邏輯、或稱為特殊用途核心）；及 4) 可包括在相同晶粒上之上述 CPU（有時稱為應用核心或應用處理器）、上述共處理器、及額外功能之晶片上的系統。接下來描述範例核心架構，而接著描述範例處理器及電腦架構。

範例核心架構

依序及失序核心方塊圖

圖 8A 為一方塊圖，其說明依據本發明之實施例的範例依序的管線及範例暫存器重新命名的、失序的發出/執行管線兩者。圖 8B 為一方塊圖，其說明包括於一依據本發明之實施例的處理器中之依序架構核心之範例實施例及範例暫存器重新命名的、失序的發出/執行架構核心兩者。圖 8A-B 中之實線方塊係說明依序管線及依序核心，而虛線方塊之選配性加入則說明暫存器重新命名的、失序的發出/執行管線及核心。假設依序形態為失序形態之子集，則將描述失序形態。

於圖 8A 中，處理器管線 800 包括提取級 802、長度解碼級 804、解碼級 806、配置級 808、重新命名級 810、排程（亦已知為調度(dispatch)或發出）級 812、暫存器讀取/記憶體讀取級 814、執行級 816、寫回/記憶體寫入級 818、異常處置級 822、及確定(commit)級 824。

圖 8B 顯示處理器核心 890，其包括耦合至執行引擎單元 850 之前端單元 830，兩者均耦合至記憶體單元 870。核心 890 可為精簡指令集計算(RISC)核心、複雜指令集計算(CISC)核心、極長指令字元(VLIW)核心、或者混合或替代核心型。當作又另一選項，核心 890 可為特殊用途核心，諸如(例如)網路或通訊核心、壓縮引擎、共處理器核心、通用計算圖形處理單元(GPGPU)核心、圖形核心，等等。

前端單元 830 包括一耦合至指令快取單元 834 之分支

預測單元 832，指令快取單元 834 係耦合至指令翻譯旁看緩衝器 (TLB) 836，指令翻譯旁看緩衝器 (TLB) 836 係耦合至指令提取單元 838，指令提取單元 838 係耦合至解碼單元 840。解碼單元 840 (或解碼器) 可解碼指令，並產生下列之一者或更多者以當作輸出：微操作、微碼進入點、微指令、其他指令、或其他控制信號，其係解碼 (或者反射、或被衍生) 自原始指令。解碼單元 840 可使用各種不同機制而被實施。適當機制之範例包括 (但不限定於) 查找表、硬體實施、可編程邏輯陣列 (PLA)、微碼唯讀記憶體 (ROM)，等等。於一實施例中，核心 890 包括微碼 ROM 或其他媒體，其係儲存某些微指令之微碼 (例如，於解碼單元 840 中或者另外於前端單元 830 內)。解碼單元 840 係耦合至執行引擎單元 850 中之重新命名/配置器單元 852。

執行引擎單元 850 包括重新命名/配置器單元 852，其係耦合至收回單元 854 及一組一或更多排程器單元 856。排程器單元 856 代表任何數目的不同排程器，包括保留站、中央指令視窗，等等。排程器單元 856 被耦合至實體暫存器檔案單元 858。實體暫存器檔案單元 858 之每一者代表一或更多實體暫存器檔案，其各不同者係儲存一或更多不同的資料類型，諸如純量整數、純量浮點、緊縮整數、緊縮浮點、向量整數、向量浮點、狀態 (例如，其為待執行之下一指令的位址之指令指針)，等等。於一實施例中，實體暫存器檔案單元 858 包含向量暫存器單元、寫

入遮罩暫存器單元、及純量暫存器單元。這些暫存器單元可提供架構向量暫存器、向量遮罩暫存器、及通用暫存器。實體暫存器檔案單元 858 被收回單元 854 疊置以說明各種方式，其中暫存器重新命名及失序執行可被實施（例如，使用重排序緩衝器及收回暫存器檔案；使用未來檔案、歷史緩衝器、及收回暫存器檔案；使用暫存器映圖及一群暫存器，等等）收回單元 854 及實體暫存器檔案單元 858 被耦合至執行叢集 860。執行叢集 860 包括一組一或更多執行單元 862 及一組一或更多記憶體存取單元 864。執行單元 862 可履行各種操作（例如，位移、相加、相減、相乘）並針對各種類型的資料（例如，純量浮點、緊縮整數、緊縮浮點、向量整數、向量浮點）。雖然某些實施例可包括專用於特別功能或功能集的數個執行單元，而其他實施例可包括僅有一個執行單元或者多個均履行所有功能之執行單元。排程器單元 856、實體暫存器檔案單元 858、執行叢集 860 被顯示為可能多數的，因位某些實施例係產生分離的管線給某些類型的資料/操作（例如，純量整數管線；純量浮點/緊縮整數/緊縮浮點/向量整數/向量浮點管線；及/或各具有其本身的排程器單元、實體暫存器檔案單元、及/或執行叢集的記憶體存取管線—而於分離記憶體存取管線之情況下，實施某些實施例，其中僅有此管線之執行叢集具有記憶體存取單元 864）。亦應理解其中使用分離管線時，一或更多這些管線可為失序發出/執行而其他的為依序。

該組記憶體存取單元 864 被耦合至記憶體單元 870，其包括資料 TLB 單元 872，其耦合至資料快取單元 874，其耦合至第二階（L2）快取單元 876。於一範例實施例中，記憶體存取單元 864 可包括載入單元、儲存位址單元、及儲存資料單元，其每一者係耦合至記憶體單元 870 中之資料 TLB 單元 872。指令快取單元 834 被進一步耦合至記憶體單元 870 中之第二階（L2）快取單元 876。L2 快取單元 876 被耦合至一或更多其他階的快取且最終耦合至主記憶體。

舉例而言，範例暫存器重新命名、失序發出/執行核心架構可實施管線 800 如下：1) 指令提取 838 履行提取和長度解碼級 802 和 804；2) 解碼單元 840 履行解碼級 806；3) 重新命名/配置器單元 852 履行配置級 808 和重新命名級 810；4) 排程器單元 856 履行排程級 812；5) 實體暫存器檔案單元 858 和記憶體單元 870 履行暫存器讀取/記憶體讀取級 814；執行叢集 860 履行執行級 816；6) 記憶體單元 870 和實體暫存器檔案單元 858 履行寫回/記憶體寫入級 818；7) 各種單元可被關聯於異常處置級 822；及 8) 收回單元 854 和實體暫存器檔案單元 858 履行確定級 824。

核心 890 可支援一或更多指令集（例如，x86 指令集（具有某些已隨著較新版本而加入之擴充）；MIPS Technologies of Sunnyvale, CA 之 MIPS 指令集；ARM Holdings of Sunnyvale, CA 之 ARM 指令集（具有諸如

NEON 等選擇性額外擴充))，包括文中所述之指令。於一實施例中，核心 890 包括邏輯以支援緊縮資料指令集擴充（例如，AVX1、AVX2，及/或一般性向量友善指令格式 (U=0 及/或 U=1) 之某形式，如先前所描述），藉此容許由許多多媒體應用程式所使用之操作得以使用緊縮資料來履行。

應理解其核心可支援多執行緒 (multi-threading) (指令二或更多組的操作或執行緒)，及可用包括時間切割多執行緒、同時多執行緒 (其中單一實體核心提供邏輯核心給其實體核心正在同時多執行緒之每一線程)、或其組合之各種方式來進行 (例如，時間切割的提取和解碼以及之後的同時多執行緒，諸如於 Intel 的超執行緒技術)。

雖然暫存器重新命名被描述於失序執行之情境，應理解其暫存器重新命名可被使用於依序架構。雖然處理器之例示實施例亦包括分離指令和資料快取單元 834/874 以及共用的第二階 (L2) 快取單元 876，但替代實施例可具有用於指令和資料之單一內部快取，諸如 (例如) 第一階 (L1) 內部快取、或多階的內部快取。於某些實施例中，系統可包括內部快取與核心及/或處理器外之外部快取的組合。替代地，所有快取可於核心及/或處理器之外。

特定範例依序核心架構

圖 9A-B 說明更特定的範例依序核心架構之方塊圖，

該核心將為一晶片中之數個邏輯區塊（包括相同類型及/或不同類型的其他核心）之一。邏輯區塊透過高頻寬互連網路（例如，環狀網路）而通連與某固定功能邏輯、記憶體 I/O 介面、及其他必要的 I/O 邏輯，根據應用而定。

圖 9A 為依據本發明之實施例的單一處理器核心之方塊圖，連同其連接至晶粒上互連網路 902 且具有其第二階（L2）快取之局部子集 904。於一實施例中，指令解碼器 900 支援具有緊縮資料指令集擴充之 x86 指令集。L1 快取 906 容許針對快取記憶體之低潛時存取進入純量及向量單元。雖然於一實施例中（為了簡化設計），純量單元 908 和向量單元 910 係使用分離的暫存器組（個別地，純量暫存器 912 和向量暫存器 914）且於其間轉移之資料被寫入至記憶體並從第一階（L1）快取 906 讀回，但本發明之替代實施例亦可使用不同的方式（例如，使用單一暫存器組或包括一通訊路徑，其容許資料被轉移於兩暫存器檔案之間而不被寫入或讀回）。

L2 快取之局部子集 904 為劃分為分離之局部子集（每一處理器核心一個）的總體 L2 快取之部分。各處理器核心具有通至 L2 快取 904 之其本身局部子集的直接存取路徑。由處理器核心所讀取之資料被儲存於其 L2 快取子集 904 中並可被快速地存取，平行與存取其本身局部 L2 快取子集之其他處理器核心。由處理器核心所寫入之資料被儲存於其本身的 L2 快取子集 904 且被清除自其他子集（假如需要的話）。環狀網路確保共用資料之相干

(coherency)。環狀網路為雙向的，以容許諸如處理器核心、L2 快取及其他邏輯區塊等代理器於晶片內彼此通連。各環狀資料路徑為 1012 位元寬於每方向。

圖 9B 為依據本發明之實施例的圖 9A 中之處理器核心的部分之擴充視圖。圖 9B 包括 L1 快取 904 之 L1 資料快取 906A 部分，以及有關向量單元 910 及向量暫存器 914 之更多細節。明確地，向量單元 910 為 16 寬的向量處理單元 (VPU) (參見 16 寬的 ALU 928)，其執行整數、單一精確浮點與雙精確浮點指令之一或更多者。VPU 支援：利用拌和單元 920 以拌和暫存器輸入、利用數字轉換單元 922A-B 之數字轉換、及利用記憶體輸入上之複製單元 924 的複製。寫入遮罩暫存器 926 容許闡述所得的向量寫入。

具有集成記憶體控制器及圖形之處理器

圖 10 為依據本發明之實施例的處理器 1000 之方塊圖，該處理器可具一個以上的核心、可具有一集成記憶體控制器、且可具有集成圖形。圖 10 中之實線方塊係說明一具有單一核心 1002A、系統代理器 1010、一組一或更多匯流排控制器單元 1016 之處理器 1000，而虛線方塊之額外加入則說明一具有多個核心 1002A-N、系統代理器 1010 中之一組一或更多集成記憶體控制器單元 1014、及特殊用途邏輯 1008 之替代處理器 1000。

因此，處理器 1000 之不同實施方式可包括：1) 具有

特殊用途邏輯 1008 之 CPU 為集成圖形及 / 或科學（通量）邏輯（其可包括一或更多核心），而核心 1002A-N 為一或更多通用核心（例如，通用依序核心、通用失序核心、兩者之組合）；2）具有核心 1002A-N 之共處理器為主要用於圖形及 / 或科學（通量）之大量特殊用途核心；及 3）具有核心 1002A-N 之共處理器為大量通用依序核心。因此，處理器 1000 可為通用處理器、共處理器或特殊用途處理器，諸如（例如）網路或通訊處理器、壓縮引擎、圖形處理器、GPGPU（通用圖形處理單元）、高通量多集成核心（MIC）共處理器（包括 30 或更多核心）嵌入處理器，等等。處理器可被實施於一或更多晶片上。處理器 1000 可為一或更多基底之一部分及 / 或可被實施於一或更多基底上，使用數種製程科技之任一種，諸如（例如）BiCMOS、CMOS、或 NMOS。

記憶體階層包括核心內之一或更多階的快取、一組或者一或更多共用快取單元 1006、及耦合至該組集成記憶體控制器單元 1014 之外部記憶體（未顯示）。該組共用快取單元 1006 可包括一或更多中階快取，諸如第二階（L2）、第三階（L3）、第四階（L4）、或其他階的快取、最後階快取（LLC）、及 / 或其組合。雖然於一實施例中，一種環狀為基的互連單元 1012 係互連集成圖形邏輯 1008、該組共用快取單元 1006、以及系統代理器單元 1010 / 集成記憶體控制器單元 1014，但替代實施例可使用任何眾所周知的技術來互連此等單元。於一實施例中，相

干性被維持於一或更多快取單元 1006 與核心 1002A-N 之間。

於某些實施例中，一或更多核心 1002A-N 能夠進行多執行緒。系統代理器 1010 包括那些協調及操作核心 1002A-N 之組件。系統代理器單元 1010 可包括（例如）電力控制單元（PCU）及顯示單元。PCU 可為或者可包括用以調節核心 1002A-N 和集成圖形邏輯 1008 之電力狀態所需的邏輯和組件。顯示單元係用以驅動一或更多外部連接的顯示。

就架構指令集而言，核心 1002A-N 可為同質的或異質的；亦即，二或更多核心 1002A-N 能夠執行相同的指令集，而其他者能夠執行該指令集之僅僅一子集或者一不同指令集。

範例電腦架構

圖 11-14 為範例電腦架構之方塊圖。用於筆記型電腦、桌上型電腦、手持式 PC、個人數位助理、工程工作站、伺服器、網路裝置、網路集線器、開關、嵌入式處理器、數位信號處理器（DSP）、圖形裝置、視頻遊戲裝置、機上盒、微處理器、行動電話、可攜式媒體播放器、手持式裝置、及各種其他電子裝置之習知技術中已知的其他系統設計和組態亦為適當的。一般而言，能夠結合處理器及/或其他執行邏輯之多種系統或電子裝置（如文中所揭示者）通常為適當的。

現在參考圖 11，其顯示依據本發明之一實施例的系統 1100 之方塊圖。系統 1100 可包括一或更多處理器 1110、1115，其被耦合至控制器集線器 1120。於一實施例中，控制器集線器 1120 包括一圖形記憶體控制器集線器（GMCH）1190 及一輸入/輸出集線器（IOH）1150（其可於分離的晶片上）；GMCH 1190 包括記憶體和圖形控制器，其係耦合記憶體 1140 和共處理器 1145；IOH 1150 將輸入/輸出（I/O）裝置 1160 耦合至 GMCH 1190。替代地，記憶體和圖形控制器之一或兩者被集成於處理器內（如文中所述者），記憶體 1140 和共處理器 1145 被直接耦合至處理器 1110、以及一具有 IOH 1150 之單一晶片中的控制器集線器 1120。

額外處理器 1115 之選擇性本質係以虛線被標示於圖 11 中。各處理器 1110、1115 可包括文中所述之一或更多處理核心並可為某版本的處理器 1000。

記憶體 1140 可為（例如）動態隨機存取記憶體（DRAM）、相位改變記憶體（PCM）、或兩者之組合。針對至少一實施例，控制器集線器 1120 通連與處理器 1110、1115，經由諸如前側匯流排（FSB）等多點（multi-drop）匯流排、諸如快速路徑互連（QPI）等點對點介面、或類似連接 1195。

於一實施例中，共處理器 1145 為特殊用途處理器，諸如（例如）高通量 MIC 處理器、網路或通訊處理器、壓縮引擎、圖形處理器、GPGPU、嵌入處理器，等等。於一

實施例中，控制器集線器 1120 可包括一集成圖形加速器。

有多種差異於實體資源 1110、1115 之間，關於包括架構、微架構、熱、電力耗損特性等等重要量度之波譜。

於一實施例中，處理器 1110 執行其控制一般類型之資料處理操作的指令。共處理器指令可嵌入指令之內。處理器 1110 識別這些共處理器指令為應由附加共處理器 1145 所執行之類型。因此，處理器 1110 將共處理器匯流排或其他互連上之這些共處理器指令（或代表共處理器指令之控制信號）發出至共處理器 1145。共處理器 1145 接受並執行所接收的共處理器指令。

現在參考圖 12，其顯示依據本發明之一實施例的第一更特定範例系統 1200 之方塊圖。如圖 12 中所示，多處理器系統 1200 為點對點互連系統，並包括經由點對點互連 1250 而耦合之第一處理器 1270 和第二處理器 1280。處理器 1270 與 1280 之每一者可為相同版本的處理器 1000。於本發明之一實施例中，處理器 1270 和 1280 個別為處理器 1110 和 1115，而共處理器 1238 為共處理器 1145。於另一實施例中，處理器 1270 和 1280 個別為處理器 1110 和共處理器 1145。

處理器 1270 和 1280 被顯示為個別地包括集成記憶體控制器（IMC）單元 1272 和 1282。處理器 1270 亦包括點對點（P-P）介面 1276 和 1278 為其匯流排控制器單元之部分；類似地，第二處理器 1280 包括 P-P 介面 1286 和

1288。處理器 1270、1280 可使用 P-P 介面電路 1278、1288 而經由點對點 (P-P) 介面 1250 以交換資訊。如圖 12 中所示，IMC 1272 和 1282 將處理器耦合至個別記憶體，亦即記憶體 1232 和記憶體 1234，其可為局部地裝附至個別處理器之主記憶體的部分。

處理器 1270、1280 可各使用點對點介面電路 1276、1294、1286、1298 而經由個別 P-P 介面 1252、1254 與晶片組 1290 交換資訊。晶片組 1290 可選擇性地經由高性能介面 1239 而與共處理器 1238 交換資訊。於一實施例中，共處理器 1238 為特殊用途處理器，諸如 (例如) 高通量 MIC 處理器、網路或通訊處理器、壓縮引擎、圖形處理器、GPGPU、嵌入處理器，等等。

共用快取 (未顯示) 可被包括於任一處理器中或者於兩處理器之外部，而經由 P-P 互連與處理器連接，以致處理器之局部快取資訊的任一者或兩者可被儲存於共用快取中，假如處理器被置於低電力模式下的話。

晶片組 1290 可經由介面 1296 而被耦合至第一匯流排 1216。於一實施例中，第一匯流排 1216 可為周邊組件互連 (PCI) 匯流排，或者諸如 PCI Express 匯流排或另一第三代 I/O 互連匯流排之匯流排，雖然本發明之範圍不因此受限。

如圖 12 中所示，各種 I/O 裝置 1214 可被耦合至第一匯流排 1216，連同一將第一匯流排 1216 耦合至第二匯流排 1220 之匯流排橋 1218。於一實施例中，諸如共處理

器、高通量 MIC 處理器、GPGPU、加速器（諸如，例如，圖形加速器或數位信號處理（DSP）單元）、場可編程閘極陣列、或任何其他處理器等一或更多額外處理器 1215 被耦合至第一匯流排 1216。於一實施例中，第二匯流排 1220 為低接腳數（LPC）匯流排。各種裝置可被耦合至第二匯流排 1220，包括（例如）鍵盤及/或滑鼠 1222、通訊裝置 1227 及儲存單元 1228，諸如碟片驅動器或其他大量儲存裝置，可包括指令/碼及資料 1230，於一實施例中。再者，聲頻 I/O 1224 可被耦合至第二匯流排 1220。注意：其他架構是可能的。例如，取代圖 12 之點對點架構，系統可實施多點匯流排其他此類架構。

現在參考圖 13，其顯示依據本發明之一實施例的第二更特定範例系統 1300 之方塊圖。圖 12 和 13 中之類似元件係使用類似的參考數字，且圖 13 已省略了圖 12 之某些形態以避免混淆圖 13 之其他形態。

圖 13 顯示其處理器 1270、1280 可個別地包括集成記憶體和 I/O 控制邏輯（「CL」）1272 和 1282。因此，CL 1272、1282 包括集成記憶體控制器單元並包括 I/O 控制邏輯。圖 13 顯示其不僅記憶體 1232、1234 耦合至 CL 1272、1282，同時 I/O 裝置 1314 亦耦合至控制邏輯 1272、1282。傳統 I/O 裝置 1315 被耦合至晶片組 1290。

現在參考圖 14，其顯示依據本發明之一實施例的 SoC 1400 之方塊圖。與圖 10 類似的元件係使用類似參考數字。同時，虛線方塊為更先進的 SoC 上之選擇性特徵。於

圖 14 中，互連單元 1402 被耦合至：應用程式處理器 1410，其包括一組一或更多核心 202A-N 及共用快取單元 1006；系統代理器單元 1010；匯流排控制器單元 1016；集成記憶體控制器單元 1014；一組一或更多共處理器 1420，其可包括集成圖形邏輯、影像處理器、聲頻處理器、和視頻處理器；靜態隨機存取記憶體（SRAM）單元 1430；直接記憶體存取（DMA）單元 1432；及顯示單元 1440，用以耦合至一或更多外部顯示。於一實施例中，共處理器 1420 包括特殊用途處理器，諸如（例如）網路或通訊處理器、壓縮引擎、GPGPU、高通量 MIC 處理器、嵌入處理器，等等。

文中所揭露之機構的實施例可被實施以硬體、軟體、韌體、或此類實施方式之組合。本發明之實施例可被實施為電腦程式或程式碼，其係執行在包括至少一處理器、儲存系統（包括揮發性和非揮發性記憶體及/或儲存元件）、至少一輸入裝置、及至少一輸出裝置之可編程系統上。

程式碼（諸如圖 12 中所示之碼 1230）可被應用於輸入指令，以履行文中所述之功能並產生輸出資訊。輸出資訊可被以已知方式應用於一或更多輸出裝置。為了本申請案，處理系統包括任何系統，其具有一處理器，諸如（例如）數位信號處理器（DSP）、微控制器、特殊應用積體電路（ASIC）、或微處理器。

程式碼可被實施以高階程序或物件導向的編程語言來

與處理系統通連。程式碼亦可被實施以組合或機械語言（假如需要的話）。事實上，文中所描述之機構對於任何特定編程語言並無範圍上之限制。於任何情況下，該語言可為編譯的或解讀的語言。

至少一實施例之一或更多形態可由機器可讀取媒體上所儲存之代表性指令來實施，該媒體代表處理器內之各種邏輯，當由機器所讀取時其致使機器製造用以履行文中所述之技術的邏輯。此類表示（已知為「IP 核心」）可被儲存於有形的、機器可讀取的媒體上且被供應至各個消費者或製造商，以供載入其實際上製造該邏輯或處理器之製造機器內。

此類機器可讀取儲存媒體可包括（無限制地）由機器或裝置所製造或形成之物件之非暫態的、有形的配置，包括：諸如硬碟、包括軟碟、光碟、微型碟片唯讀記憶體（CD-ROM）、微型碟片可寫入（CD-RW）和磁光碟之任何類型碟片等儲存媒體；諸如唯讀記憶體（ROM）、隨機存取記憶體（RAM），諸如動態隨機存取記憶體（DRAM）、靜態隨機存取記憶體（SRAM）、可抹除可編程唯讀記憶體（EPROM）、快閃記憶體、電可抹除可編程唯讀記憶體（EEPROM）、相位改變記憶體（PCM）等半導體裝置；磁或光學卡；或者適於儲存電子指令之任何其他類型的媒體。

因此，本發明之實施例亦包括非暫態的、有形的機器可讀取媒體，其含有指令或含有設計資料，諸如硬體描述

語言（HDL），其定義文中所述之結構、電路、設備、處理器及/或系統特徵。此類實施例亦可被稱為程式產品。

仿真（包括二元翻譯、碼編輯，等等）

於某些情況下，指令轉換器可被用以將來自來源指令集之指令轉換為目標指令集。例如，指令轉換器可將指令翻譯（例如，使用靜態二元翻譯、包括動態編譯之動態二元翻譯）、編輯、仿真、或者轉換為一或更多其他指令以供由核心所處理。指令轉換器可被實施以軟體、硬體、韌體、或其組合。指令轉換器可位於處理器上、處理器外、或部分於處理器上部分於處理器外。

圖 15 為方塊圖，其對比軟體指令轉換器之使用，以將來源指令集中之二元指令轉換為目標指令集中之二元指令，依據本發明之實施例。於例示之實施例中，指令轉換器為軟體指令轉換器，雖然替代地該指令轉換器可被實施以軟體、韌體、硬體、或其各種組合。圖 15 顯示高階語言 1502 之程式可使用 x86 編譯器 1504 來編譯以產生 x86 二元碼 1506，其可由一具有至少一 x86 指令集核心之處理器 1516 所本地執行。具有至少一 x86 指令集核心之處理器 1516 代表任何可履行如具有至少一 x86 指令集核心之 Intel 處理器的實質上相同功能之處理器，藉由相容地執行或者處理（1）Intel x86 指令集核心之指令集的基本部分或（2）用來運行於具有至少一 x86 指令集核心之 Intel 處理器上的應用程式或其他軟體的物件碼版本，以

便達成如具有至少一 x86 指令集核心之 Intel 處理器的實質上相同結果。x86 編譯器 1504 代表一種能夠產生 x86 二元碼 1506 (例如, 物件碼) 之編譯器, x86 二元碼 1506 可 (具有或不具有額外鏈路處理) 被執行於具有至少一 x86 指令集核心之處理器 1516 上。類似地, 圖 15 顯示高階語言 1502 之程式可使用替代指令集編譯器 1508 而被編譯以產生替代的指令集二元碼 1510, 其可由一不具有至少一 x86 指令集核心之處理器 1514 所本地執行 (例如, 具有執行 MIPS Technologies of Sunnyvale, CA 之 MIPS 指令集及 / 或 ARM Holdings of Sunnyvale, CA 之 ARM 指令集的核心之處理器)。指令轉換器 1512 被用以將 x86 二元碼 1506 轉換為可由不具有至少一 x86 指令集核心之處理器 1514 所本地執行的碼。此轉換的碼不太可能相同於替代的指令集二元碼 1510, 因為能夠執行此操作之指令轉換器是難以製造的; 然而, 該轉換的碼將完成一般操作並由來自替代指令集之指令所組成。因此, 指令轉換器 1512 代表軟體、韌體、硬體、或其組合, 其 (透過仿真、模擬或任何其他程序) 容許不具有 x86 指令集處理器或核心之處理器或其他電子裝置來執行 x86 二元碼 1506。

雖然圖形中之流程圖顯示由本發明之某些實施例所履行的操作之特定順序, 但應理解此順序為範例性的 (例如, 替代實施例可以不同順序履行該些操作、結合某些操作、重疊某些操作, 等等)。

雖然本發明已依數個實施例而描述, 但那些熟悉本項

技術人士將理解本發明不限於所述之實施例，可於後附申請專利範圍之精神及範圍內被實行以修飾及更改。此描述因而被視為說明性而非限制性。

【圖式簡單說明】

本發明可藉由參考下列描述及後附圖式而被最佳地瞭解，該些圖式被用以說明本發明之實施例。於圖式中：

圖 1 顯示依據一實施例之向量頻率壓縮指令的範例執行；

圖 2 顯示依據一實施例之向量頻率壓縮指令的範例執行；

圖 3 為一流程圖，其顯示依據一實施例藉由執行處理器中之向量頻率壓縮指令以將來自來源向量暫存器之值壓縮至目的地向量暫存器的範例操作；

圖 4A 顯示一範例 AVX 指令格式，其包括 VEX 前綴、真實運算碼欄位、Mod R/M 位元組、SIB 位元組、置換欄位、及 IMM8，依據一實施例；

圖 4B 顯示來自圖 4A 之哪些欄位組成全運算碼欄位及基礎操作欄位，依據一實施例；

圖 4C 顯示來自圖 4A 之哪些欄位組成暫存器指標欄位，依據一實施例；

圖 5A 為方塊圖，其說明依據本發明之實施例的一般性向量友善指令格式及其類別 A 指令模板；

圖 5B 為方塊圖，其說明依據本發明之實施例的一般

性向量友善指令格式及其類別 B 指令模板；

圖 6A 為方塊圖，其說明依據本發明之實施例的範例特定向量友善指令格式；

圖 6B 為方塊圖，其說明組成全運算碼欄位之特定向量友善指令格式的欄位，依據本發明之一實施例；

圖 6C 為方塊圖，其說明組成暫存器指標欄位之特定向量友善指令格式的欄位，依據本發明之一實施例；

圖 6D 為方塊圖，其說明組成擴增操作欄位之特定向量友善指令格式的欄位，依據本發明之一實施例；

圖 7 為依據本發明之一實施例的暫存器架構之方塊圖；

圖 8A 為一方塊圖，其說明依據本發明之實施例的範例依序的管線及範例暫存器重新命名的、失序的發出/執行管線兩者；

圖 8B 為一方塊圖，其說明依據本發明之實施例的處理器中所包括的依序架構核心之範例實施例及範例暫存器重新命名的、失序的發出/執行架構核心兩者；

圖 9A 為依據本發明之實施例的單一處理器核心之方塊圖，連同其連接至晶粒上互連網路且具有其第二階 (L2) 快取之局部子集；

圖 9B 為依據本發明之實施例的圖 9A 中之處理器核心的部分之擴充視圖；

圖 10 為依據本發明之實施例的一可具有一個以上核心、可具有集成記憶體控制器、及可具有集成圖形之處理

器的方塊圖；

圖 11 為依據本發明之一實施例的系統之方塊圖；

圖 12 為依據本發明之一實施例的第一更特定範例系統之方塊圖；

圖 13 為依據本發明之一實施例的第二更特定範例系統之方塊圖；

圖 14 為依據本發明之一實施例的晶片上系統（SoC）之方塊圖；及

圖 15 為方塊圖，其對比軟體指令轉換器之使用，以將來源指令集中之二元指令轉換為目標指令集中之二元指令，依據本發明之實施例。

【主要元件符號說明】

100：向量頻率壓縮指令

105：目的地運算元

110：向量遮罩 MASK

115：來源運算元

120：選擇性運算元

125：已使用的元件控制遮罩

130：已使用的元件計數

200：向量頻率壓縮指令

205：目的地

210：MASK

215：來源

- 220 : 即刻值
- 225 : 已使用的元件控制遮罩
- 230 : 已使用的元件計數
- 402 : VEX 前綴
- 405 : REX 欄位
- 415 : 運算元映圖欄位
- 420 : VEX.vvvv 欄位
- 425 : 前綴編碼欄位
- 430 : 真實運算碼欄位
- 440 : Mod R/M 欄位
- 442 : MOD 欄位
- 444 : Reg 欄位
- 446 : R/M 欄位
- 450 : SIB 位元組
- 452 : SS
- 454 : SIB.xxx
- 456 : SIB.bbb
- 462 : 置換欄位
- 464 : W 欄位
- 468 : VEX.L 大小欄位
- 472 : 即刻欄位 (IMM8)
- 474 : 全運算碼欄位
- 500 : 一般性向量友善指令格式
- 505 : 無記憶體存取

- 510：無記憶體存取、全捨入控制類型操作
- 512：無記憶體存取、寫入遮罩控制、部分捨入控制
類型操作
- 515：無記憶體存取、資料轉變類型操作
- 517：無記憶體存取、寫入遮罩控制、vsize 類型操作
- 520：記憶體存取
- 527：記憶體存取、寫入遮罩控制
- 540：格式欄位
- 542：基礎操作欄位
- 544：暫存器指標欄位
- 546：修飾符欄位
- 550：擴增操作欄位
- 552：阿爾發欄位
- 552A：RS 欄位
- 552A.1：捨入
- 552A.2：資料轉變
- 552B：逐出暗示欄位
- 552B.1：暫時
- 552B.2：非暫時
- 554：貝他欄位
- 554A：捨入控制欄位
- 554B：資料轉變欄位
- 554C：資料調處欄位
- 556：SAE 欄位

- 557A : RL 欄位
- 557A.1 : 捨入
- 557A.2 : 向量長度 (VSIZE)
- 557B : 廣播欄位
- 558 : 捨入操作控制欄位
- 559A : 捨入操作欄位
- 559B : 向量長度欄位
- 560 : 比率欄位
- 562A : 置換欄位
- 562B : 置換因數欄位
- 564 : 資料元件寬度欄位
- 568 : 類別欄位
- 568A : 類別 A
- 568B : 類別 B
- 570 : 寫入遮罩欄位
- 572 : 即刻欄位
- 574 : 全運算碼欄位
- 600 : 特定向量友善指令格式
- 602 : EVEX 前綴
- 605 : REX 欄位
- 610 : REX' 欄位
- 615 : 運算碼映圖欄位
- 620 : VVVV 欄位
- 625 : 前綴編碼欄位

- 630 : 真實運算碼欄位
- 640 : Mod R/M 欄位
- 642 : MOD 欄位
- 644 : Reg 欄位
- 646 : R/M 欄位
- 654 : SIB.xxx
- 656 : SIB.bbb
- 700 : 暫存器架構
- 710 : 向量暫存器
- 715 : 寫入遮罩暫存器
- 725 : 通用暫存器
- 745 : 純量浮點堆疊暫存器檔案
- 750 : MMX 緊縮整數平坦暫存器檔案
- 800 : 處理器管線
- 802 : 提取級
- 804 : 長度解碼級
- 806 : 解碼級
- 808 : 配置級
- 810 : 重新命名級
- 812 : 排程級
- 814 : 暫存器讀取 / 記憶體讀取級
- 816 : 執行級
- 818 : 寫回 / 記憶體寫入級
- 822 : 異常處置級

- 824 : 確定級
- 830 : 前端單元
- 832 : 分支預測單元
- 834 : 指令快取單元
- 836 : 指令翻譯旁看緩衝器 (TLB)
- 838 : 指令提取單元
- 840 : 解碼單元
- 850 : 執行引擎單元
- 852 : 重新命名 / 配置器單元
- 854 : 收回單元
- 856 : 排程器單元
- 858 : 實體暫存器檔案單元
- 860 : 執行叢集
- 862 : 執行單元
- 864 : 記憶體存取單元
- 870 : 記憶體單元
- 872 : 資料 TLB 單元
- 874 : 資料快取單元
- 876 : 第二階 (L2) 快取單元
- 890 : 處理器核心
- 900 : 指令解碼器
- 902 : 晶粒上互連網路
- 904 : 第二階 (L2) 快取之局部子集
- 906 : L1 快取

- 906A : L1 資料快取
- 908 : 純量單元
- 910 : 向量單元
- 912 : 純量暫存器
- 914 : 向量暫存器
- 920 : 拌和單元
- 922A-B : 數字轉換單元
- 924 : 複製單元
- 926 : 寫入遮罩暫存器
- 928 : 16 寬的 ALU
- 1000 : 處理器
- 1002A-N : 核心
- 1006 : 共用快取單元
- 1008 : 特殊用途邏輯
- 1010 : 系統代理器
- 1012 : 環狀為基的互連單元
- 1014 : 集成記憶體控制器單元
- 1016 : 匯流排控制器單元
- 1100 : 系統
- 1110, 1115 : 處理器
- 1120 : 控制器集線器
- 1140 : 記憶體
- 1145 : 共處理器
- 1150 : 輸入/輸出集線器 (IOH)

- 1160 : 輸入 / 輸出 (I/O) 裝置
- 1190 : 圖形記憶體控制器集線器 (GMCH)
- 1195 : 連接
- 1200 : 多處理器系統
- 1214 : I/O 裝置
- 1215 : 額外處理器
- 1216 : 第一匯流排
- 1218 : 匯流排橋
- 1220 : 第二匯流排
- 1222 : 鍵盤及 / 或滑鼠
- 1224 : 聲頻 I/O
- 1227 : 通訊裝置
- 1228 : 儲存單元
- 1230 : 指令 / 碼及資料
- 1232 : 記憶體
- 1234 : 記憶體
- 1238 : 共處理器
- 1239 : 高性能介面
- 1250 : 點對點互連
- 1252, 1254 : P-P 介面
- 1270 : 第一處理器
- 1272, 1282 : 集成記憶體控制器 (IMC) 單元
- 1276, 1278 : 點對點 (P-P) 介面
- 1280 : 第二處理器

- 1286, 1288 : P-P 介面
- 1290 : 晶片組
- 1294, 1298 : 點對點介面電路
- 1296 : 介面
- 1300 : 系統
- 1314 : I/O 裝置
- 1315 : 傳統 I/O 裝置
- 1400 : SoC
- 1402 : 互連單元
- 1410 : 應用程式處理器
- 1420 : 共處理器
- 1430 : 靜態隨機存取記憶體 (SRAM) 單元
- 1432 : 直接記憶體存取 (DMA) 單元
- 1440 : 顯示單元
- 1502 : 高階語言
- 1504 : x86 編譯器
- 1506 : x86 二元碼
- 1508 : 指令集編譯器
- 1510 : 指令集二元碼
- 1512 : 指令轉換器
- 1514 : 不具有至少一 x86 指令集核心之處理器
- 1516 : 具有至少一 x86 指令集核心之處理器

七、申請專利範圍：

1. 一種履行電腦處理器中之向量頻率壓縮指令的方法，包含：

提取包括來源運算元和目的地運算元之該向量頻率壓縮指令，其中該來源運算元指明一包括複數來源資料元件之來源向量暫存器，該些來源資料元件包括一或更多列相同的資料元件，其中該目的地運算元識別目的地向量暫存器且其中該些一或更多列相同值之每一者將被壓縮於該目的地向量暫存器中而成爲一值與列長度對；

解碼該提取的向量頻率壓縮指令；及

執行該解碼的向量頻率壓縮指令，其針對各來源資料元件致使一值被複製入該目的地向量暫存器以指示該來源資料元件之值，其中等於一壓縮值之一或更多列的一或更多來源資料元件被編碼於該目的地向量暫存器中而成爲該預定壓縮值，接續以該列之列長度。

2. 如申請專利範圍第 1 項之方法，其中該提取的向量頻率壓縮指令進一步包含其被編碼至一值與列長度對之該壓縮值。

3. 如申請專利範圍第 1 項之方法，其中當因爲該些來源資料元件不含其針對列長度編碼而被最佳化之值而使該些來源資料元件無法被壓縮入該目的地向量暫存器時，該執行該向量頻率壓縮指令進一步致使一例外被提出。

4. 如申請專利範圍第 1 項之方法，其中該執行該解碼的向量頻率壓縮指令進一步致使一值被寫入一已使用元件

指示器，以指示該目的地向量暫存器中之哪些元件在壓縮期間被寫入。

5.如申請專利範圍第 4 項之方法，其中該提取的向量頻率壓縮指令進一步包含一已使用元件指示器目的地，以指示該已使用元件指示器應被寫入何處。

6.如申請專利範圍第 1 項之方法，其中該提取的向量頻率壓縮指令進一步包含一控制遮罩，其指示將被複製至該目的地向量暫存器之來自該些來源資料元件之一或更多值。

7.如申請專利範圍第 6 項之方法，其中該執行該解碼的向量頻率壓縮指令進一步致使藉由讀取該控制遮罩以判斷該壓縮值。

8.一種處理器核心，包含：

一硬體解碼單元，用以解碼向量頻率壓縮指令，其中該向量頻率壓縮指令包括來源運算元和目的地運算元，其中該來源運算元指明一包括複數來源資料元件之來源向量暫存器，該些來源資料元件包括一或更多列相同的資料元件，其中該目的地運算元識別目的地向量暫存器且其中該些一或更多列相同值之每一者將被壓縮於該目的地向量暫存器中而成爲一值與列長度對；及

執行引擎單元，用以執行該解碼的向量頻率壓縮指令，其針對各來源資料元件致使一值被複製入該目的地向量暫存器以指示該來源資料元件之值，其中等於一壓縮值之一或更多列之一或更多來源資料元件被編碼於該目的地

向量暫存器中而成爲該預定壓縮值，接續以該列之列長度。

9.如申請專利範圍第 8 項之處理器核心，其中該向量頻率壓縮指令進一步包含其被編碼至一值與列長度對之該壓縮值。

10.如申請專利範圍第 8 項之處理器核心，該執行單元進一步致使一例外被提出，當因爲該些來源資料元件不含其針對列長度編碼而被最佳化之值而使該些來源資料元件無法被壓縮入該目的地向量暫存器時。

11.如申請專利範圍第 8 項之處理器核心，其中該執行單元進一步致使一值被寫入一已使用元件指示器，以指示該目的地向量暫存器中之哪些元件在壓縮期間被寫入。

12.如申請專利範圍第 11 項之處理器核心，其中該向量頻率壓縮指令進一步包含一已使用元件指示器目的地，以指示該已使用元件指示器應被寫入何處。

13.如申請專利範圍第 8 項之處理器核心，其中該向量頻率壓縮指令進一步包含一控制遮罩，其指示將被複製至該目的地向量暫存器之來自該些來源資料元件的一或更多值。

14.如申請專利範圍第 13 項之處理器核心，該執行單元進一步致使藉由讀取該控制遮罩以判斷該壓縮值。

15.一種製造物件，包含：

一有形機器可讀取儲存媒體，其上儲存有向量頻率壓縮指令，其中該向量頻率壓縮指令包括來源運算元和目的

地運算元，其中該來源運算元指明一包括複數來源資料元件之來源向量暫存器，該些來源資料元件包括一或更多列相同的資料元件，其中該目的地運算元識別目的地向量暫存器且其中該些一或更多列相同值之每一者將被壓縮於該目的地向量暫存器中而成爲一值與列長度對；及

其中該向量頻率壓縮指令包括一運算碼，其指示機器執行該指令，該指令針對各來源資料元件致使一值被複製入該目的地向量暫存器以指示該來源資料元件之值，其中等於一壓縮值之一或更多列的一或更多來源資料元件被編碼於該目的地向量暫存器中而成爲該預定壓縮值，接續以該列之列長度。

16.如申請專利範圍第 15 項之製造物件，其中該向量頻率壓縮指令進一步包含其被編碼至一值與列長度對之該壓縮值。

17.如申請專利範圍第 15 項之製造物件，其中當因爲該些來源資料元件不含其針對列長度編碼而被最佳化之值而使該些來源資料元件無法被壓縮入該目的地向量暫存器時，該向量頻率壓縮指令進一步致使該機器提出一例外。

18.如申請專利範圍第 15 項之製造物件，其中該向量頻率壓縮指令進一步致使該機器將一值寫入一已使用元件指示器，以指示該目的地向量暫存器中之哪些元件在壓縮期間被寫入。

19.如申請專利範圍第 18 項之製造物件，其中該向量頻率壓縮指令進一步包含一已使用元件指示器目的地，以

指示該已使用元件指示器應被寫入何處。

20.如申請專利範圍第 15 項之製造物件，其中該向量頻率壓縮指令進一步包含一控制遮罩，其指示將被複製至該目的地向量暫存器之來自該些來源資料元件的一或更多值。

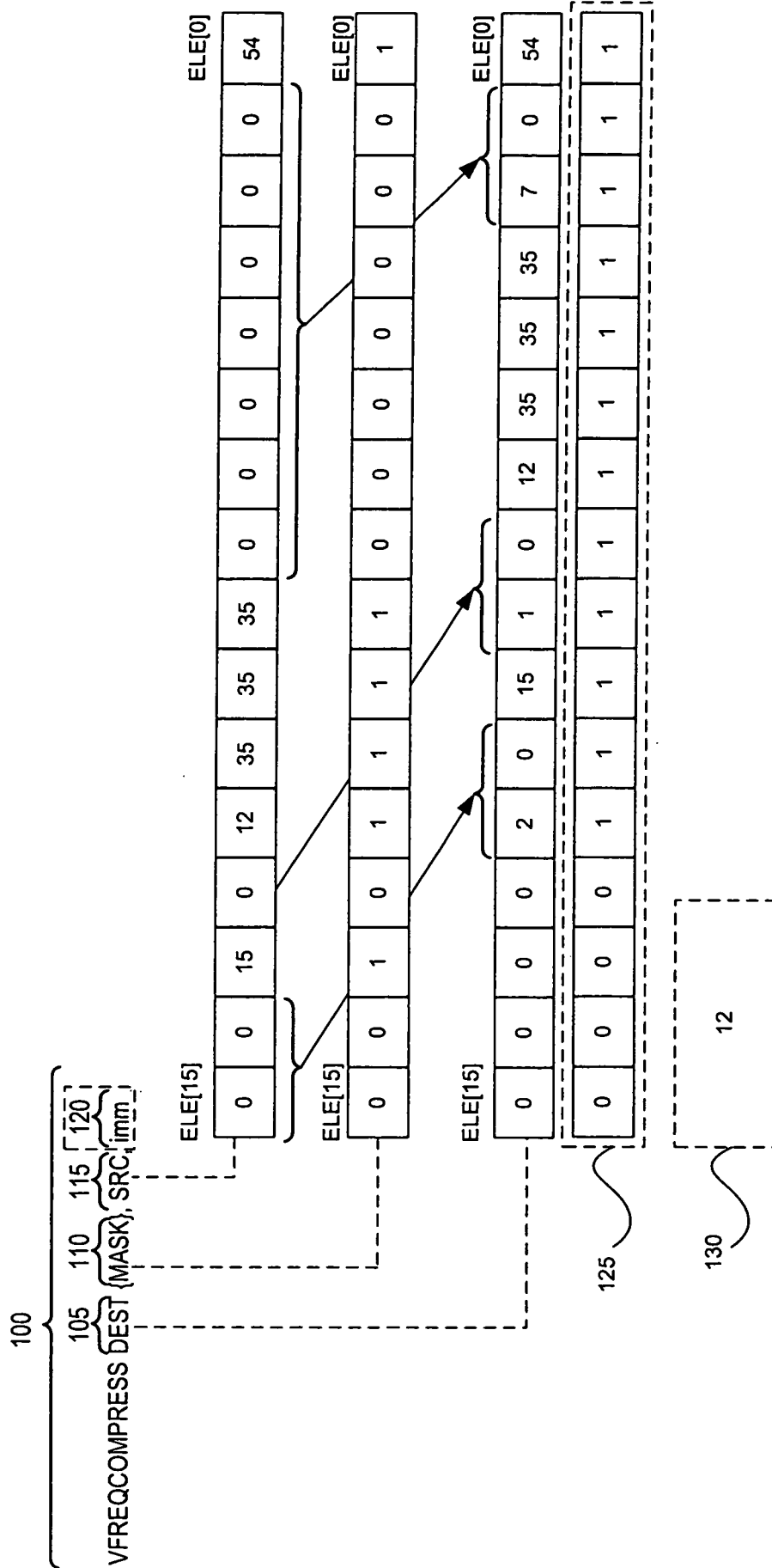


圖 1

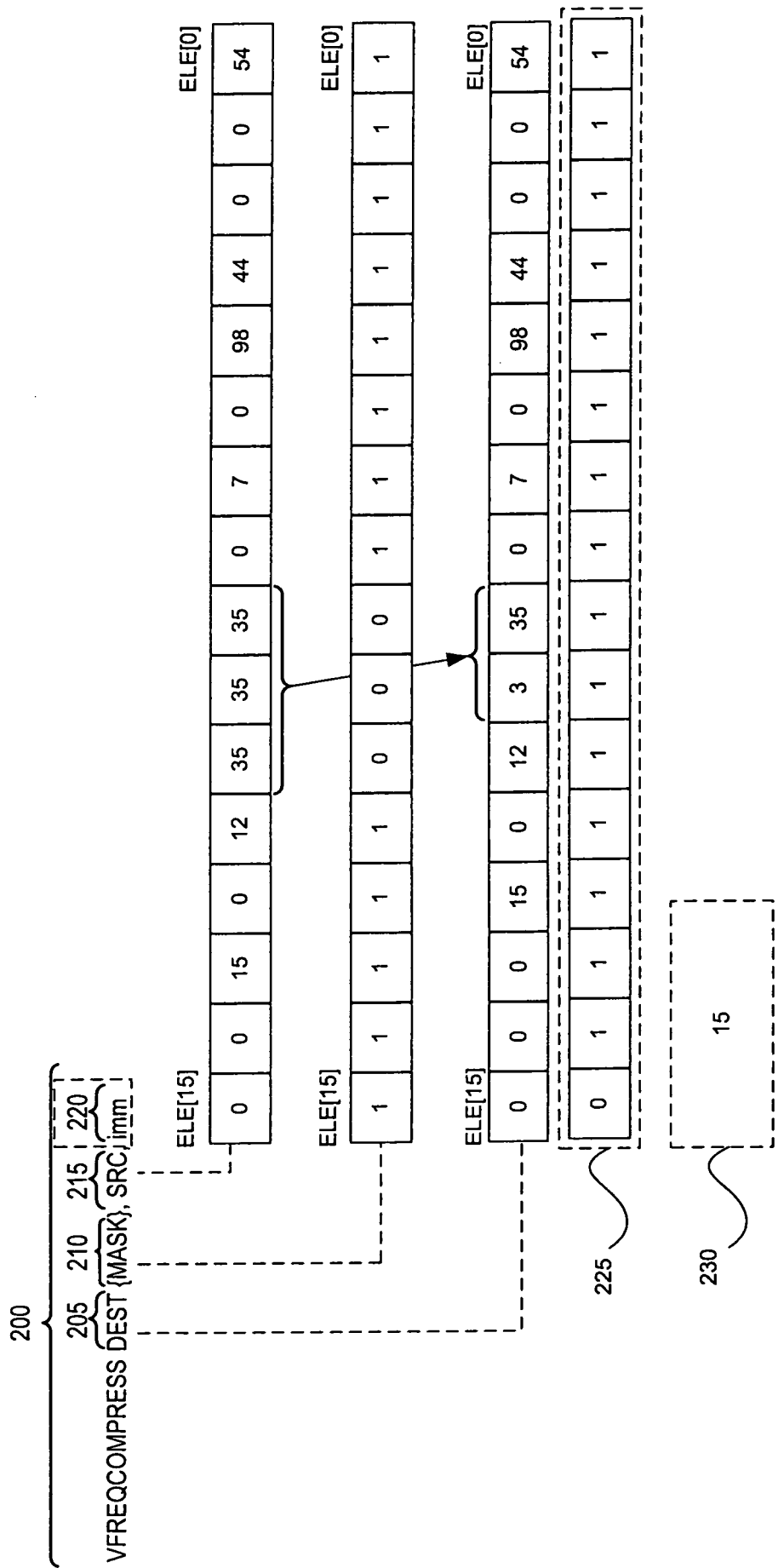


圖2

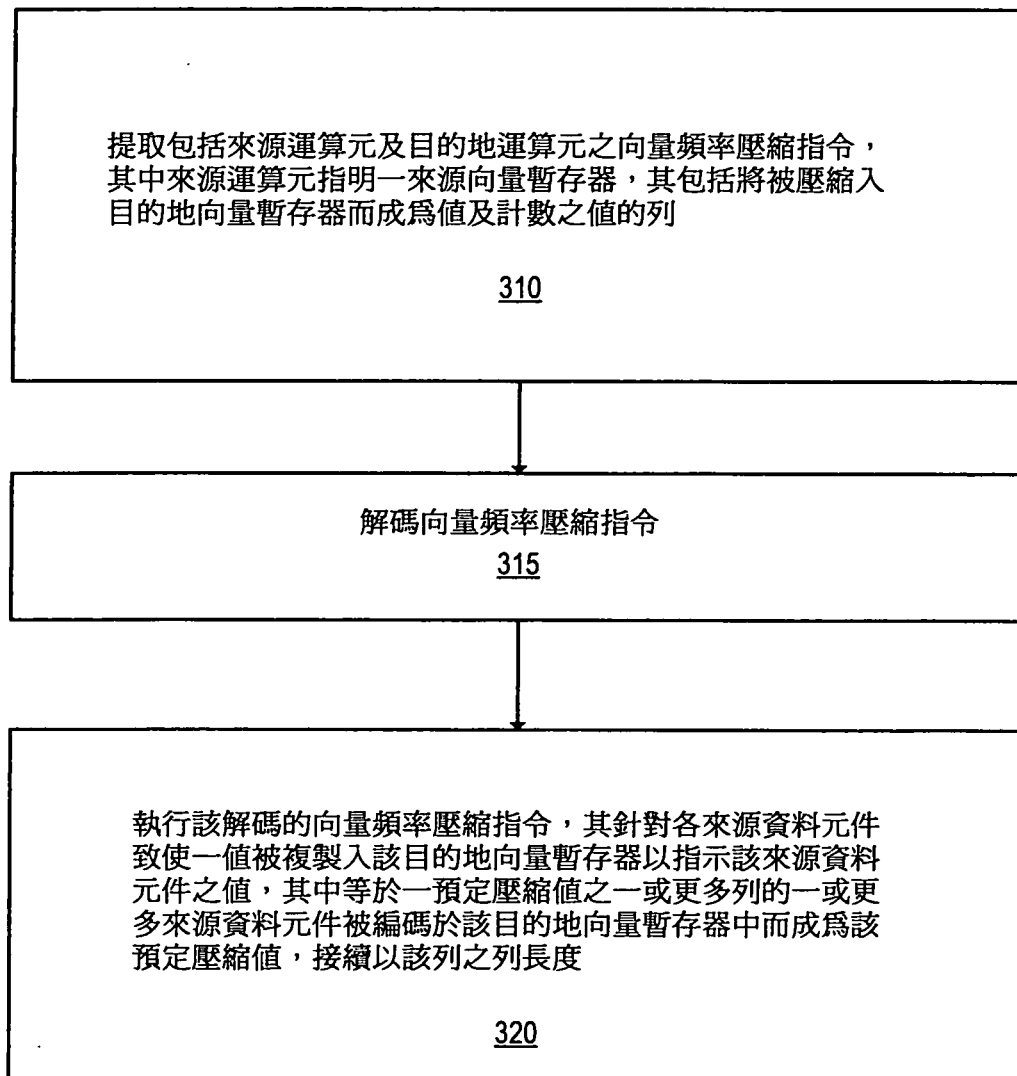


圖 3

圖 4A

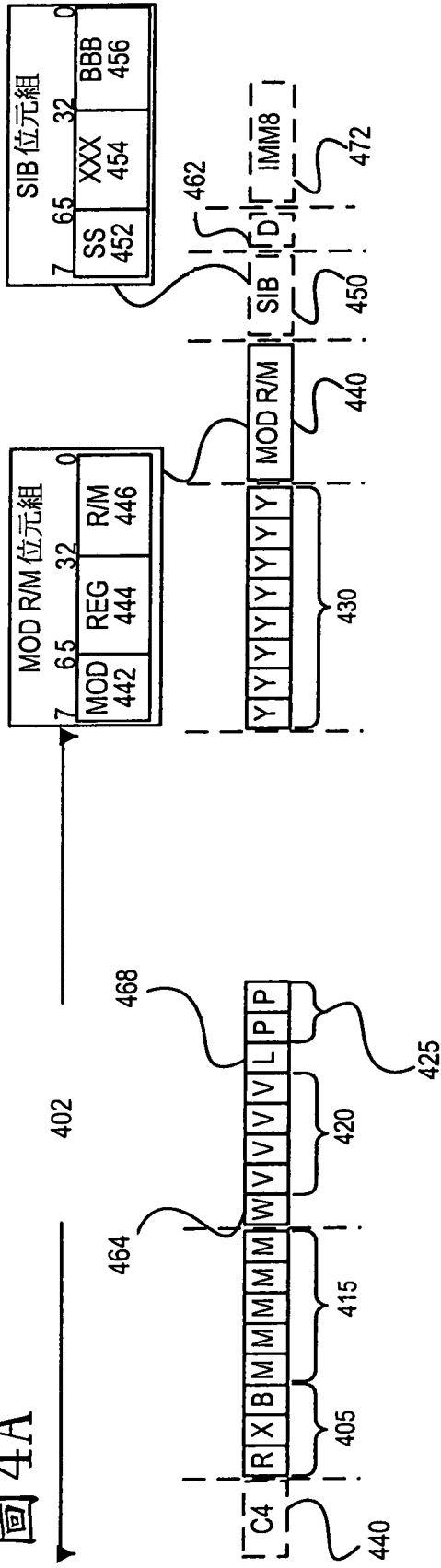


圖 4B

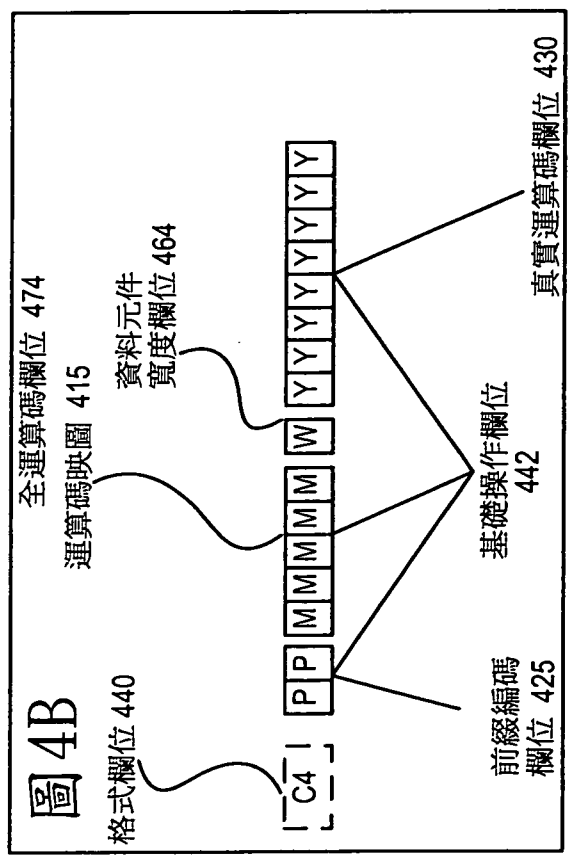


圖 4C

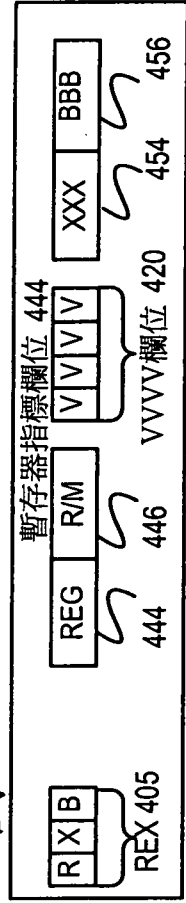


圖5A

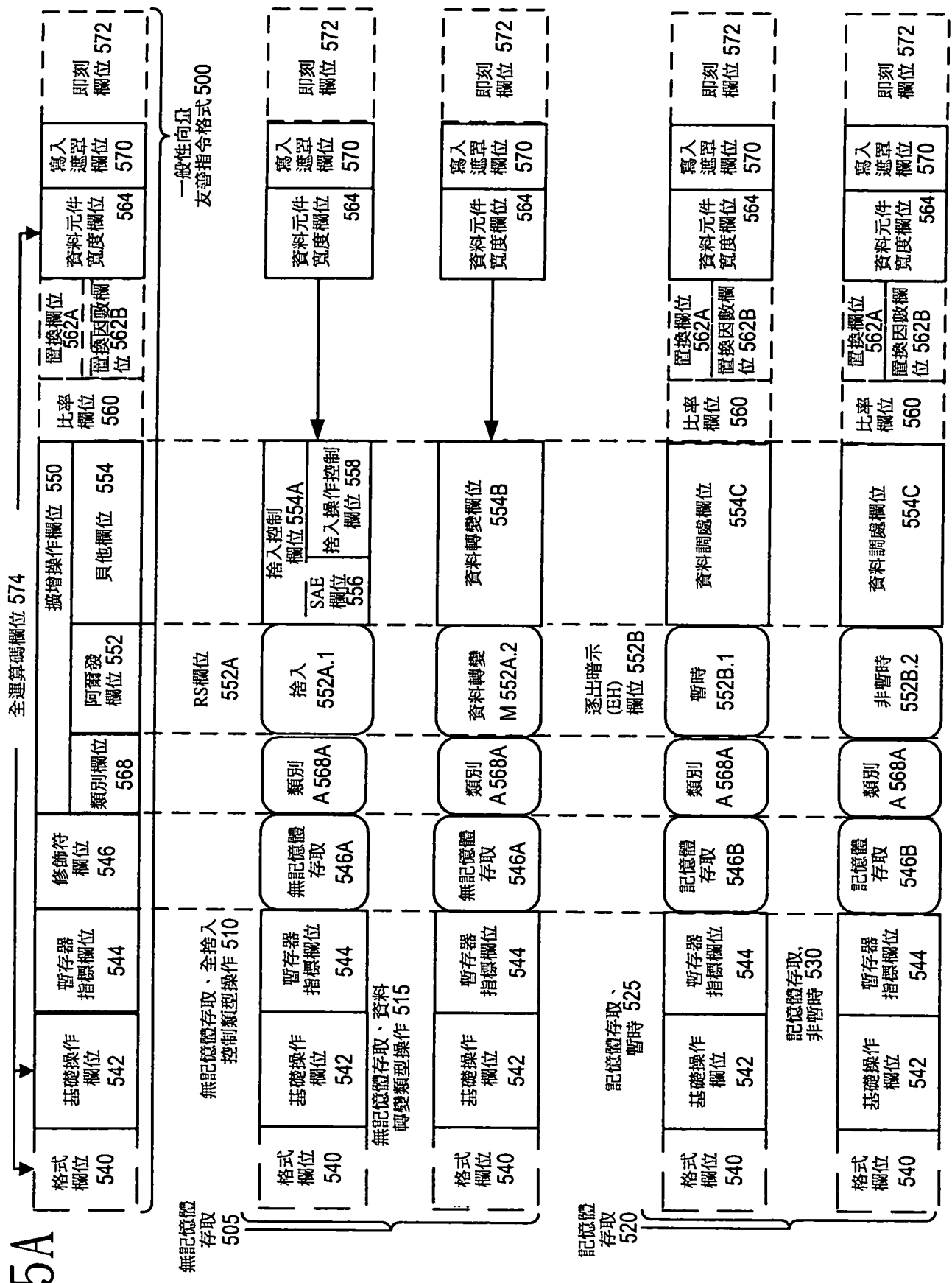


圖 5B

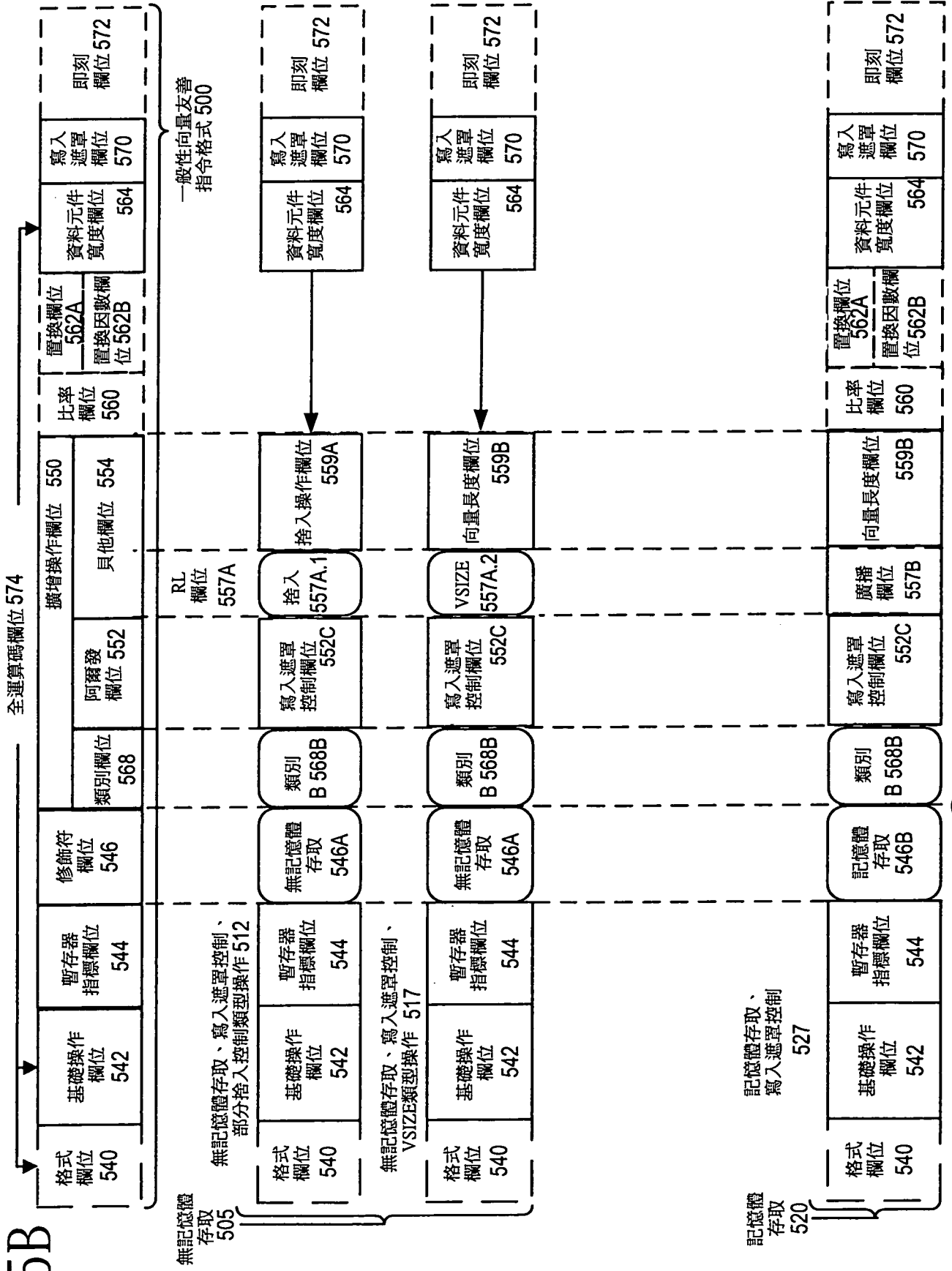
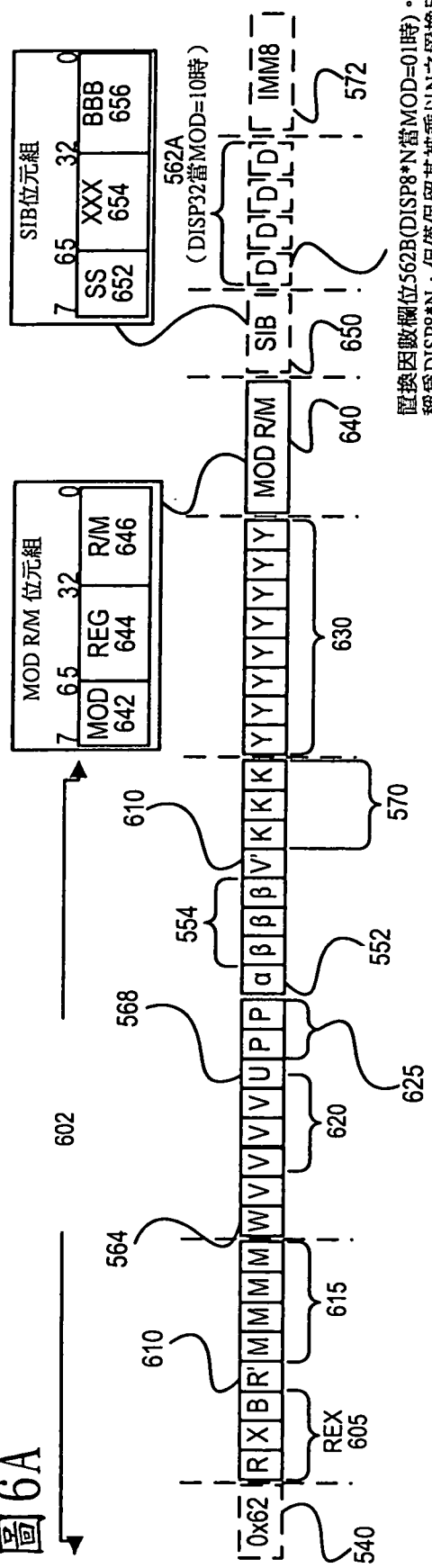


圖 6A



置換因數欄位 562B (DISP8*N 當 MOD=0 時)。稱爲 DISP8*N，但僅保留其被乘以 N 之置換因數。

圖 6B

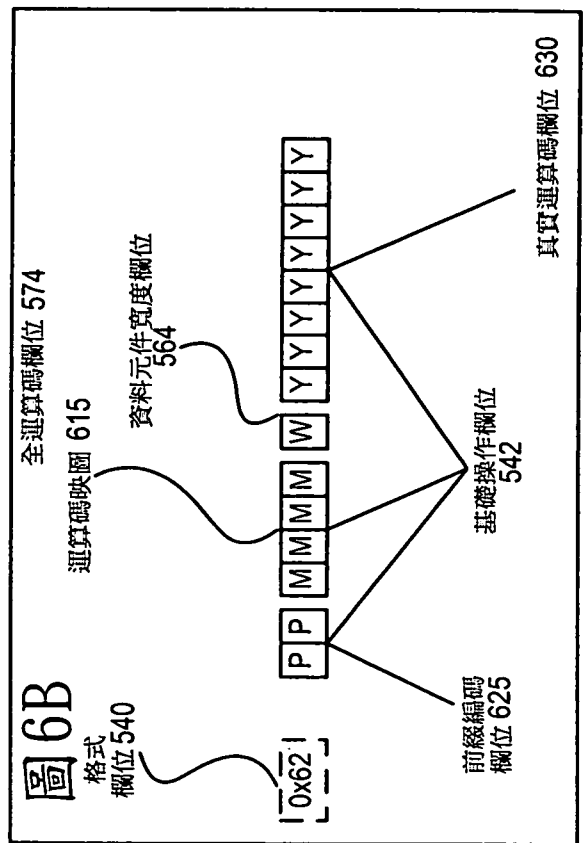


圖 6C

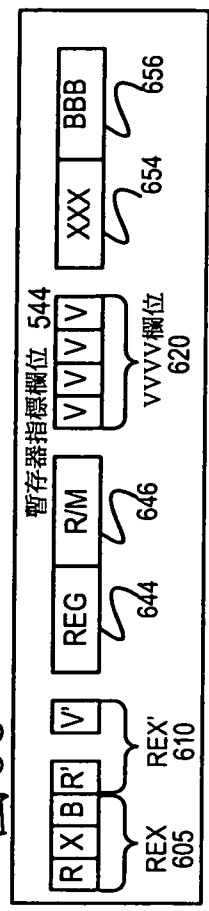
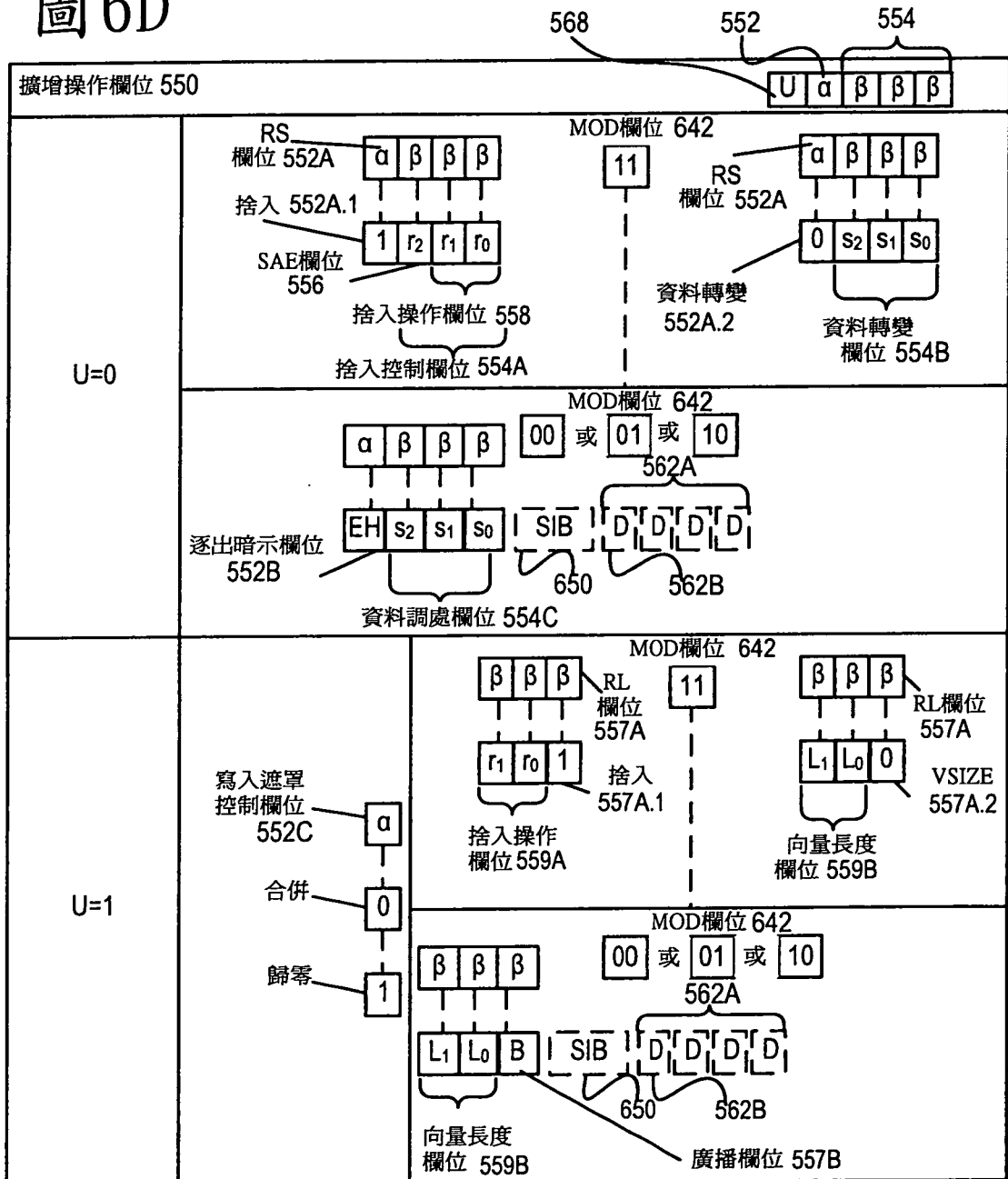
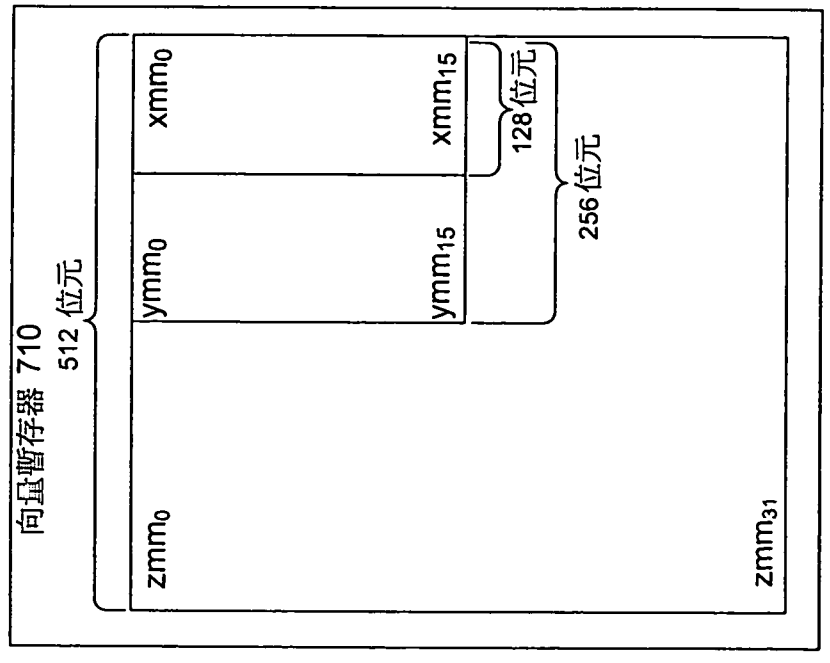
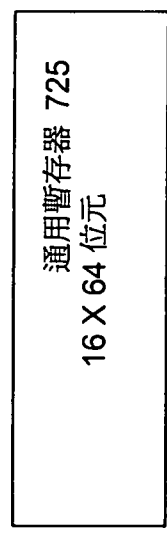


圖 6D

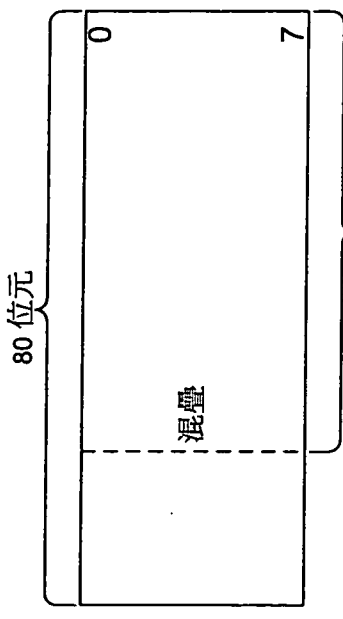


暫存器架構 700

圖 7



純量FP堆疊暫存器檔案 745
(X87FP)



64 位元
MMX緊縮整數平坦
暫存器檔案 750

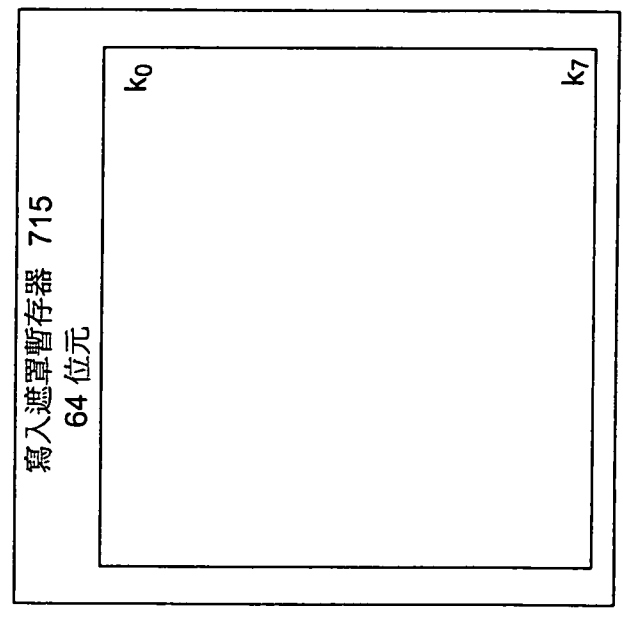
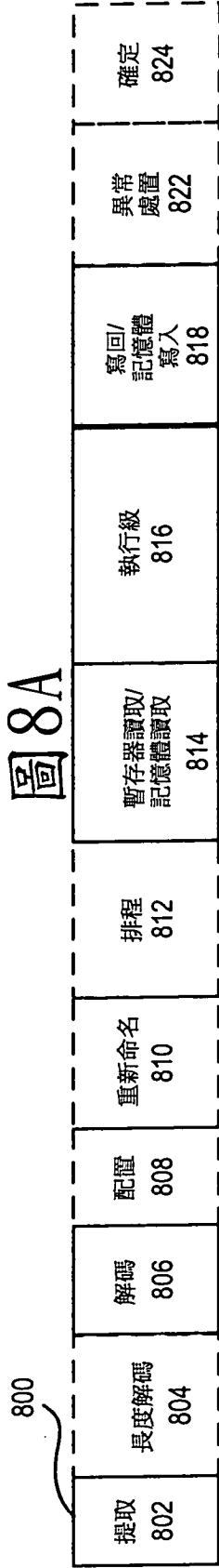


圖8A



890

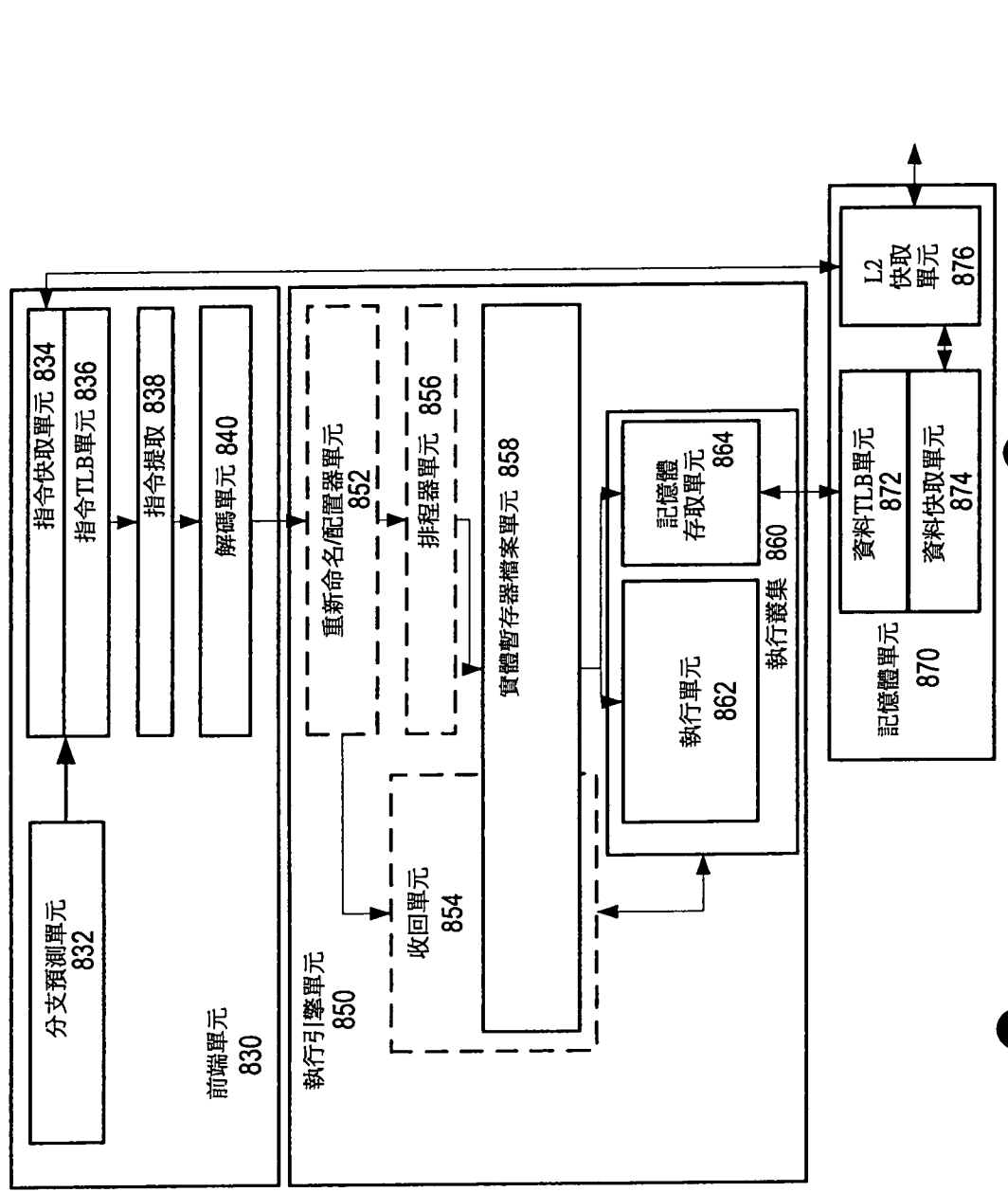


圖8B

圖9A

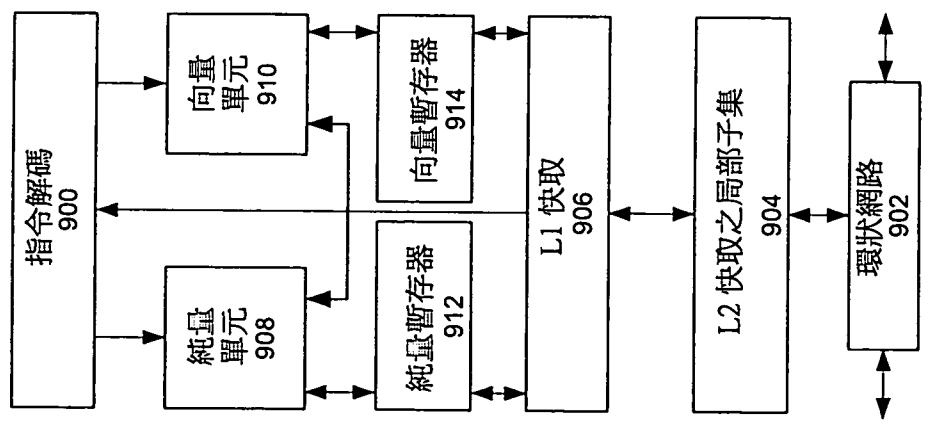
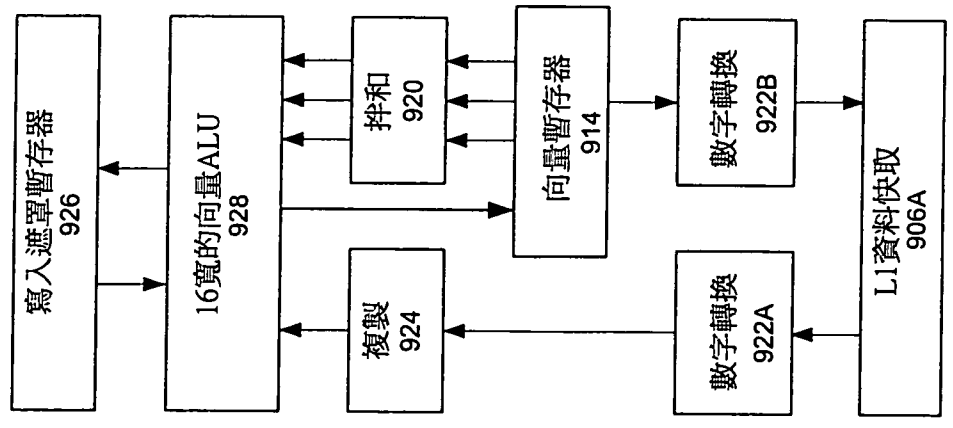


圖9B



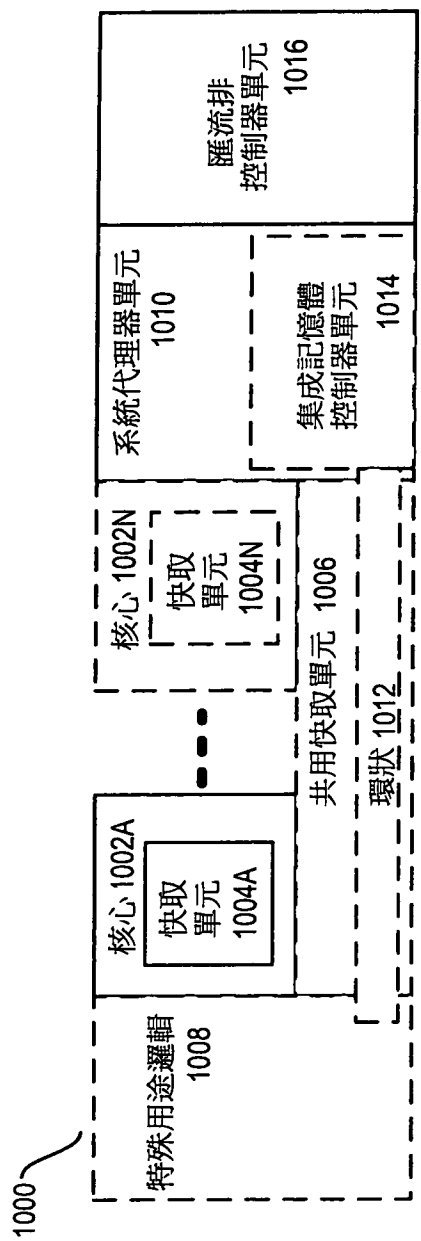


圖 10

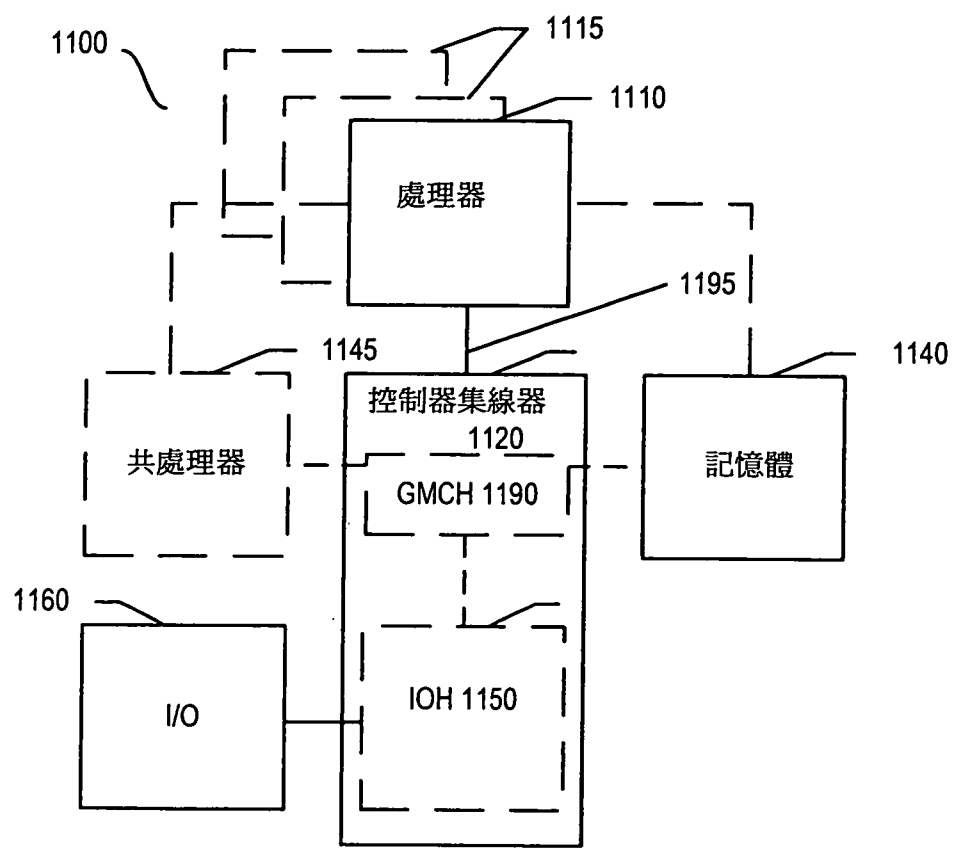
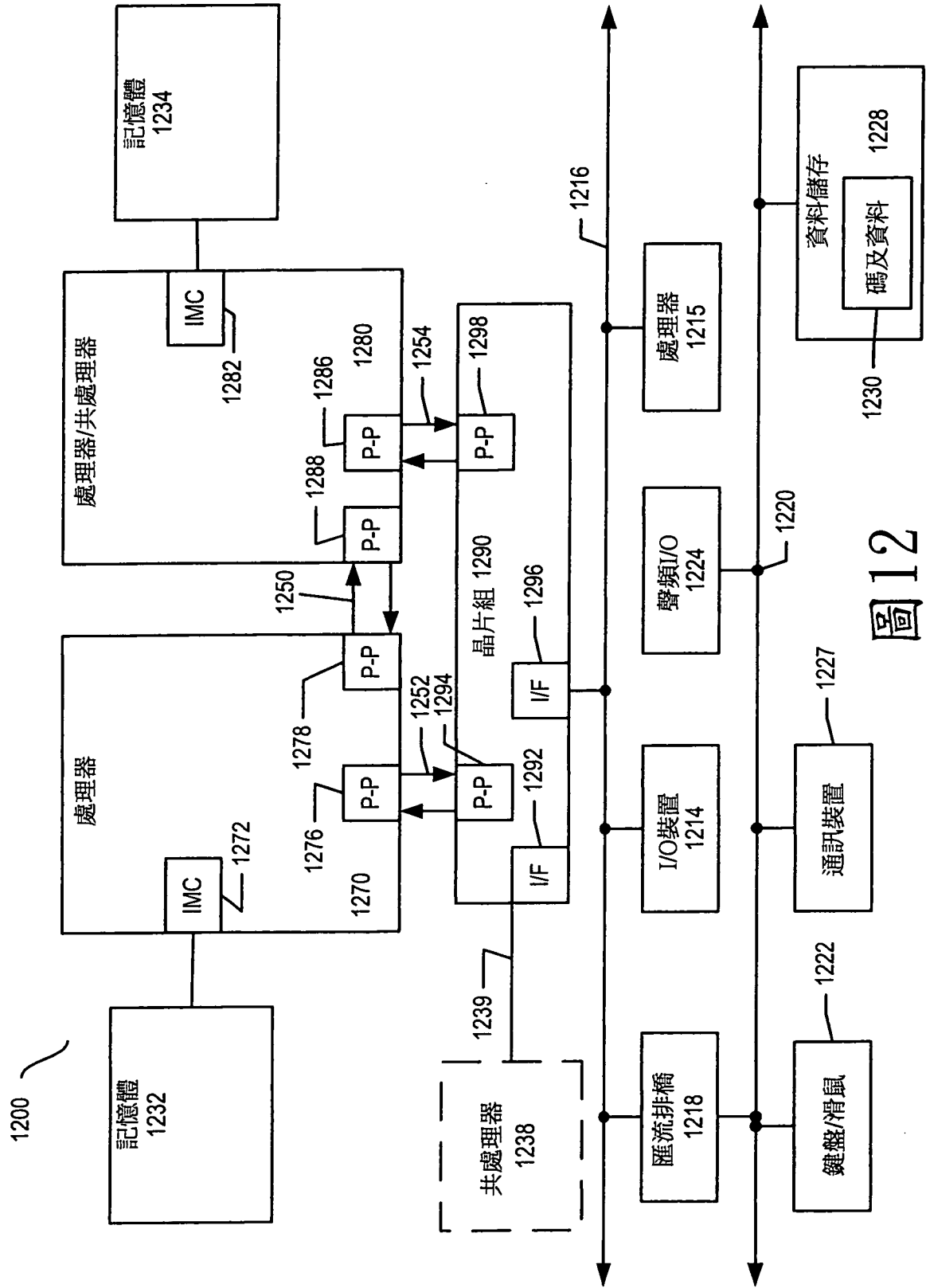


圖 11



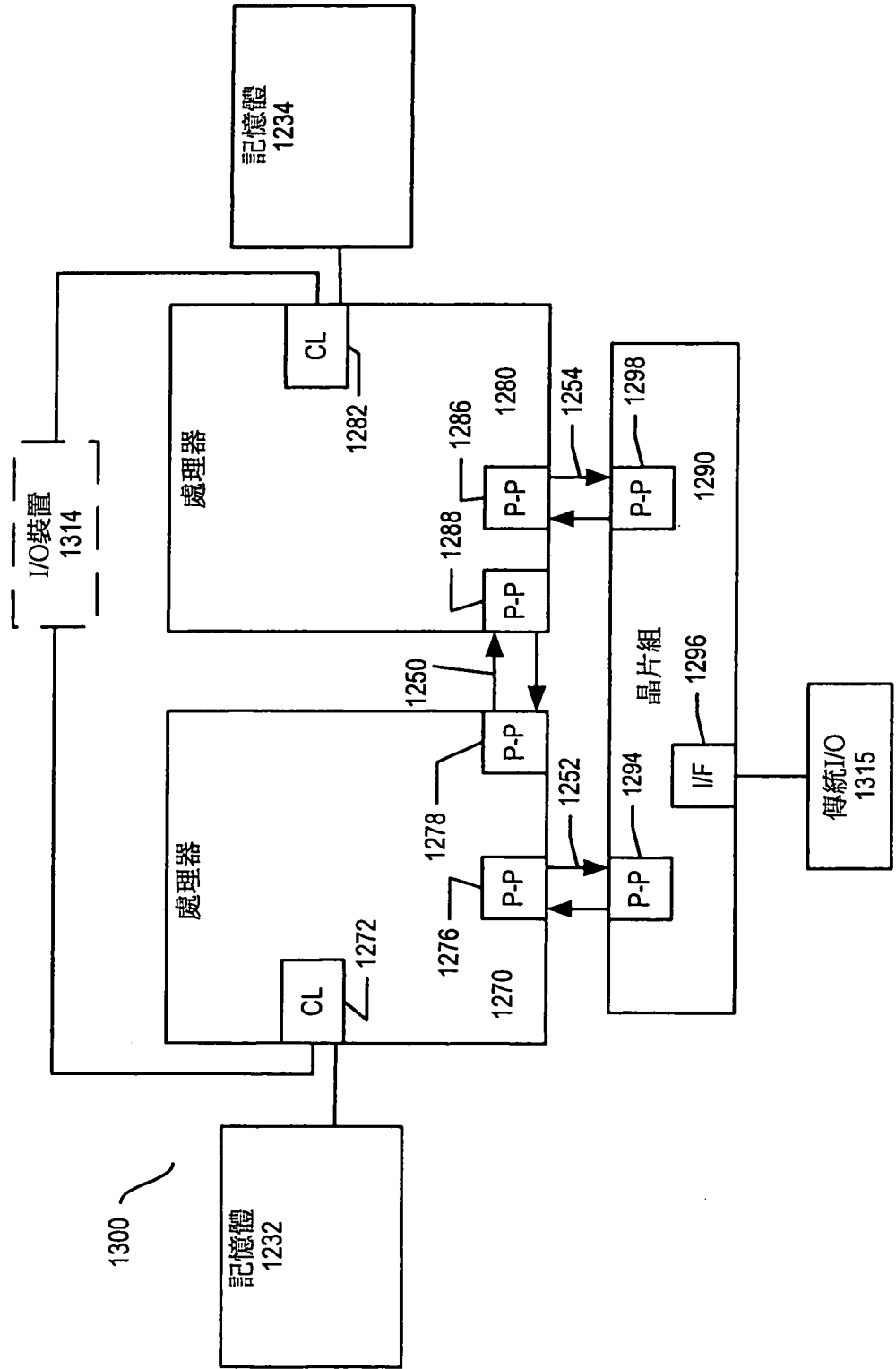
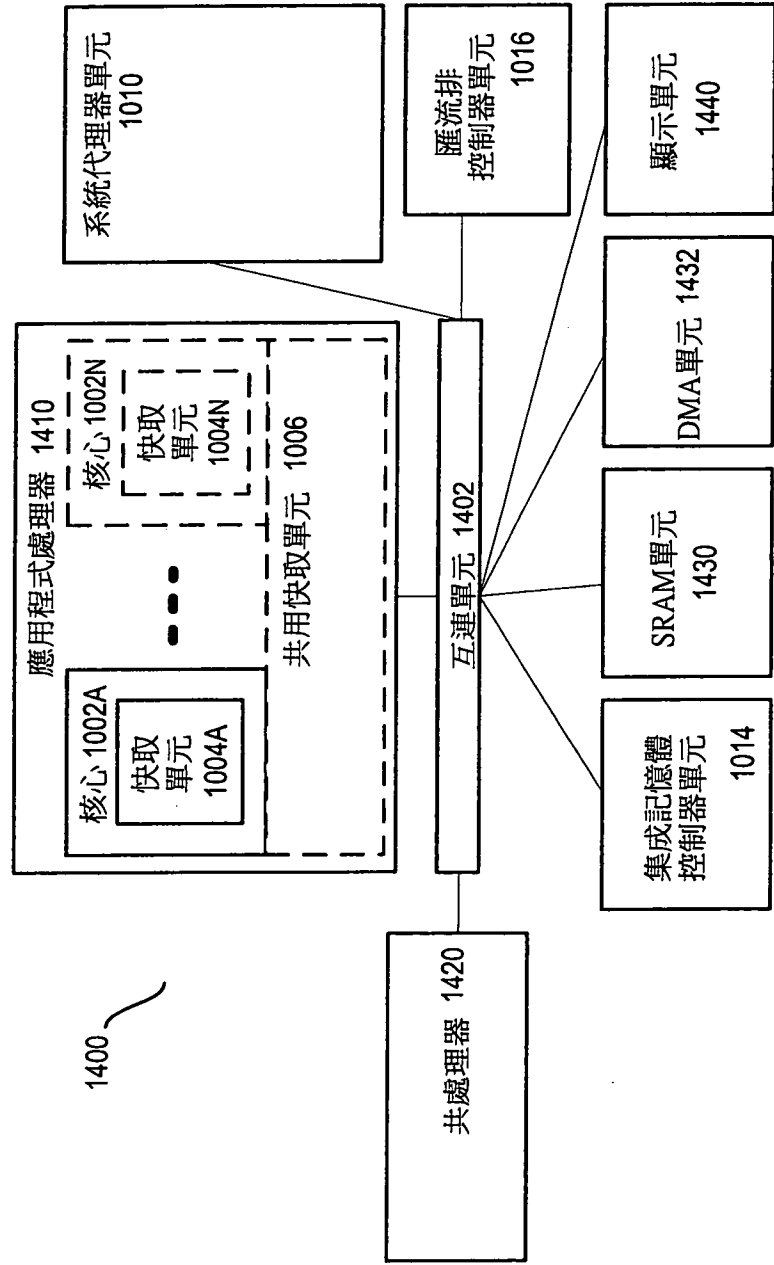


圖13



1400

圖14

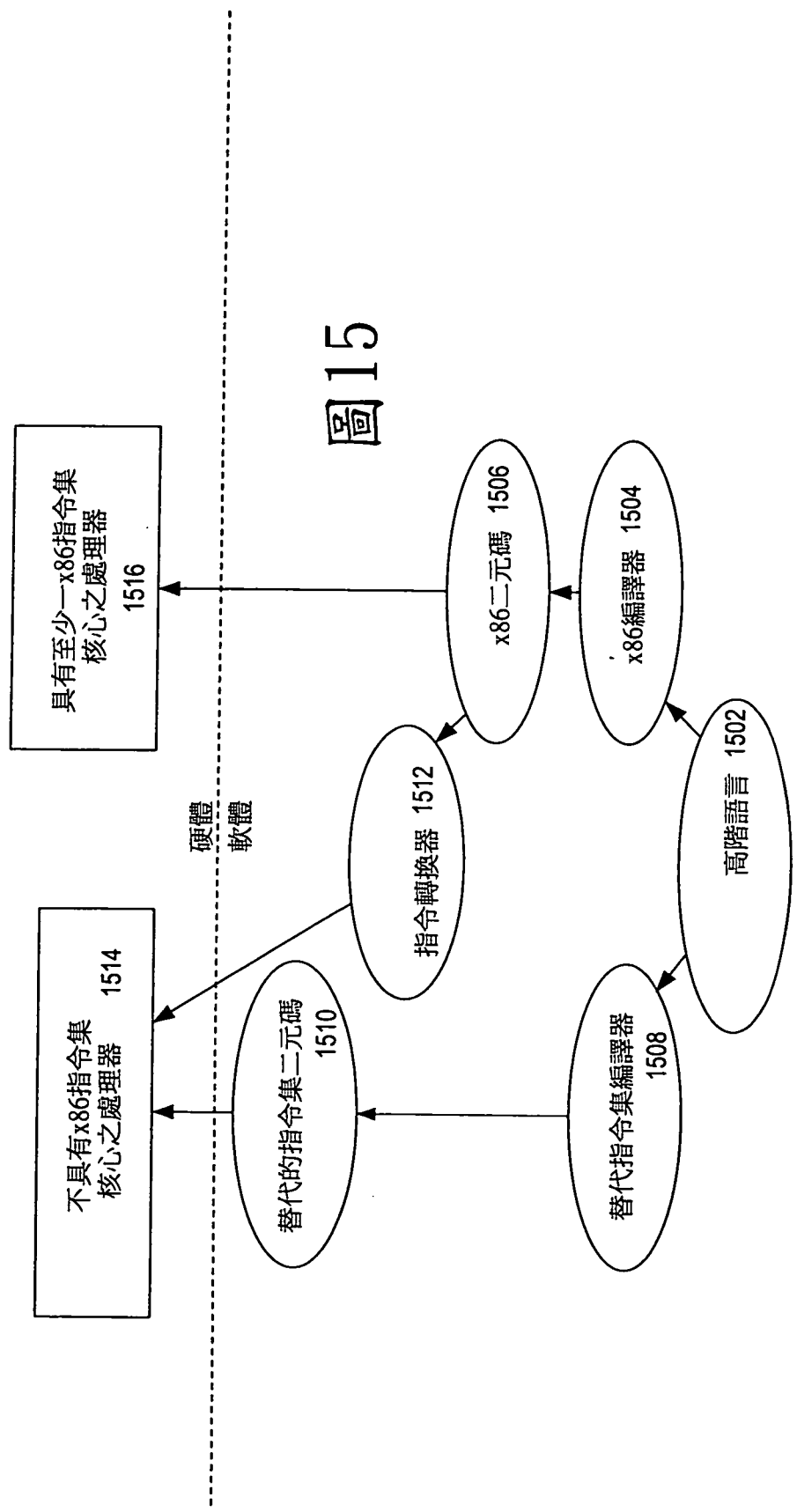


圖15