



(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 I489382 B

(45)公告日：中華民國 104 (2015) 年 06 月 21 日

(21)申請案號：101147489 (22)申請日：中華民國 101 (2012) 年 12 月 14 日

(51)Int. Cl. : G06F9/30 (2006.01) G06F9/305 (2006.01)

(30)優先權：2011/12/23 世界智慧財產權組織 PCT/US11/67182

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

美國

(72)發明人：歐德亞麥德維爾 艾蒙斯特阿法 OULD-AHMED-VALL, ELMOUSTAPHA (MR) ;

瓦倫泰 羅柏 VALENTINE, ROBERT (US) ; 柯柏 吉瑟斯 CORBAL, JESUS

(ES) ; 托爾 布萊特 TOLL, BRET (US) ; 查尼 馬克 CHARNEY, MARK J. (US)

(74)代理人：林志剛

(56)參考文獻：

TW 200811705A

US 04881168A

US 05832288A

US 2002/0184480A1

US 2009/0187739A1

審查人員：李國隆

申請專利範圍項數：20 項 圖式數：16 共 120 頁

(54)名稱

改良的萃取指令背景之設備及方法

APPARATUS AND METHOD OF IMPROVED EXTRACT INSTRUCTIONS

(57)摘要

描述一種設備，其包括用以執行第一、第二、第三及第四指令之指令執行邏輯。該第一指令和該第二指令兩者從個別第一和第二輸入向量的多個第一無重疊區段之一選擇第一族群的輸入向量元件。該第一族群具有第一位元寬度。該些多個第一無重疊區段之每一者具有如該第一族群之相同位元寬度。該第三指令和該第四指令兩者從個別第三和第四輸入向量的多個第二無重疊區段之一選擇第二族群的輸入向量元件。該第二族群具有大於該第一位元寬度之第二位元寬度。該些多個第二無重疊區段之每一者具有如該第二族群之相同位元寬度。該設備包括遮蔽層電路，用以遮蔽該第一和第三指令之該第一和第二族群於第一粒度，其中，如此產生之個別所得為該第一和第三指令之個別所得。遮蔽電路亦遮蔽該第二和第四指令之該第一和第二族群於第二粒度，其中，如此產生之個別所得為該第二和第四指令之個別所得。

An apparatus is described that includes instruction execution logic circuitry to execute first, second, third and fourth instructions. Both the first instruction and the second instruction select a first group of input vector elements from one of multiple first non overlapping sections of respective first and second input vectors. The first group has a first bit width. Each of the multiple first non overlapping sections have a same bit width as the first group. Both the third instruction and the fourth instruction select a second group of input vector elements from one of multiple second non overlapping sections of respective third and fourth input vectors. The second group has a second bit width that is larger than the first bit width. Each of the multiple second non overlapping sections have a same bit width as the second group. The apparatus includes masking layer circuitry to mask the first and second groups of the first and third instructions at a first

granularity, where, respective resultants produced therewith are respective resultants of the first and third instructions. The masking circuitry is also to mask the first and second groups of the second and fourth instructions at a second granularity, where, respective resultants produced therewith are respective resultants of the second and fourth instructions.

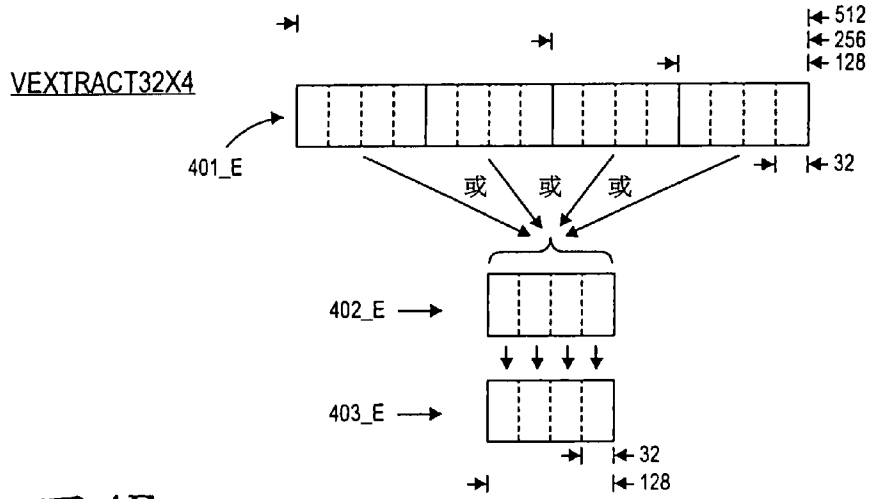


圖 4E

# 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：101147489

※申請日：101年12月14日

※IPC分類：

G06F 9/30 (2006.01)

9/30 (2006.01)

一、發明名稱：(中文/英文)

改良的萃取指令背景之設備及方法

Apparatus and method of improved extract instructions

二、中文發明摘要：

描述一種設備，其包括用以執行第一、第二、第三及第四指令之指令執行邏輯。該第一指令和該第二指令兩者從個別第一和第二輸入向量的多個第一無重疊區段之一選擇第一族群的輸入向量元件。該第一族群具有第一位元寬度。該些多個第一無重疊區段之每一者具有如該第一族群之相同位元寬度。該第三指令和該第四指令兩者從個別第三和第四輸入向量的多個第二無重疊區段之一選擇第二族群的輸入向量元件。該第二族群具有大於該第一位元寬度之第二位元寬度。該些多個第二無重疊區段之每一者具有如該第二族群之相同位元寬度。該設備包括遮蔽層電路，用以遮蔽該第一和第三指令之該第一和第二族群於第一粒度，其中，如此產生之個別所得為該第一和第三指令之個別所得。遮蔽電路亦遮蔽該第二和第四指令之該第一和第二族群於第二粒度，其中，如此產生之個別所得為該第二和第四指令之個別所得。

### 三、英文發明摘要：

An apparatus is described that includes instruction execution logic circuitry to execute first, second, third and fourth instructions. Both the first instruction and the second instruction select a first group of input vector elements from one of multiple first non overlapping sections of respective first and second input vectors. The first group has a first bit width. Each of the multiple first non overlapping sections have a same bit width as the first group. Both the third instruction and the fourth instruction select a second group of input vector elements from one of multiple second non overlapping sections of respective third and fourth input vectors. The second group has a second bit width that is larger than the first bit width. Each of the multiple second non overlapping sections have a same bit width as the second group. The apparatus includes masking layer circuitry to mask the first and second groups of the first and third instructions at a first granularity, where, respective resultants produced therewith are respective resultants of the first and third instructions. The masking circuitry is also to mask the first and second groups of the second and fourth instructions at a second granularity, where, respective resultants produced therewith are respective resultants of the second and fourth instructions.

四、指定代表圖：

(一) 本案指定代表圖為：第(4E)圖。

(二) 本代表圖之元件符號簡單說明：無

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】

本發明一般係有關計算科學之領域；及，更明確地，有關改良的萃取指令背景之設備及方法。

### 【先前技術】

圖 1 顯示於半導體晶片上實施有邏輯電路之處理核心 100 的高階圖。該處理核心包括一管線 101。管線係由多個級所組成，該些級各設計成在欲完整地執行程式碼指令所需的多步驟程序中履行一特定步驟。這些通常包括至少：1) 指令提取及解碼；2) 資料提取；3) 執行；4) 寫回。執行級履行由一被提取並解碼於先前級（例如，於上述步驟 1) ) 的指令所識別的特定操作在其由相同指令所識別並被提取於另一先前級（例如，上述步驟 2) ) 的資料上。所被操作之資料通常被提取自（通用）暫存器儲存空間 102。於該操作之完成時所產生的新資料通常亦被「寫回」至暫存器儲存空間（例如，於上述級 4) ) 。

與執行級相關的邏輯電路通常係由多個「執行單元」或「功能單元」103\_1 至 103\_N 所構成，該些「執行單元」或「功能單元」103\_1 至 103\_N 各被設計成履行其本身獨特的操作子集（例如，第一功能單元履行整數數學操作、第二功能單元履行浮點指令、第三功能單元履行載入/儲存操作自/至快取/記憶體，等等）。由所有功能單元所履行之所有操作的集合係相應於由處理核心 100 所支援之

「指令集」。

以下兩種類型的處理器架構被廣泛地認識於電腦科學之領域：「純量」及「向量」。純量處理器被設計成執行其在單一資料集上履行操作之指令，而向量處理器被設計成執行其在多個資料集上履行操作之指令。圖 2A 及 2B 提出一比較範例，其展示介於純量處理器與向量處理器之間的基本差異。

圖 2A 顯示一純量 AND 指令之範例，其中單一運算元集（A 與 B）被 AND 一起以產生獨特（或「純量」）結果 C（亦即， $AB = C$ ）。反之，圖 2B 顯示一向量 AND 指令之範例，其中兩運算元集（A/B 與 D/E）被個別地平行 AND 一起以同時地產生向量結果 C、F（亦即， $A.AND.B = C$  及  $D.AND.E = F$ ）。就術語而言，「向量」為具有多個「元件」之資料元件。例如，向量  $V = Q、R、S、T、U$  具有五個不同元件：Q、R、S、T 及 U。範例向量 V 之「大小」為五（因為其具有五個元件）。

圖 1 亦顯示不同於通用暫存器空間 102 之向量暫存器空間 104 的存在。明確地，通用暫存器空間 102 被額定地用以儲存純量值。如此一來，當任一執行單元履行純量操作時，其係額定地使用從（並將結果寫回至）通用暫存器空間 102 所呼叫的運算元。反之，當任一執行單元履行向量操作時，其係額定地使用從（並將結果寫回至）向量暫存器空間 107 所呼叫的運算元。記憶體之不同區可類似地配置給純量值及向量值之儲存。



亦注意於功能單元 103\_1 至 103\_N 之個別輸入及輸出上的遮蔽邏輯 104\_1 至 104\_N 及 105\_1 至 105\_N 的存在。於各個實施方式中，僅有這些層之一者被實際地實施 - 雖然其並非嚴格的條件。針對任何利用遮蔽之指令，輸入遮蔽邏輯 104\_1 至 104\_N 及 / 或輸出遮蔽邏輯 105\_1 至 105\_N 可被用以控制哪些元件被有效率地操作於該向量指令。於此，一遮罩向量被讀取自一遮罩暫存器空間 106 ( 例如，連同讀取自向量暫存器儲存空間 107 之輸入資料向量 ) 且被提出給遮罩邏輯 104、105 層之至少一者。

於執行向量程式碼之過程，各向量指令無需要求完整的資料字元。例如，某些指令之輸入向量可僅為 8 元件，其他指令之輸入向量可為 16 元件，其他指令之輸入向量可為 32 元件，等等。遮蔽層 104/105 因此被用以識別一請求特定指令之完整向量資料字元的一組元件，以使涵蓋指令之不同向量大小生效。通常，針對各向量指令，遮罩暫存器空間 106 中所保持的特定遮罩型態係由指令呼叫出來、從遮罩暫存器空間提取並提供至遮蔽層 104/105 之一者或兩者以「致能」該特定向量操作之正確組元件。

圖 3A 至 3E 顯示習知技術 VINSERT、VEXTRACT 及 VPERMUTE 指令之邏輯操作。注意：相較於其確實名稱該些指令之名稱已被縮寫或簡化。

圖 3A 顯示習知技術 VINSERT 指令之邏輯操作。如圖 3A 中所見，第一輸入運算元相應於 128 位元之資訊

301\_A 而第二輸入運算元相應於 256 位元向量 302\_A。第三、即刻輸入運算元（未顯示）係指明 256 位元向量 302\_A 之哪一半（下半或右半）將被取代以第一輸入運算元之 128 位元資訊 301\_A。所得的結構被儲存於具有 256 位元之大小的目的地/結果向量。128 位元之資訊 301、輸入向量 302\_A 及結果為浮點值，其可為 32 位元或 64 位元之大小。

圖 3B 顯示習知技術 VEXTRACT 指令之邏輯。如圖 3B 中所見，第一輸入運算元相應於 256 位元之向量 301\_B。第二、即刻輸入運算元（未顯示）係指明 256 位元向量 301\_B 之哪一半（下半或右半）將被覆寫於目的地暫存器 302\_B 中所儲存之 256 位元向量的最低階 128 位元上。輸入向量 301\_B 向量被構成為浮點值，其可為 32 位元或 64 位元之大小。指令格式可替代地指明記憶體中之 128 位元為目的地而非目的地暫存器 302\_B。

圖 3C 至 3E 顯示三個不同 VPERMUTE 指令（VPERMILPS、VPERMILPD、VPERM2F128）之個別邏輯操作。

圖 3C 顯示 VPERMILPS 指令之邏輯操作。如圖 3C 中所見，VPERMILPS 指令接受一相應於具有八個 32 位元（單一精確）浮點值之 256 位元輸入向量的輸入運算元 301\_C。其結果亦為具有八個 32 位元單一精確浮點值為其元件 302\_C 的 256 位元向量。第二輸入向量（未顯示）係針對該結果之下半中的四個元件之每一個獨特地指明輸入

向量 301\_C 之下半中的四個元件 301\_C\_1 至 301\_C\_4 之哪個將以其內容提供輸出元件。

圖 3C 顯示僅針對輸出元件 302\_C\_1 及 302\_C\_5 之操作。於此，輸出元件 302\_C\_1 之內容可被「填充」以輸入元件 301\_C\_1 至 301\_C\_4 之任一者的內容。於第二輸入向量（未顯示）中明確表達輸入元件 301\_C\_1 至 301\_C\_4 之哪一者被選來填充輸出元件 302\_C\_1。於此，第二輸入向量含有針對輸出向量中之八個元件的每一者之分離的 2 位元控制欄位。結果 302\_C 之下半中的輸出元件之來源需被選自輸入向量 301\_C 之下半。同樣地，結果 302\_C 之上半中的輸出元件之來源需被選自輸入向量 301\_C 之上半。

雖未明確地顯示於圖 3C 中，輸出元件 302\_C\_2 至 302\_C\_4 之每一者的內容係藉由第二輸入向量中所含有之資訊而被獨特地指明為輸入元件 301\_C\_1 至 301\_C\_4 之任一者。類似地，如圖 3C 中所見，輸出元件 302\_C\_5 之內容可被「填充」以輸入元件 301\_C\_5 至 301\_C\_8 之任一者的內容。再次，於第二輸入向量（未顯示）中亦明確表達輸入元件 301\_C\_5 至 301\_C\_8 之哪一者被選來填充輸出元件 302\_C\_5。輸出元件 302\_C\_6 至 302\_C\_8 之每一者的內容係藉由（未顯示）第二輸入向量而被獨特地指明為輸入元件 301\_C\_5 至 301\_C\_8 之任一者。

VPERMILPS 指令之另一版本係使用即刻運算元以取代第二輸入向量來選擇輸入向量 301\_C 之選擇型態。於

此，目的地之下半的輸入元件選擇型態係匹配目的地之上半的輸入元件選擇型態。

圖 3D 顯示 VPERMILPD 指令之邏輯操作。如圖 3D 中所見，VPERMILPD 指令接受一相應於具有四個 64 位元（雙精確）浮點值之 256 位元輸入向量的輸入運算元 301\_D。其結果亦為具有四個 64 位元雙精確浮點值為其元件的 256 位元向量 302\_D。第二輸入向量（未顯示）係針對該結果之下半中的兩個元件之每一個獨特地指明輸入向量 301\_D 之下半中的兩個元件 301\_D\_1 至 301\_D\_2 之哪個將以其內容提供輸出元件。

如圖 3D 中所見，輸出元件 302\_D\_1 及 302\_D\_2 之每一者可被獨特地「填充」以輸入元件 301\_D\_1 或 301\_D\_2 之任一者。同樣地，輸出元件 302\_D\_3 及 302\_D\_4 之每一者可被獨特地「填充」以輸入元件 301\_D\_3 或 301\_D\_4 之任一者。於第二輸入向量（未顯示）中明確表達哪一輸入元件被選來填充一特定的輸出元件。於此，第二輸入向量含有針對輸出向量中之四個元件的每一者之分離的 2 位元控制欄位。

VPERMILPD 指令之另一版本係使用即刻運算元以取代第二輸入向量來選擇輸入向量 301\_D 之選擇型態。於此，目的地之下半的輸入元件選擇型態係匹配目的地之上半的輸入元件選擇型態。

針對 VPERMILPS 及 VPERMIPLD 指令兩者，其結果被儲存於以指令之指令格式所指明的向量暫存器中。第一

輸入向量之來源被指明以指令格式並相應於當第二輸入向量被用來判定選擇型態時的向量暫存器。於此情況下，第二輸入向量之來源亦被指明以指令格式並相應於第二向量暫存器或記憶體位置之任一者。反之，假如即刻運算元被用以判定選擇型態，則第一輸入向量之來源被指明以指令格式並可為一向量暫存器或一記憶體位置。

圖 3E 顯示 VPERM2F128 指令之邏輯操作。如圖 3E 中所見，VPERM2F128 指令接受兩分離的 256 位元向量輸入運算元 301\_E、302\_E。256 位元結果 303\_E 之下與上 128 位元半 303\_E\_1、303\_E\_2 兩者可被填充以輸入向量 301\_E、302\_E 兩者之下或上半 301\_E\_1、301\_E\_2、302\_E\_1、302\_E\_2 的任一者。其結果被儲存於以指令之指令格式所指明之向量暫存器中。輸入向量 301\_E、302\_E 兩者之來源被指明以指令格式並可相應於一對向量暫存器或者一向量暫存器及一記憶體位置。

### 【發明內容及實施方式】

#### 詳細描述

圖 4A 至 4E 顯示四個新的 VINSERT 指令之邏輯操作。明確地，圖 4A 至 4D 個別地顯示 VINSERTF32X4 指令、VINSERTF64X2 指令、VINSERTF32X8 指令及 VINSERTF64X4 指令。

圖 4A 顯示 VINSERTF32X4 指令之邏輯操作。如圖 4A 中所見，128 位元結構 401\_A 被接收為第一輸入運算

元。該 128 位元結構含有四個 32 位元單一精確浮點值。512 位元向量亦被接收為第二輸入運算元 402\_A。512 位元向量可被視為由四個鄰接「塊」的資料所組成，其每個為 128 位元之大小。一即刻運算元（未顯示）指示第二運算元 402\_A 之哪個 128 位元「塊」將被覆寫以第一輸入運算元 401\_A。確認該第二運算元被覆寫為真的意義在於其被讀取自向量暫存器空間之資訊的執行單元中之局部副本被覆寫。如本技術中眾所周知者，向量暫存器空間中之原始來源資訊通常僅被覆寫於假如指令格式指示其來源暫存器亦為目的地暫存器時。

接著將一遮罩施加至所得的資料結構 403\_A。於此，一遮蔽層 404\_A 接收一輸入遮罩向量（未顯示），其識別所得資料結構 403\_A 之哪些 32 位元資料值將被寫入至目的地 406\_A。遮罩暫存器空間（例如，諸如圖 1 之遮罩暫存器空間 106）中之遮罩向量的位置以及向量暫存器空間中之目的地的位置兩者被識別於指令格式中。第一和第二輸入運算元 401\_A、402\_A 之來源亦被識別於指令格式中。於一實施例中，第一輸入運算元 401\_A 可源自向量暫存器空間或記憶體位置，而第二輸入運算元 402\_A 係源自向量暫存器空間。即刻運算元亦包括於指令格式中。於一實施例中，其中儲存有目的地之向量暫存器空間係不同於其中輸入運算元所源自之向量暫存器空間。

於一實施例中，取代從遮罩暫存器空間讀取遮罩，遮罩係嵌入於指令本身中（例如，類似於即刻運算元）。於

此情況下，指令格式包括一含有實際遮罩型態之欄位。反之，假如遮罩被提取自遮罩暫存器空間，則指令格式包括一識別其遮罩型態被儲存於遮罩暫存器空間中之何處的位址欄位。爲了簡化，以下剩餘說明書之討論被寫成遮罩係提取自遮罩暫存器空間。然而，讀者應瞭解其指令亦可被實施以遮罩嵌入指令中而成爲一種即刻運算元。

於一種向量友善指令格式之實施方式中，以下詳細地描述其實施例，指令格式係支援這些遮罩型態技術之兩者。於此情況下，指令格式包括一識別針對指令將採取哪種方式之額外欄位（例如，1 = 遮罩型態類似於嵌入指令中之即刻運算元，0 = 遮罩型態將被提取自遮罩暫存器空間）。

分離地或結合地，所施加之遮蔽的類型可被「合併」或「歸零」。於合併遮蔽之情況下，所得之資料結構的「遮蔽掉」欄位不被覆寫。反之，在該位置上之目的地暫存器中的原始值被保存。相對地，於歸零遮蔽之情況下，所得之資料結構的「遮蔽掉」欄位以 0 之值覆寫目的地暫存器中之該位置。於各個實施例中（諸如剛剛如上所述之關聯與向量友善指令格式之那些），應施加合併或歸零被指明於指令格式之另一欄位中。

關於遮蔽之上述評論亦適用於關聯與圖 4B 至 4P 之指令的下列討論。爲了便利起見，其不被重複於下。

圖 4B 顯示 VINSERTF64X2 指令之邏輯操作。如圖 4B 中所見，128 位元結構 401\_B 被接收爲第一輸入運算

元。該 128 位元結構含有兩個 64 位元雙精確浮點值。512 位元向量亦被接收為第二輸入運算元 402\_B。512 位元向量可被視為由四個鄰接「塊」的資料所組成，其每個為 128 位元之大小。一即刻運算元（未顯示）指示第二運算元 402\_B 之哪個 128 位元「塊」將被覆寫以第一輸入運算元 401\_B。

接著將一遮罩施加至所得的資料結構 403\_B。於此，一遮蔽層 404\_B 接收一輸入遮罩向量（未顯示），其識別所得資料結構 403\_B 之哪些 64 位元資料值將被寫入至目的地 406\_B。遮罩暫存器空間中之遮罩向量的位置以及向量暫存器空間中之目的地的位置兩者被識別於指令格式中。第一和第二輸入運算元 401\_B、402\_B 之來源亦被識別於指令格式中。於一實施例中，第一輸入運算元 401\_B 可源自向量暫存器空間或記憶體位置，而第二輸入運算元 402\_B 係源自向量暫存器空間。即刻運算元亦包括於指令格式中。於一實施例中，其中儲存有目的地之向量暫存器空間係不同於其中輸入運算元所源自之向量暫存器空間。

圖 4C 顯示 VINSERTF32X8 指令之邏輯操作。如圖 4C 中所見，256 位元結構 401\_C 被接收為第一輸入運算元。該 256 位元結構含有八個 32 位元單精確浮點值。512 位元向量亦被接收為第二輸入運算元 402\_C。512 位元向量可被視為由兩個鄰接「塊」的資料所組成，其每個為 256 位元之大小。一即刻運算元（未顯示）指示第二運算元 402\_C 之哪個 256 位元「塊」將被覆寫以第一輸入運算



元 401\_C。

接著將一遮罩施加至所得的資料結構 403\_C。於此，一遮蔽層 404\_C 接收一輸入遮罩向量（未顯示），其識別所得資料結構 403\_C 之哪些 32 位元資料值將被寫入至目的地 406\_C。遮罩暫存器空間中之遮罩向量的位置以及向量暫存器空間中之目的地的位置兩者被識別於指令格式中。第一和第二輸入運算元 401\_C、402\_C 之來源亦被識別於指令格式中。於一實施例中，第一輸入運算元 401\_C 可源自向量暫存器空間或記憶體位置，而第二輸入運算元 402\_C 係源自向量暫存器空間。即刻運算元亦包括於指令格式中。於一實施例中，其中儲存有目的地之向量暫存器空間係不同於其中輸入運算元所源自之向量暫存器空間。

圖 4D 顯示 VINSERTF64X4 指令之邏輯操作。如圖 4D 中所見，256 位元結構 401\_D 被接收為第一輸入運算元。該 256 位元結構含有四個 64 位元雙精確浮點值。512 位元向量亦被接收為第二輸入運算元 402\_D。512 位元向量可被視為由四個鄰接「塊」的資料所組成，其每個為 256 位元之大小。一即刻運算元（未顯示）指示第二運算元 402\_D 之哪個 256 位元「塊」將被覆寫以第一輸入運算元 401\_D。

接著將一遮罩施加至所得的資料結構 403\_D。於此，一遮蔽層 404\_D 接收一輸入遮罩向量（未顯示），其識別所得資料結構 403\_D 之哪些 64 位元資料值將被寫入至目的地 406\_D。遮罩暫存器空間中之遮罩向量的位置以及

向量暫存器空間中之目的地的位置兩者被識別於指令格式中。第一和第二輸入運算元 401\_D、402\_D 之來源亦被識別於指令格式中。於一實施例中，第一輸入運算元 401\_D 可源自向量暫存器空間或記憶體位置，而第二輸入運算元 402\_D 係源自向量暫存器空間。即刻運算元亦包括於指令格式中。於一實施例中，其中儲存有目的地之向量暫存器空間係不同於其中輸入運算元所源自之向量暫存器空間。

圖 4E 至 4H 顯示四個新的 VEXTRACT 指令。明確地，圖 4E 顯示 VEXTRACT32X4 指令，圖 4F 顯示 VEXTRACT64X2 指令，圖 4G 顯示 VEXTRACT32X8 指令及圖 4H 顯示 VEXTRACT64X4 指令。

圖 4E 顯示 VEXTRACT32X4 指令之邏輯操作。如圖 4E 中所見，VEXTRACT32X4 指令接受 512 位元輸入運算元 401\_E。該 512 位元輸入運算元 401\_E 可被視為由四個鄰接之 128 位元「塊」的資料所組成，其中，每塊含有四個單一精確（32 位元）浮點資料值。依據 VEXTRACT32X4 指令之邏輯操作，這些 128 位元塊之一被「選擇」，一遮蔽層 402\_E 被施加至所選擇的塊，且所得的資料結構被寫入至目的地 403\_E。

於一實施例中，一即刻運算元（未顯示）指明四個 128 位元塊之哪個將被選擇，輸入運算元 401\_E 被提供自一向量暫存器，及目的地 401\_E 可於向量暫存器空間或記憶體位置之任一者中。指令格式識別任何可應用的來源/目的地向量暫存器及/或記憶體位置。於一實施例中，目

的地被儲存於其中輸入向量所源自之向量暫存器空間以外的不同向量暫存器空間中。於指令之操作期間，一遮罩向量被讀取自遮罩暫存器空間並施加至遮蔽層 402\_E。遮罩暫存器空間中之遮罩向量的位置亦被提供於指令格式中。遮蔽之粒度 (granularity) 為 32 位元。亦即，遮蔽向量分別地指明選定資料塊中之四個 32 位元值的哪個將被寫入至目的地。

圖 4F 顯示 VEXTRACT64X2 指令之邏輯操作。如圖 4F 中所見，VEXTRACT64X2 指令接受 512 位元輸入運算元 401\_F。該 512 位元輸入運算元 401\_F 可被視為由四個鄰接之 128 位元「塊」的資料所組成，其中，每塊含有兩個雙精確 (64 位元) 浮點資料值。依據 VEXTRACT64X2 指令之邏輯操作，這些 128 位元塊之一被「選擇」，一遮蔽層 402\_F 被施加至所選擇的塊，且所得的資料結構被寫入至目的地 403\_F。

於一實施例中，一即刻運算元 (未顯示) 指明四個 128 位元塊之哪個將被選擇，輸入運算元 401\_F 被提供自一向量暫存器，及目的地 403\_F 可於向量暫存器空間或記憶體位置之任一者中。指令格式識別任何可應用的來源/目的地向量暫存器及/或記憶體位置。於一實施例中，目的地被儲存於其中輸入向量所源自之向量暫存器空間以外的不同向量暫存器空間中。於指令之操作期間，一遮罩向量被讀取自遮罩暫存器空間並施加至遮蔽層 402\_F。遮罩暫存器空間中之遮罩向量的位置亦被提供於指令格式中。

遮蔽之粒度 (granularity) 為 64 位元。亦即，遮蔽向量分別地指明選定資料塊中之兩個 64 位元值的哪個將被寫入至目的地。

圖 4G 顯示 VEXTRACT32X8 指令之邏輯操作。如圖 4G 中所見，VEXTRACT32X8 指令接受 512 位元輸入運算元 401\_G。該 512 位元輸入運算元 401\_G 可被視為由兩個鄰接之 256 位元「塊」的資料所組成，其中，每塊含有八個單一精確 (32 位元) 浮點資料值。依據 VEXTRACT32X8 指令之邏輯操作，這些 256 位元塊之一被「選擇」，一遮蔽層 402\_G 被施加至所選擇的塊，且所得的資料結構被寫入至目的地 403\_G。

於一實施例中，一即刻運算元 (未顯示) 指明兩 256 位元塊之哪個將被選擇，輸入運算元 401\_G 被提供自一向量暫存器，及目的地 403\_G 可於向量暫存器空間或記憶體位置之任一者中。指令格式識別任何可應用的來源/目的地向量暫存器及/或記憶體位置。於一實施例中，目的地被儲存於其中輸入向量所源自之向量暫存器空間以外的不同向量暫存器空間中。於指令之操作期間，一遮罩向量被讀取自遮罩暫存器空間並施加至遮蔽層 402\_G。遮罩暫存器空間中之遮罩向量的位置亦被提供於指令格式中。遮蔽之粒度 (granularity) 為 32 位元。亦即，遮蔽向量分別地指明選定資料塊中之八個 32 位元值的哪個將被寫入至目的地。

圖 4H 顯示 VEXTRACT64X4 指令之邏輯操作。如圖

4H 中所見，VEXTRACT64X4 指令接受 512 位元輸入運算元 401\_H。該 512 位元輸入運算元 401\_H 可被視為由兩個鄰接之 256 位元「塊」的資料所組成，其中，每塊含有四個雙精確（64 位元）浮點資料值。依據 VEXTRACT64X4 指令之邏輯操作，這些 256 位元塊之一被「選擇」，一遮蔽層 402\_H 被施加至所選擇的塊，且所得的資料結構被寫入至目的地 403\_H。

於一實施例中，一即刻運算元（未顯示）指明兩個 256 位元塊之哪個將被選擇，輸入運算元 401\_H 被提供自一向量暫存器，及目的地 403\_H 可於向量暫存器空間或記憶體位置之任一者中。指令格式識別任何可應用的來源/目的地向量暫存器及/或記憶體位置。於一實施例中，目的地被儲存於其中輸入向量所源自之向量暫存器空間以外的不同向量暫存器空間中。於指令之操作期間，一遮罩向量被讀取自遮罩暫存器空間並施加至遮蔽層 402\_H。遮罩暫存器空間中之遮罩向量的位置亦被提供於指令格式中。遮蔽之粒度（granularity）為 64 位元。亦即，遮蔽向量分別地指明選定資料塊中之四個 64 位元值的哪個將被寫入至目的地。

圖 4I 至 4P 提出新的 VPERMUTE 指令。這些包括 VPERMW、VPERMD、VPERMQ、VPERMILPS、VPERMILPD。

圖 4I 顯示 VPERMW 指令之邏輯操作。VPERMW 指令接受 512 位元輸入向量為第一輸入運算元 401\_I。該 512

位元輸入向量被視為具有三十二個 16 位元資料值（字元）。一遮蔽層 402\_I 類似地具有粒度以遮蔽於 16 位元粒度。亦即，目的地/所得向量 403\_I 被視為具有三十二個 16 位元值，且遮蔽層 402\_I 具有提供個別遮蔽給所得向量 403\_I 中之各元件的能力。

依據 VPERMW 指令之邏輯操作，所得向量 403\_I 中之各元件被填充以輸入向量 401\_I 中之三十二個元件的任一個。此能力被顯示於所得向量 403\_I 之最右邊元件 403\_I\_1。於此，如圖 4I 中所見，遮蔽層元件 402\_I\_1 可被提供以輸入向量 401\_I 中之三十二個元件的任一個。如此一來，假如所得元件 403\_I\_1 未被「遮蔽掉」於遮蔽元件 402\_I\_1，則所得元件 403\_I\_1 可被「填充」以輸入向量 401\_I 中之三十二個 16 位元元件的任一個。

雖未明確地顯示於圖 4I 中，此相同能力/功能被應用於所得向量 403\_I 中之剩餘三十一個元件的每一個。亦即，所得向量 403\_I 中之任何元件（假如未被用於該所得中之該特定元件的遮蔽層 402\_I 中之專屬遮蔽元件所遮蔽掉的話）可被填充以輸入向量 401\_I 中之 16 位元值的任一者。此表示（例如）所得向量 403\_I 中之二或更多元件可被填充以來自輸入向量 401\_I 之相同元件。

輸入向量 401\_I 之來源及所得向量 403\_I 之目的地被指明於 VPERMW 指令格式。於一實施例中，輸入向量 401\_I 被提供自向量暫存器空間或記憶體中之一位置，且所得被寫入不同向量暫存器空間中之一位置。遮蔽層

402\_I 亦被提供以一來自遮罩暫存器空間之遮蔽向量，該遮罩暫存器空間係指明所得向量 403\_I 中之哪些元件將被遮蔽掉（以及所得向量 403\_I 中之哪些元件將不被遮蔽掉）。

亦提供第二輸入向量（未顯示），稱之為「指標」向量，其係針對輸出向量中之各元件明確表達哪個特定輸入元件將被用以填充輸出向量中之該元件（假如其未被遮蔽掉的話）。例如，於一實施例中，指標向量為具有至少五個位元於各元件之 32 元件向量。指標向量中之各元件係相應於所得中之一獨特元件。五個位元被用以指明第一輸入向量 401\_I 中之哪個元件將被用以填充特定的所得元件（假如其未被遮蔽掉的話）。

於一實施例中，亦支援 VPERMW 指令之第二版本，其接收一含有可用來填充任何輸出向量元件位置之三十二個額外 16 位元值的第三輸入向量（未顯示）。基本上，第三輸入向量將其可被用以填充任何輸出元件位置之可用 16 位元輸入值的數目乘以二。針對 VPERMW 指令之此版本，上述之指標向量針對 5 位元以外之各輸出位置使用至少 6 位元。於此，額外位元被用以識別第一（401\_I）或第三（未顯示）輸入向量將被用以源取特定的輸出向量 403\_I 元件。此版本之第一變異係使目的地暫存器空間相等於其供應指標向量之暫存器空間。亦即，其結果被覆寫向量暫存器空間中之指標向量。此版本之第二變異係以該結果覆寫向量暫存器空間中之第三輸入向量。於任一變異

中，目的地位置未被分離地識別於指標向量位置（第一變異）或第三輸入向量（第二變異）以外的指令格式。

圖 4J 顯示 VPERMD 指令之邏輯操作。VPERMD 指令接受 512 位元輸入向量為第一輸入運算元 401\_J。該 512 位元輸入向量被視為具有十六個 32 位元資料值。32 位元值可為雙字元或單一精確 32 位元值。一遮蔽層 402\_J 類似地具有粒度以遮蔽於 32 位元粒度。亦即，目的地/所得向量 403\_J 被視為具有十六個 32 位元值，且遮蔽層 402\_J 具有提供個別遮蔽給所得向量 403\_J 中之各元件的能力。

依據 VPERMD 指令之邏輯操作，所得向量 403\_J 中之各元件被填充以輸入向量 401\_J 中之十六個元件的任一個。此能力被顯示於所得向量 403\_J 之最右邊元件 403\_J\_1。於此，如圖 4J 中所見，遮蔽層元件 402\_J\_1 可被提供以輸入向量 401\_J 中之十六個元件的任一個。如此一來，假如所得元件 403\_J\_1 未被「遮蔽掉」於遮蔽元件 402\_J\_1，則所得元件 403\_J\_1 可被「填充」以輸入向量 401\_J 中之十六個 32 位元元件的任一個。

雖未明確地顯示於圖 4J 中，此相同能力/功能被應用於所得向量 403\_J 中之剩餘十五個元件的每一個。亦即，所得向量 403\_J 中之任何元件（假如未被用於該所得中之該特定元件的遮蔽層 402\_J 中之專屬遮蔽元件所遮蔽掉的話）可被填充以輸入向量 401\_J 中之 32 位元值的任一者。此表示（例如）所得向量 403\_J 中之二或更多元件可被填充以來自輸入向量 401\_J 之相同元件。



輸入向量 401\_J 之來源及所得向量 403\_J 之目的地被指明於 VPERMD 指令格式。於一實施例中，輸入向量 401\_J 被提供自向量暫存器空間或記憶體中之一位置，且所得被寫入不同向量暫存器空間中之一位置。遮蔽層 402\_J 亦被提供以一來自遮罩暫存器空間之遮蔽向量，該遮罩暫存器空間係指明所得向量 403\_J 中之哪些元件將被遮蔽掉（以及所得向量 403\_J 中之哪些元件將不被遮蔽掉）。

亦提供第二輸入向量（未顯示），稱之為「指標」向量，其係針對輸出向量中之各元件明確表達哪個特定輸入元件將被用以填充輸出向量中之該元件（假如其未被遮蔽掉的話）。例如，於一實施例中，指標向量為具有至少五個位元於各元件之 16 元件向量。指標向量中之各元件係相應於所得中之一獨特元件。五個位元被用以指明第一輸入向量 401\_J 中之哪個元件將被用以填充特定的所得元件（假如其未被遮蔽掉的話）。

於一實施例中，亦支援 VPERMW 指令之第二版本，其接收一含有可用來填充任何輸出向量元件位置之十六個額外 32 位元值的第三輸入向量（未顯示）。基本上，第三輸入向量將其可被用以填充任何輸出元件位置之可用 32 位元輸入值的數目乘以二。針對 VPERMW 指令之此版本，上述之指標向量針對 5 位元以外之各輸出位置使用至少 6 位元。於此，額外位元被用以識別第一（401\_J）或第三（未顯示）輸入向量將被用以源取特定的輸出向量

403\_J 元件。此版本之第一變異係使目的地暫存器空間相等於其供應指標向量之暫存器空間。亦即，其結果被覆寫向量暫存器空間中之指標向量。此版本之第二變異係以該結果覆寫向量暫存器空間中之第三輸入向量。於任一這些變異中，目的地位置未被分離地識別於指標向量位置（第一變異）或第三輸入向量（第二變異）以外的指令格式。

圖 4K 顯示 VPERMQ 指令之邏輯操作。VPERMQ 指令接受第一 512 位元輸入向量為第一輸入運算元 401\_K 並接受第二 512 位元輸入向量為第二輸入運算元（未顯示）。該兩者 512 位元輸入向量均被視為具有八個 64 位元資料值。64 位元值可為四字元或雙精確浮點值。一遮蔽層 402\_K 類似地具有粒度以遮蔽於 64 位元粒度。亦即，目的地 / 所得向量 403\_K 被視為具有八個 64 位元值，且遮蔽層 402\_K 具有提供個別遮蔽給所得向量 403\_K 中之各元件的能力。

依據 VPERMQ 指令之邏輯操作，所得向量 403\_K 中之各元件被填充以輸入向量（401\_K 及該未顯示的輸入向量）之組合中之十六個元件的任一個。此能力被顯示於所得向量 403\_K 之最右邊元件 403\_K\_1。於此，如圖 4K 中所見，遮蔽層元件 402\_K\_1 可被提供以輸入向量 401\_K 中之八個元件的任一個、或者第二輸入向量（未顯示）中之八個元件的任一個。如此一來，假如所得元件 403\_K\_1 未被「遮蔽掉」於遮蔽元件 402\_K\_1，則所得元件 403\_K\_1 可被「填充」以來自該對輸入向量之十六個 64

位元元件的任一個。

雖未明確地顯示於圖 4K 中，此相同能力/功能被應用於所得向量 403\_K 中之剩餘八個元件的每一個。亦即，所得向量 403\_K 中之任何元件（假如未被用於該所得中之該特定元件的遮蔽層 402\_K 中之專屬遮蔽元件所遮蔽掉的話）可被填充以任一輸入向量中之 64 位元值的任一者。此表示（例如）所得向量 403\_K 中之二或更多元件可被填充以來自輸入向量 401\_K 之相同元件。

輸入向量兩者之來源被指明於 VPERMQ 指令格式。於一實施例中，輸入向量 401\_K 被提供自向量暫存器空間或記憶體中之一位置，且第二輸入向量被提供自向量暫存器空間中之其本身的個別位置。遮蔽層 402\_K 亦被提供以一來自遮罩暫存器空間之遮蔽向量，該遮罩暫存器空間係指明所得向量 403\_K 中之哪些元件將被遮蔽掉（以及所得向量 403\_K 中之哪些元件將不被遮蔽掉）。

亦提供另一輸入向量（未顯示），稱之為「指標」向量，其係針對輸出向量中之各元件明確表達哪個特定輸入元件將被用以填充輸出向量中之該元件（假如其未被遮蔽掉的話）。例如，於一實施例中，指標向量為具有至少六個位元於各元件之八元件向量。指標向量中之各元件係相應於所得中之一獨特元件。上述六個位元之五個被用以指明第一輸入向量 401\_K 或第二輸入向量中之哪個元件將被用以填充特定的所得元件（假如其未被遮蔽掉的話）。第六個位元被用以識別第一（401\_K）或第二（未顯示）

輸入向量將被用以源取該特定輸出向量 403\_K 元件。

VPERMQ 之第一變異使目的地暫存器空間相等於其供應指標向量之暫存器空間。亦即，其結果被覆寫向量暫存器空間中之指標向量。第二變異係以該結果覆寫向量暫存器空間中之第二輸入向量。於任一變異中，目的地位置未被分離地識別於指標向量位置（第一變異）或第二輸入向量（第二變異）以外的指令格式。

圖 4L 顯示新的 VPERMILPS 指令。如圖 4L 中所見，新的 VPERMILPS 指令接受一相應於具有十六個 32 位元（單一精確）浮點值之 512 位元輸入向量 401\_L 的輸入運算元。其結果亦為具有十六個 32 位元單一精確浮點值為其元件 403\_L 的 512 位元向量。一遮蔽層 402\_L 類似地具有粒度以遮蔽於 32 位元粒度。亦即，目的地/所得向量 403\_L 被視為具有十六個 32 位元值，且遮蔽層 402\_L 具有提供個別遮蔽給所得向量 403\_L 中之各元件的能力。

第二輸入向量（未顯示），稱之為「指標」向量，獨特地指明輸入向量 401\_L 中之四個元件的哪個可「填充」所得 403 中之一特定元件（假如其未被遮蔽層 402\_L 遮蔽掉的話）。例如，所得 403\_L 之最低有效 128 位元中之四個元件 403\_L\_1 至 403\_L\_4 的每個僅可被填充以輸入向量 401\_L 之最低有效 128 位元中之四個元件 401\_L\_1 至 403\_L\_4 的任一個。所得中之每個元件的來源係獨立於所得中之另一元件的來源。因此輸出向量中之兩不同元件可被填充以相同的輸入向量元件。

圖 4L 顯示其被提供至遮蔽層 402\_L 之輸出元件 403\_L\_1 的可用輸入源。再次，這些相同的輸入源亦可用於輸出元件 403\_L\_2 至 403\_L\_4 之每一者。一類似的配置係針對 128 位元之每一下個族群而被協調。亦即，假設無遮蔽，則來自族群 403\_L\_Q2 之元件需源自族群 401\_L\_Q2 中之元件，來自族群 403\_L\_Q3 之元件需源自族群 401\_L\_Q3 中之元件，以及來自族群 403\_L\_Q4 之元件需源自族群 401\_L\_Q4 中之元件，

於一實施例中，新 VPERMILPS 指令之不同版本被支援於相同處理器之指令集中。第一版本儲存輸入向量 401\_L 於第一向量暫存器位置中，儲存指標向量於第二向量暫存器空間位置或記憶體位置中並儲存結果於其本身的專屬向量暫存器空間位置中（亦即，輸入向量 401\_L 及指標向量均未被覆寫）。第二版本儲存輸入向量 401\_L 於第一向量暫存器空間位置或記憶體位置中，指明該指令中之指標向量為即刻運算元並儲存結果於其本身的專屬向量暫存器空間位置中（亦即，輸入向量 401\_L 未被覆寫）。

遮蔽層 402\_L 接收來自遮罩暫存器空間之遮罩向量。遮罩暫存器空間中之遮罩向量的位置被指明於該指令中。於其中指標向量被用以指明介於輸入向量元件與輸出向量元件之間的選擇型態之版本中，指標向量具有十六個元件（輸出向量中之各元件有一個元件），及兩個位元於各元件上，以選擇可用來填充個別輸出向量元件的四個輸入向量元件之一。於其中利用即刻運算元之版本中，即刻運算

元具有八個元件，其中每元件有兩位元。於此，相同的選擇型態被用於輸出向量之下半或上半。

圖 4M 顯示新的 VPERMILPD 指令。如圖 4M 中所見，新的 VPERMILPD 指令接受一相應於具有八個 64 位元（雙精確）浮點值之 512 位元輸入向量的輸入運算元 401\_M。其結果亦為具有八個 64 位元雙精確浮點值為其元件的 512 位元向量 403\_M。一遮蔽層 402\_M 類似地具有粒度以遮蔽於 64 位元粒度。亦即，目的地/所得向量 403\_M 被視為具有八個 64 位元值，且遮蔽層 402\_M 具有提供個別遮蔽給所得向量 403\_M 中之各元件的能力。

第二輸入向量（未顯示），稱之為「指標」向量，獨特地指明輸入向量 401\_M 中之兩個元件的哪個可「填充」所得 403\_M 中之一特定元件（假如其未被遮蔽層 402\_M 所遮蔽的話）。例如，所得 403\_M 之最低有效 128 位元中之元件 403\_M\_1 及 403\_M\_2 的兩者僅可被填充以輸入向量 401\_K 之最低有效 128 位元中之元件 401\_M\_1 與 403\_M\_2 的任一個。所得中之每個元件的來源係獨立於所得中之另一元件的來源。因此輸出向量中之兩不同元件可被填充以相同的輸入向量元件。

圖 4M 顯示其被提供至遮蔽層 402\_M 之輸出元件 403\_M\_1 及 403\_M\_2 的可用輸入源。一類似的配置係針對 128 位元之剩餘族群而被協調於輸出向量 403\_M 中。亦即，假設無遮蔽，則來自族群 403\_M\_Q2 之元件需源自族群 401\_M\_Q2 中之元件，來自族群 403\_M\_Q3 之元件需

源自族群 401\_M\_Q3 中之元件，以及來自族群 403\_M\_Q4 之元件需源自族群 401\_M\_Q4 中之元件。

於一實施例中，新 VPERMILPD 指令之不同版本被支援於相同處理器之指令集中。第一版本儲存輸入向量 401\_M 於第一向量暫存器空間位置中，儲存指標向量於第二向量暫存器空間位置或記憶體位置中並儲存結果於其本身的專屬向量暫存器空間位置中（亦即，輸入向量 401\_M 及指標向量均未被覆寫）。第二版本儲存輸入向量 401\_M 於第一向量暫存器空間位置或記憶體位置中，指明該指令中之指標向量為即刻運算元並儲存結果於其本身的專屬向量暫存器空間位置中（亦即，輸入向量 401\_M 未被覆寫）。

遮蔽層 402\_M 接收來自遮罩暫存器空間之遮罩向量。遮罩暫存器空間中之遮罩向量的位置被指明於該指令中。於其中指標向量被用以指明介於輸入向量元件與輸出向量元件之間的選擇型態之版本中，指標向量具有八個元件（輸出向量中之各元件有一個元件），及一位元於各元件上，以選擇可用來填充個別輸出向量元件的兩個輸入向量元件之一。於其中利用即刻運算元之版本中，即刻運算元具有八個元件，其中每元件有一位元。

圖 4N 顯示一 VPERM64X1 指令。如圖 4N 中所見，VPERM64X1 指令接受一相應於具有八個 64 位元（雙精確）浮點值之 512 位元輸入向量 401\_N 的輸入運算元。其結果亦為具有八個 64 位元雙精確浮點值為其元件的

512 位元向量 403\_N。一遮蔽層 402\_N 類似地具有粒度以遮蔽於 64 位元粒度。亦即，目的地 / 所得向量 403\_N 被視為具有八個 64 位元值，且遮蔽層 402\_N 具有提供個別遮蔽給所得向量 403\_N 中之各元件的能力。

第二輸入向量（未顯示），稱之為「指標」向量，獨特地指明輸入向量 401\_N 中之八個元件的哪個可「填充」所得 403\_N 中之一特定元件（假如其未被遮蔽層 402\_N 所遮蔽的話）。亦即，輸入向量 401\_N 中之任何元件可被用以填充輸出向量 403\_N 中之任何元件。所得中之每個元件的來源係獨立於所得中之另一元件的來源。因此輸出向量中之兩不同元件可被填充以相同的輸入向量元件。

圖 4N 顯示其被提供至遮蔽層 402\_N 之輸出元件 403\_N\_1 的可用輸入源。一類似的配置被協調於輸出向量 403\_N 中之各元件。如以下之更詳細描述，於一其中即刻運算元被使用為指標向量之實施例中，來源選擇被進一步限制。

於一實施例中，新 VPERM64X1 指令之不同版本被支援於相同處理器之指令集中。第一版本儲存輸入向量 401\_N 於第一向量暫存器空間位置或記憶體位置中，儲存指標向量於第二向量暫存器空間位置中並儲存結果於其本身的專屬向量暫存器空間位置中（亦即，輸入向量 401\_N 及指標向量均未被覆寫）。第二版本儲存輸入向量 401\_N 於第一向量暫存器空間位置或記憶體位置中，指明該指令



中之指標向量為即刻運算元並儲存結果於其本身的專屬向量暫存器空間位置中（亦即，輸入向量 401\_N 未被覆寫）。

遮蔽層 402\_N 接收來自遮罩暫存器空間之遮罩向量。遮罩暫存器空間中之遮罩向量的位置被指明於該指令中。於其中指標向量被用以指明介於輸入向量元件與輸出向量元件之間的選擇型態之版本中，指標向量具有八個元件（輸出向量中之各元件有一個元件），及三位元於各元件上，以選擇可用來填充個別輸出向量元件的八個輸入向量元件之一。於其中利用即刻運算元之版本中，即刻運算元具有四個元件，其中每元件有二位元。於此，各輸出元件並未具有其可得之所有八個輸入元件為來源。反之，輸出向量 403\_N 之最低有效 256 位元中的任何元件需由輸入向量 403\_N 之最低有效 256 位元中的元件所源取。輸入/輸出向量之最低有效 256 位元的來源之相同選擇型態被使用於輸入/輸出向量之最高有效位元。

圖 40 顯示一 VPERM64X2 指令之邏輯操作。依據 VPERM64X2 指令之操作，512 位元輸入向量 401\_O 被接收。512 位元輸入向量 401\_O 被視為具有八個 64 位元雙精確浮點或整數值。其結果亦為具有八個 64 位元雙精確浮點或整數值為其元件的 512 位元向量 403\_O。一遮蔽層 402\_O 類似地具有粒度以遮蔽於 64 位元粒度。亦即，目的地/所得向量 403\_O 被視為具有八個 64 位元值，且遮蔽層 402\_O 具有提供個別遮蔽給所得向量 403\_O 中之各元

件的能力。

第二輸入向量（未顯示），稱之為「指標」向量，獨特地指明輸入向量 401\_O 中之哪個相鄰對的元件可「填充」所得 403\_O 中之一特定對的相鄰元件（假如該對元件未被遮蔽層 402\_O 所遮蔽的話）。亦即，輸入向量 401\_O 中之四對相鄰元件的任一者可被用以填充輸出向量 403\_O 中之任何對元件。所得中之第一對元件的來源係獨立於所得中之另一對元件的來源。因此，輸出向量中之兩不同對元件可被填充以相同對的輸入向量元件。

圖 40 顯示其被提供至遮蔽層 402\_O 之輸出元件對 403\_O\_1 的可用輸入源。一類似的配置被協調於其他對元件 403\_O\_2、403\_O\_3 及 403\_O\_4。

於一實施例中，VPERM64X2 指令之不同版本被支援於相同處理器之指令集中。第一版本儲存輸入向量 401\_O 於第一向量暫存器空間位置或記憶體位置中，儲存指標向量於第二向量暫存器空間位置中並儲存結果於其本身的專屬向量暫存器空間位置中（亦即，輸入向量 401\_O 及指標向量均未被覆寫）。第二版本儲存輸入向量 401\_O 於第一向量暫存器空間位置或記憶體位置中，指明該指令中之指標向量為即刻運算元並儲存結果於其本身的專屬向量暫存器空間位置中（亦即，輸入向量 401\_O 未被覆寫）。

遮蔽層 402\_O 接收來自遮罩暫存器空間之遮罩向量。遮罩暫存器空間中之遮罩向量的位置被指明於該指令

中。於其中指標向量被用以指明介於輸入向量元件與輸出向量元件之間的選擇型態之版本中，指標向量具有四個元件（輸出向量中之各對元件有一個元件），及二位元於各元件上，以選擇可用來填充個別對輸出向量元件的四對輸入向量元件之一。於其中利用即刻運算元之版本中，即刻運算元具有如指標向量之相同結構。

圖 4P 顯示一 VPERM32X4 指令之邏輯操作。依據 VPERM32X4 指令之操作，512 位元輸入向量 401\_P 被接收。512 位元輸入向量 401\_P 被視為具有十六個 32 位元單一精確浮點或整數值。其結果亦為具有十六個 32 位元單一精確浮點或整數值為其元件的 512 位元向量 403\_P。一遮蔽層 402\_P 類似地具有粒度以遮蔽於 32 位元粒度。亦即，目的地/所得向量 403\_P 被視為具有八個 64 位元值，且遮蔽層 402\_P 具有提供個別遮蔽給所得向量 403\_P 中之各元件的能力。

第二輸入向量（未顯示），稱之為「指標」向量，獨特地指明輸入向量 401\_P 中之哪組四個 32 位元元件可「填充」所得 403\_P 中之特定組相鄰 32 位元元件（假如該組元件未被遮蔽層 402\_P 所遮蔽的話）。亦即，輸入向量 401\_P 中之四組四個相鄰元件的任一者可被用以填充輸出向量 403\_P 中之某些組四個元件。所得中之第一組元件的來源係獨立於所得中之另一組四個元件的來源。因此，輸出向量中之兩不同組的四個元件可被填充以相同組的四個輸入向量元件。

圖 4P 顯示其被提供至遮蔽層 402\_P 之輸出元件組 403\_P\_1 的可用輸入源。一類似的配置被協調於其他元件組 403\_P\_2、403\_P\_3 及 403\_P\_4。

於一實施例中，VPERM32X4 指令之不同版本被支援於相同處理器之指令集中。第一版本儲存輸入向量 401\_P 於第一向量暫存器空間位置或記憶體位置中，儲存指標向量於第二向量暫存器空間位置中並儲存結果於其本身的專屬向量暫存器空間位置中（亦即，輸入向量 401\_P 及指標向量均未被覆寫）。第二版本儲存輸入向量 401\_P 於第一向量暫存器空間位置或記憶體位置中，指明該指令中之指標向量為即刻運算元並儲存結果於其本身的專屬向量暫存器空間位置中（亦即，輸入向量 401\_P 未被覆寫）。

遮蔽層 402\_P 接收來自遮罩暫存器空間之遮罩向量。遮罩暫存器空間中之遮罩向量的位置被指明於該指令中。於其中指標向量被用以指明介於輸入向量元件與輸出向量元件之間的選擇型態之版本中，指標向量具有四個元件（輸出向量中之各組四個元件有一個元件），及二位元於各元件上，以選擇可用來填充個別組四個輸出向量元件的四組四個輸入向量元件之一。於其中利用即刻運算元之版本中，即刻運算元具有如指標向量之相同結構。

根據實施方式，一支援這些指令之管線的資料提取級可被耦合至記憶體以從記憶體提取運算元，及/或，被耦合至向量暫存器空間以提取/儲存一運算元/結果自/至向量暫存器空間。

此外，針對那些被描述為僅支援浮點資料值之指令，可理解地，這些相同指令可被擴充以處理整數及浮點資料值。於此，圖 1 之暫存器空間 102 可包括專用於儲存整數值之第一部分及專用於儲存浮點值之另一部分。無擴充至整數值，管線之資料提取及寫回級被耦合至浮點暫存器空間且無需被耦合至整數暫存器空間，針對其中這些指令被處理之方式。反之，資料提取及寫回級可被耦合至任一者（取決於指令之解碼），假如這些指令被設計為支援兩種資料型式的話。

圖 5A 顯示一種邏輯設計，用於可實施以上於圖 4A 至 4P 中所討論之（包括所有）指令的任何組合之執行。於其他實施例中，不同的個別執行單元被用以執行如以上討論之向量萃取、向量插入及向量排列指令。於又進一步可能的實施例中，單一執行單元可執行來自兩個這些指令家族之指令集。

參考圖 5A，第一暫存器 501 保存如上討論之第一輸入運算元。明顯地，第一暫存器 501 之大小應夠大以保存其預期能夠複製之最大資料結構。第一暫存器 501 可置於一管線中之資料提取級的後端上。替代地或結合地，第一暫存器 501 可被視為其中儲存有第一輸入運算元之向量暫存器空間 107 內的暫存器。替代地，第一暫存器可被置於執行單元之「前端」。第一暫存器 501 被耦合至向量元件路由邏輯電路 502。向量元件路由邏輯電路 502 被設計成將第一暫存器 501 中之輸入運算元的個別元件指向至符合

資訊及指標向量（或即刻運算元）之遮蔽層邏輯電路 504 的適當元件位置，且整體指令被執行。針對其被微編碼之那些實施方式，向量元件路由邏輯電路 502 之術語「邏輯電路」等將包括微碼及其回應於微碼而作用之邏輯。暫存器 507 保存指標向量（或即刻運算元）且被耦合至向量元件路由邏輯電路 502 以控制第一運算元之向量元件的「路由」。

向量元件路由邏輯電路 502 之輸出 503 被耦合至遮蔽層邏輯電路 504。於一實施例中，輸出 503 中之個別輸出的數目係相應於所得向量之最大可能大小（例如，512 位元）。遮蔽層邏輯電路 502 被建構以支援其由執行單元所支援之指令的特定向量元件粒度。例如，假如執行單元支援 16 位元、32 位元、及 64 位元所得元件粒度，則遮蔽層邏輯電路 502 被設計成支援這些粒度之每一者上的遮蔽。

如以上之詳細討論，遮蔽層邏輯電路 504 接收來自暫存器 505 之遮蔽型態以供應一遮罩至來自向量元件路由之輸出 503，以產生所得資料結構於暫存器 506 中。暫存器 505 可回應於遮罩暫存器空間中之一暫存器、管線中之資料提取級的輸出上之一暫存器、執行單元之前端上之一暫存器、或者一保存即刻值於指令中（當遮罩型態被嵌入指令中時）之暫存器（例如，於管線之指令解碼級的輸出上）。

根據實施方式，暫存器 506 可回應於向量暫存器空間

107 內之一暫存器，或者暫存器 506 可為位於執行單元之輸出上的暫存器。

輸入暫存器 510 儲存用於如上討論之接受第二輸入向量的那些指令之一額外輸入運算元，該第二輸入向量之元件可被選擇以包括入所得（例如，VPERMW、VPERMD 及 VPERMQ）。輸入暫存器 510（或其他暫存器）亦可被用以儲存如上討論之 VINSERT 指令的第二運算元。如此一來，該暫存器被顯示為饋送目的地 506。暫存器 510 可位於向量暫存器空間中，於指令執行管線之資料提取級的輸出上，或者可位於執行單元之輸入上。

於進一步實施例中，執行單元邏輯電路被設計為不僅支援圖 4A 至 4P 之任何/所有指令，同時亦支援圖 3A 至 3E 之任何/所有習知技術指令。於此情況下，因為圖 3A 至 3E 之習知技術指令不支援遮蔽，所以存在一繞過遮蔽層邏輯電路 504 之旁通路徑（當這些指令正被執行時）。替代地，可藉由以其相應於未被遮蔽之所得中的各元件之資訊自動地饋送遮蔽層來實現該旁通。

雖然以上指令之描述包括資料值、將被萃取/插入/排列之向量元件及結果之大小的特定位元寬度，但那些熟悉本項技術人士將理解文中所描述之觀念可被擴充至不同的個別寬度。

圖 5B 顯示一種可由圖 5A 之邏輯電路所履行的方法。將被萃取/插入/排列之輸入向量被置於第一暫存器中 510，及至少假如遮蔽適用的話，一遮罩型態被置於第二

暫存器中。輸入向量之特定元件被接著選擇以包括入其符合指令之所得中 511。假如遮蔽適用的話，遮罩型態被施加 512 至操作 511 之結果以產生所得。該所得被接著寫入至目的地 513。假如遮蔽不適用，則所得為操作 511 之結果。

### 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。

### 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 6A-6B 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 6A 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 6B 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 600 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 605 指令模板和記憶體存取 620 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 6A 中之類別 A 指令模板包括：1) 於無記憶體存取 605 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 610 指令模板及無記憶體存取、資

料轉變類型操作 615 指令模板；以及 2) 於記憶體存取 620 指令模板內顯示有記憶體存取、暫時 625 指令模板及記憶體存取、非暫時 630 指令模板。圖 6B 中之類別 B 指令模板包括：1) 於無記憶體存取 605 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 612 指令模板及無記憶體存取、寫入遮罩控制、vsize 類型操作 617 指令模板；以及 2) 於記憶體存取 620 指令模板內顯示有記憶體存取、寫入遮罩控制 627 指令模板。

一般性向量友善指令格式 600 包括依圖 6A-6B 中所示之順序所列出於下的如下欄位。配合以上之討論，於一實施例中，參考圖 6A-B 及 7 中以下所提出之格式細節，無記憶體存取指令類型 605 或記憶體存取指令類型 620 可被使用。讀取遮罩、輸入向量運算元及目的地之位址可被識別於以下所描述之暫存器位址欄位 644 中。於進一步實施例中，寫入遮罩被指明於寫入遮罩欄位 670 中。

格式欄位 640 - 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 642 - 其內容係分辨不同的基礎操作。

暫存器指標欄位 644 - 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， $32 \times 512$ 、 $16 \times 128$ 、 $32 \times 1024$ 、 $64 \times 1024$ ）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 646 - 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 605 指令模板與記憶體存取 620 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/或目的地地址），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 650 - 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 668、阿爾發欄位 652、及貝他欄位 654。擴增操作欄位 650 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 660 - 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 662A - 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 662B (注意其直接於置換因數欄位 662B 上方的置換欄位 662A 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而因此, 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 674 (文中稍後所描述) 及資料調處欄位 654C 而由處理器硬體判定於運行時間。置換欄位 662A 及置換因數欄位 662B 是選擇性的, 因為其並未用於無記憶體存取 605 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 664 - 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 670 - 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 670 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 670 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 670 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 670 之內容直接地指明應履行之遮蔽。

即刻欄位 672 - 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 668 - 其內容係分辨於不同類別的指令之間。參考圖 6A-B，此欄位之內容係選擇於類別 A 與類別 B 指令之間。於圖 6A-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 6A-B 中的類別欄位 668 之類別 A 668A 及類別 B 668B）。

### 類別 A 之指令模板

於類別 A 之無記憶體存取 605 指令模板的情況下，阿爾發欄位 652 被解讀為 RS 欄位 652A，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 652A.1 及資料轉變 652A.2 被個別地指明給無記憶體存取、捨入類型操作 610 及無記憶體存取、資料轉變類型操作 615 指令模板），而貝他欄位 654 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 605 指令模板中，比率欄位 660、置換欄位 662A、及置換比率欄位 662B 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 610 指令模板中，貝他欄位 654 被解讀為捨入控制欄位 654A，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 654A 包括一抑制所有浮點例外（SAE）欄位 656 及一捨入操作控制欄位 658，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 658）。

SAE 欄位 656 - 其內容係分辨是否使例外事件報告失效；當 SAE 欄位 656 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 658 - 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 658 容許以每指令爲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 650 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 615 指令模板中，貝他欄位 654 被解讀爲資料轉變欄位 654B，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 620 指令模板的情況下，阿爾發欄位 652 被解讀爲逐出（eviction）暗示欄位 652B，其內容係分辨逐出暗示之何者應被使用（於圖 6A 中，暫時 652B.1 及非暫時 652B.2 被個別地指明給記憶體存取、暫時 625 指令模板及記憶體存取、非暫時 630 指令模板），而貝他欄位 654 被解讀爲資料調處欄位 654C，其內容係分辨數個資料調處操作（亦已知爲基元）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 620 指令模板包括比率欄位 660、及選擇性地置換欄位 662A 或置換比率欄位 662B。

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 652 被解讀為寫入遮罩控制 (Z) 欄位 652C，其內容係分辨其由寫入遮罩欄位 670 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 605 指令模板的情況下，貝



他欄位 654 之部分被解讀為 RL 欄位 657A，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 657A.1 及向量長度（VSIZE）657A.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 612 指令模板及無記憶體存取、寫入遮罩控制、VSIZE 類型操作 617 指令模板），而具他欄位 654 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 605 指令模板中，比率欄位 660、置換欄位 662A、及置換比率欄位 662B 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 610 指令模板中，具他欄位 654 之剩餘者被解讀為捨入操作欄位 659A 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理器）。

捨入操作控制欄位 659A - 正如同捨入操作控制欄位 658，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 659A 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 650 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 617 指令模板中，具他欄位 654 被解讀為向量長度欄位 659B，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 620 指令模板的情況下，貝他欄位 654 之部分被解讀為廣播欄位 657B，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 654 之剩餘者被解讀為向量長度欄位 659B。記憶體存取 620 指令模板包括比率欄位 660、及選擇性的置換欄位 662A、或置換比率欄位 662B。

針對一般性向量友善指令格式 600，一全運算碼欄位 674 顯示為包括格式欄位 640、基礎操作欄位 642、及資料元件寬度欄位 664。雖然一實施例係顯示全運算碼欄位 674 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 674 可包括少於所有這些欄位。全運算碼欄位 674 提供運算碼 ( opcode ) 。

擴增操作欄位 650、資料元件寬度欄位 664、及寫入遮罩欄位 670 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 7 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 7 顯示一特定向量友善指令格式 700，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 700 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 7 之欄位的來自圖 6 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 600 之背景下的特定向量友善指令格式 700 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 700。例如，一般性向量友善指令格式 600 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 700 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 664 被顯示為特定向量友善指令格式 700 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 600 係考量資料元件寬度欄位 664 之其他大小）。

一般性向量友善指令格式 600 包括依圖 7A 所示之順序的如下欄位。

EVEX 前綴（位元組 0-3）702 - 被編碼以四位元組之形式。

格式欄位 640（EVEX 位元組 0，位元 [7:0]） - 第一位元組（EVEX 位元組 0）為格式欄位 640 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 705 (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 657BEX 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 610 - 此為 REX'欄位 610 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 715 (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 664 (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 720 (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.vvvv720 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 668 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 725 (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，為了支援其使用 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為傳統 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為運算碼擴充，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 652 (EVEX 位元組 3, 位元 [7] - EH; 亦已知為 EVEX.EH、EVEX.rs、EVEX.RL、EVEX.寫入遮罩控制、及 EVEX.N; 亦以  $\alpha$  顯示) - 如先前所述，此欄位為背景特定的。

貝他欄位 654 (EVEX 位元組 3, 位元 [6:4] - SSS, 亦已知為 EVEX.s<sub>2-0</sub>、EVEX.r<sub>2-0</sub>、EVEX.rr1、EVEX.LL0、EVEX.LLB; 亦以  $\beta \beta \beta$  顯示) - 如先前所述，此欄位為背景特定的。

REX'欄位 610 - 此為 REX'欄位之剩餘者且為可用以將擴充的 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 670 (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  $E\text{VEX.kkk}=000$  具有一特殊行爲，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 730（位元組 4）亦已知爲運算碼位元組。運算碼之部分被指明於此欄位中。

Mod R/M 欄位 740（位元組 5）包括 MOD 欄位 742、Reg 欄位 744、及 R/M 欄位 746。如先前所述，MOD 欄位 742 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 744 之角色可被概述爲兩種情況：編碼目的地暫存器運算元或來源暫存器運算元之任一者、或者被視爲運算碼擴充且不被用於編碼任何指令運算元。R/M 欄位 746 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 650 之內容被用於記憶體位址產生。SIB.xxx 754 及 SIB.bbb 756 - 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 662A（位元組 7-10）- 當 MOD 欄位 742 含有 10 時，位元組 7-10 爲置換欄位 662A，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 662B（位元組 7）- 當 MOD 欄位 742 含有 01 時，位元組 7 爲置換因數欄位 662B。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 662B 為 disp8 之再解讀；當使用置換因數欄位 662B 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 662B 取代傳統 x86 指令集 8 位元置換。因此，置換因數欄位 662B 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 672 係操作如先前所述。

### 全運算碼欄位

圖 7B 為方塊圖，其說明組成全運算碼欄位 674 之特

定向量友善指令格式 700 的欄位，依據本發明之一實施例。明確地，全運算碼欄位 674 包括格式欄位 640、基礎操作欄位 642、及資料元件寬度 (W) 欄位 664。基礎操作欄位 642 包括前綴編碼欄位 725、運算碼映圖欄位 715、及真實運算碼欄位 730。

### 暫存器指標欄位

圖 7C 為方塊圖，其說明組成暫存器指標欄位 644 之特定定向量友善指令格式 700 的欄位，依據本發明之一實施例。明確地，暫存器指標欄位 644 包括 REX 欄位 705、REX' 欄位 710、MODR/M.reg 欄位 744、MODR/M.r/m 欄位 746、VVVV 欄位 720、xxx 欄位 754、及 bbb 欄位 756。

### 擴增操作欄位

圖 7D 為方塊圖，其說明組成擴增操作欄位 650 之特定定向量友善指令格式 700 的欄位，依據本發明之一實施例。當類別 (U) 欄位 668 含有 0 時，其表示 EVEX.U0 (類別 A 668A)；當其含有 1 時，其表示 EVEX.U1 (類別 B 668B)。當 U=0 且 MOD 欄位 742 含有 11 (表示無記憶體存取操作) 時，阿爾發欄位 652 (EVEX 位元組 3, 位元 [7] - EH) 被解讀為 rs 欄位 652A。當 rs 欄位 652A 含有一個 1 (捨入 1252A.1) 時，貝他欄位 654 (EVEX 位元組 3, 位元 [6:4] - SSS) 被解讀為捨入控制欄位 654A。

捨入控制欄位 654A 包括一位元 SAE 欄位 656 及二位元捨入操作欄位 658。當 rs 欄位 652A 含有 0 (資料轉變 652A.2) 時，貝他欄位 654 (EVEX 位元組 3,位元 [6:4] - SSS) 被解讀為三位元資料轉變欄位 654B。當 U=0 且 MOD 欄位 742 含有 00、01、或 10 (表示記憶體存取操作) 時，阿爾發欄位 652 (EVEX 位元組 3,位元 [7] - EH) 被解讀為逐出暗示 (EH) 欄位 652B 而貝他欄位 654 (EVEX 位元組 3,位元 [6:4] - SSS) 被解讀為三位元資料調處欄位 654C。

當 U=1 時，阿爾發欄位 652 (EVEX 位元組 3,位元 [7] - EH) 被解讀為寫入遮罩控制 (Z) 欄位 652C。當 U=1 且 MOD 欄位 742 含有 11 (表示無記憶體存取操作) 時，貝他欄位 654 之部分 (EVEX 位元組 3,位元 [4] - S<sub>0</sub>) 被解讀為 RL 欄位 657A；當其含有 1 (捨入 657A.1) 時，貝他欄位 654 之剩餘者 (EVEX 位元組 3,位元 [6-5] - S<sub>2-1</sub>) 被解讀為捨入操作欄位 659A；而當 RL 欄位 657A 含有 0 (VSIZE 1257.A2) 時，貝他欄位 654 之剩餘者 (EVEX 位元組 3,位元 [6-5] - S<sub>2-1</sub>) 被解讀為向量長度欄位 659B (EVEX 位元組 3,位元 [6-5] - L<sub>1-0</sub>)。當 U=1 且 MOD 欄位 742 含有 00、01、或 10 (表示記憶體存取操作) 時，貝他欄位 654 (EVEX 位元組 3,位元 [6:4] - SSS) 被解讀為向量長度欄位 659B (EVEX 位元組 3,位元 [6-5] - L<sub>1-0</sub>) 及廣播欄位 657B (EVEX 位元組 3,位元 [4] - B)。

### 範例暫存器架構

圖 8 為依據本發明之一實施例的暫存器架構 800 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 810；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 700 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 659B 之指令模板	A (圖 6A ; U=0)	610、615、625、630	zmm 暫存器 (向量長度為 64 位元組)
	B (圖 6B ; U=1)	612	zmm 暫存器 (向量長度為 64 位元組)
包括向量長度欄位 659B 之指令模板	B (圖 6B ; U=1)	617、627	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 659B

換言之，向量長度欄位 659B 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 659B 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 700 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 815 - 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 1415 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 825 - 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）845，於其上係混疊 `MMX` 緊縮整數平坦暫存器檔案 850 - 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 9A 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 9B 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 9A-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 9A 中，處理器管線 900 包括提取級 902、長度解碼級 904、解碼級 906、配置級 908、重新命名級 910、排程（亦已知為調度 (dispatch) 或發出）級 912、暫存器讀取/記憶體讀取級 914、執行級 916、寫回/記憶體寫入級 918、異常處置級 922、及確定 (commit) 級 924。

圖 9B 顯示處理器核心 990，其包括耦合至執行引擎單元 950 之前端單元 930，兩者均耦合至記憶體單元 970。核心 990 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 990 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 930 包括一耦合至指令快取單元 934 之分支預測單元 932，指令快取單元 934 係耦合至指令翻譯旁看

緩衝器 (TLB) 936，指令翻譯旁看緩衝器 (TLB) 936 係耦合至指令提取單元 938，指令提取單元 938 係耦合至解碼單元 940。解碼單元 940 (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 940 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 990 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 940 中或者另外於前端單元 930 內)。解碼單元 940 係耦合至執行引擎單元 950 中之重新命名/配置器單元 952。

執行引擎單元 950 包括重新命名/配置器單元 952，其係耦合至收回單元 954 及一組一或更多排程器單元 956。排程器單元 956 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 956 被耦合至實體暫存器檔案單元 958。實體暫存器檔案單元 958 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 958 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 958 被收回單元 954 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 954 及實體暫存器檔案單元 958 被耦合至執行叢集 960。執行叢集 960 包括一組一或更多執行單元 962 及一組一或更多記憶體存取單元 964。執行單元 962 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 956、實體暫存器檔案單元 958、執行叢集 960 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 964）。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 964 被耦合至記憶體單元 970，

其包括資料 TLB 單元 972，其耦合至資料快取單元 974，其耦合至第二階（L2）快取單元 976。於一範例實施例中，記憶體存取單元 964 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 970 中之資料 TLB 單元 972。指令快取單元 934 被進一步耦合至記憶體單元 970 中之第二階（L2）快取單元 976。L2 快取單元 976 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 900 如下：1) 指令提取 938 履行提取和長度解碼級 902 和 904；2) 解碼單元 940 履行解碼級 906；3) 重新命名/配置器單元 952 履行配置級 908 和重新命名級 910；4) 排程器單元 956 履行排程級 912；5) 實體暫存器檔案單元 958 和記憶體單元 970 履行暫存器讀取/記憶體讀取級 914；執行叢集 960 履行執行級 916；6) 記憶體單元 970 和實體暫存器檔案單元 958 履行寫回/記憶體寫入級 118；7) 各種單元可被關聯於異常處置級 922；及 8) 收回單元 954 和實體暫存器檔案單元 958 履行確定級 924。

核心 990 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 990 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 934/974 以及共用的第二階（L2）快取單元 976，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 10 A-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 10A 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 1002 且具有其第二階(L2)快取之局部子集 1004。於一實施例中,指令解碼器 1000 支援具有緊縮資料指令集擴充之 x86 指令集。L1 快取 1006 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 1008 和向量單元 1010 係使用分離的暫存器組(個別地,純量暫存器 1012 和向量暫存器 1014)且於其間轉移之資料被寫入至記憶體並從第一階(L1)快取 1006 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 1004 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 1004 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 1004 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 1004 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 10B 為依據本發明之實施例的圖 10A 中之處理器核心的部分之擴充視圖。圖 10B 包括 L1 快取 1004 之 L1 資料快取 1006A 部分，以及有關向量單元 1010 及向量暫存器 1014 之更多細節。明確地，向量單元 1010 為 16 寬的向量處理單元 (VPU) (參見 16 寬的 ALU 1028)，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 1020 以拌和暫存器輸入、利用數字轉換單元 1022A-B 之數字轉換、及利用記憶體輸入上之複製單元 1024 的複製。寫入遮罩暫存器 1026 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 11 為依據本發明之實施例的處理器 1100 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 11 中之實線方塊係說明一具有單一核心 1102A、系統代理器 1110、一組一或更多匯流排控制器單元 1116 之處理器 1100，而虛線方塊之額外加入則說明一具有多個核心 1102A-N、系統代理器 1110 中之一組一或更多集成記憶體控制器單元 1114、及特殊用途邏輯 1108 之處理器 1100。

因此，處理器 1100 之不同實施方式可包括：1) 具有

特殊用途邏輯 1108 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 1102A-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 1102A-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 1102A-N 之共處理器為大量通用依序核心。因此，處理器 1100 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 1100 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 1106、及耦合至該組集成記憶體控制器單元 1114 之外部記憶體（未顯示）。該組共用快取單元 206 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 1112 係互連集成圖形邏輯 1108、該組共用快取單元 1106、以及系統代理器單元 1110/集成記憶體控制器單元 1114，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 1106 與核心 1102A-N 之間。

於某些實施例中，一或更多核心 1102A-N 能夠進行多執行緒。系統代理器 1110 包括那些協調及操作核心 1102A-N 之組件。系統代理器單元 1110 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 1102A-N 和集成圖形邏輯 1108 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 1102A-N 可為同質的或異質的；亦即，二或更多核心 1102A-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 12-15 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 12，其顯示依據本發明之一實施例的系統 1200 之方塊圖。系統 1200 可包括一或更多處理器 1210、1215，其被耦合至控制器集線器 1220。於一實施例中，控制器集線器 1220 包括一圖形記憶體控制器集線器 (GMCH) 1290 及一輸入/輸出集線器 (IOH) 1250 (其可於分離的晶片上)；GMCH 1290 包括記憶體和圖形控制器，其係耦合記憶體 1240 和共處理器 1245；IOH 1250 將輸入/輸出 (I/O) 裝置 1260 耦合至 GMCH 1290。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 1240 和共處理器 1245 被直接耦合至處理器 1210、以及一具有 IOH 1250 之單一晶片中的控制器集線器 1220。

額外處理器 1215 之選擇性本質係以虛線被標示於圖 3 中。各處理器 1210、1215 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 1100。

記憶體 1240 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 1220 通連與處理器 1210、1215，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 1295。

於一實施例中，共處理器 1245 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 1220 可包括一集成圖形加速器。

有多種差異於實體資源 1210、1215 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 1210 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 1210 識別這些共處理器指令為應由附加共處理器 1245 所執行之類型。因此，處理器 1210 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 1245。共處理器 1245 接受並執行所接收的共處理器指令。

現在參考圖 13，其顯示依據本發明之一實施例的第一更特定範例系統 1300 之方塊圖。如圖 13 中所示，多處理器系統 1300 為點對點互連系統，並包括經由點對點互連 1350 而耦合之第一處理器 1370 和第二處理器 1380。處理器 1370 與 1380 之每一者可為相同版本的處理器 1100。於本發明之一實施例中，處理器 1370 和 1380 個別為處理器 1210 和 1215，而共處理器 1338 為共處理器 1245。於另一實施例中，處理器 1370 和 1380 個別為處理器 1210 和共處理器 1245。

處理器 1370 和 1380 被顯示為個別地包括集成記憶體控制器（IMC）單元 1372 和 1382。處理器 1370 亦包括點對點（P-P）介面 1376 和 1378 為其匯流排控制器單元之部分；類似地，第二處理器 1380 包括 P-P 介面 1386 和

1388。處理器 1370、1380 可使用 P-P 介面電路 1378、1388 而經由點對點 (P-P) 介面 1350 以交換資訊。如圖 13 中所示，IMC 1372 和 1382 將處理器耦合至個別記憶體，亦即記憶體 1332 和記憶體 1334，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 1370、1380 可各使用點對點介面電路 1376、1394、1386、1398 而經由個別 P-P 介面 1352、1354 與晶片組 1390 交換資訊。晶片組 1390 可選擇性地經由高性能介面 1339 而與共處理器 1338 交換資訊。於一實施例中，共處理器 1338 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 1390 可經由介面 1396 而被耦合至第一匯流排 1316。於一實施例中，第一匯流排 1316 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 13 中所示，各種 I/O 裝置 1314 可被耦合至第一匯流排 1316，連同一將第一匯流排 1316 耦合至第二匯流排 1320 之匯流排橋 1318。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 1315 被耦合至第一匯流排 1316。於一實施例中，第二匯流排 1320 為低接腳數（LPC）匯流排。各種裝置可被耦合至第二匯流排 420，包括（例如）鍵盤及 / 或滑鼠 1322、通訊裝置 1327 及儲存單元 1328，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 1330，於一實施例中。再者，聲頻 I/O 1324 可被耦合至第二匯流排 1320。注意：其他架構是可能的。例如，取代圖 13 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 14，其顯示依據本發明之一實施例的第二更特定範例系統 1400 之方塊圖。圖 13 和 14 中之類似元件係使用類似的參考數字，且圖 14 已省略了圖 13 之某些形態以避免混淆圖 14 之其他形態。

圖 14 顯示其處理器 1370、1380 可個別地包括集成記憶體和 I/O 控制邏輯（「CL」）1372 和 1382。因此，CL 1372、1382 包括集成記憶體控制器單元並包括 I/O 控制邏輯。圖 14 顯示其不僅記憶體 1332、1334 耦合至 CL 1372、1382，同時 I/O 裝置 1414 亦耦合至控制邏輯 1372、1382。傳統 I/O 裝置 1415 被耦合至晶片組 1390。

現在參考圖 15，其顯示依據本發明之一實施例的 SoC 1500 之方塊圖。與圖 11 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 15 中，互連單元 1502 被耦合至：應用程式處理器 1510，其包括一組一或更多核心 202A-N 及共用快取單元 1106；系統代理器單元 1110；匯流排控制器單元 1116；集成記憶體控制器單元 1114；一組一或更多共處理器 1520，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 1530；直接記憶體存取（DMA）單元 1532；及顯示單元 1540，用以耦合至一或更多外部顯示。於一實施例中，共處理器 1520 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 13 中所示之碼 1330）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 16 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 16 顯示高階語言 1602 之程式可使用 x86 編譯器 1604 來編譯以產生 x86 二元碼 1606，其可由一具有至少一 x86 指令集核心之處理器 1616 所本地執行。具有至少一 x86 指令集核心之處理器 1616 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 1604 代表一種能夠產生 x86 二元碼 1606 (例如, 物件碼) 之編譯器, x86 二元碼 706 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 1616 上。類似地, 圖 16 顯示高階語言 1602 之程式可使用替代指令集編譯器 1608 而被編譯以產生替代的指令集二元碼 1610, 其可由一不具有至少一 x86 指令集核心之處理器 1614 所本地地執行 (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 1612 被用以將 x86 二元碼 1606 轉換為可由不具有至少一 x86 指令集核心之處理器 1614 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 1610, 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 1612 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 1606。

#### 【圖式簡單說明】

本發明係經由後附圖式之圖形中的範例 (而非限制) 來說明, 其中類似參考數字係指示類似元件且其中:

圖 1 顯示一指令執行管線；

圖 2A 及 2B 比較純量相對於向量處理；

圖 3A 至 3E 顯示習知技術插入、萃取及排列指令；

圖 4A 至 4P 顯示改良的插入、萃取及排列指令；

圖 5A 及 5B 係有關用以實施改良的插入、萃取及排列指令之執行邏輯電路；

圖 6A-6B 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（template）；

圖 7 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 8 為依據本發明之一實施例的暫存器架構之方塊圖；

圖 9A 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 9B 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 10A-B 說明一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的數個邏輯區塊（包括相同類型及/或不同類型之其他核心）之一。

圖 11 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 12 為依據本發明之一實施例的系統之方塊圖；

圖 13 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 14 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 15 為依據本發明之一實施例的晶片上系統 (SoC) 之方塊圖；

圖 16 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：處理核心

101：管線

102：暫存器儲存空間

103\_1-103\_N：功能單元

104：向量暫存器空間

104\_1-104\_N：遮蔽邏輯

105\_1-105\_N：遮蔽邏輯

106：遮罩暫存器空間

107：向量暫存器儲存空間

501：第一暫存器

502：邏輯電路

503：輸出

- 504 : 遮蔽層邏輯電路
- 505 : 暫存器
- 506 : 暫存器
- 507 : 暫存器
- 510 : 輸入暫存器
- 600 : 一般性向量友善指令格式
- 605 : 無記憶體存取
- 610 : 無記憶體存取、全捨入控制類型操作
- 612 : 無記憶體存取、寫入遮罩控制、部分捨入控制  
類型操作
- 615 : 無記憶體存取、資料轉變類型操作
- 617 : 無記憶體存取、寫入遮罩控制、`vsize` 類型操作
- 620 : 記憶體存取
- 627 : 記憶體存取、寫入遮罩控制
- 640 : 格式欄位
- 642 : 基礎操作欄位
- 644 : 暫存器指標欄位
- 646 : 修飾符欄位
- 650 : 擴增操作欄位
- 652 : 阿爾發欄位
- 652A : RS 欄位
- 652A.1 : 捨入
- 652A.2 : 資料轉變
- 652B : 逐出暗示欄位

- 652B.1 : 暫時
- 652B.2 : 非暫時
- 654 : 貝他欄位
- 654A : 捨入控制欄位
- 654B : 資料轉變欄位
- 654C : 資料調處欄位
- 656 : SAE 欄位
- 657A : RL 欄位
- 657A.1 : 捨入
- 657A.2 : 向量長度 ( VSIZE )
- 657B : 廣播欄位
- 658 : 捨入操作控制欄位
- 659A : 捨入操作欄位
- 659B : 向量長度欄位
- 660 : 比率欄位
- 662A : 置換欄位
- 662B : 置換因數欄位
- 664 : 資料元件寬度欄位
- 668 : 類別欄位
- 668A : 類別 A
- 668B : 類別 B
- 670 : 寫入遮罩欄位
- 672 : 即刻欄位
- 674 : 全運算碼欄位

- 700 : 特定向量友善指令格式
- 702 : EVEX 前綴
- 705 : REX 欄位
- 710 : REX' 欄位
- 715 : 運算碼映圖欄位
- 720 : VVVV 欄位
- 725 : 前綴編碼欄位
- 730 : 真實運算碼欄位
- 740 : Mod R/M 位元組
- 742 : MOD 欄位
- 744 : Reg 欄位
- 746 : R/M 欄位
- 754 : SIB.xxx
- 756 : SIB.bbb
- 800 : 暫存器架構
- 810 : 向量暫存器
- 815 : 寫入遮罩暫存器
- 825 : 通用暫存器
- 845 : 純量浮點堆疊暫存器檔案
- 850 : MMX 緊縮整數平坦暫存器檔案
- 900 : 處理器管線
- 902 : 提取級
- 904 : 長度解碼級
- 906 : 解碼級

- 908 : 配置級
- 910 : 重新命名級
- 912 : 排程級
- 914 : 暫存器讀取 / 記憶體讀取級
- 916 : 執行級
- 918 : 寫回 / 記憶體寫入級
- 922 : 異常處置級
- 924 : 確定級
- 930 : 前端單元
- 932 : 分支預測單元
- 934 : 指令快取單元
- 936 : 指令翻譯旁看緩衝器 ( TLB )
- 938 : 指令提取單元
- 940 : 解碼單元
- 950 : 執行引擎單元
- 952 : 重新命名 / 配置器單元
- 954 : 收回單元
- 956 : 排程器單元
- 958 : 實體暫存器檔案單元
- 960 : 執行叢集
- 962 : 執行單元
- 964 : 記憶體存取單元
- 970 : 記憶體單元
- 972 : 資料 TLB 單元

- 974 : 資料快取單元
- 976 : 第二階 ( L2 ) 快取單元
- 990 : 處理器核心
- 1000 : 指令解碼器
- 1002 : 晶粒上互連網路
- 1004 : 第二階 ( L2 ) 快取之局部子集
- 1006 : L1 快取
- 1006A : L1 資料快取
- 1008 : 純量單元
- 1010 : 向量單元
- 1012 : 純量暫存器
- 1014 : 向量暫存器
- 1020 : 拌和單元
- 1022A-B : 數字轉換單元
- 1024 : 複製單元
- 1026 : 寫入遮罩暫存器
- 1028 : 16 寬的 ALU
- 1100 : 處理器
- 1102A-N : 核心
- 1106 : 共用快取單元
- 1108 : 特殊用途邏輯
- 1110 : 系統代理器
- 1112 : 環狀為基的互連單元
- 1114 : 集成記憶體控制器單元

- 1116：匯流排控制器單元
- 1200：系統
- 1210, 1215：處理器
- 1220：控制器集線器
- 1240：記憶體
- 1245：共處理器
- 1250：輸入/輸出集線器 ( IOH )
- 1260：輸入/輸出 ( I/O ) 裝置
- 1290：圖形記憶體控制器集線器 ( GMCH )
- 1295：連接
- 1300：多處理器系統
- 1314：I/O 裝置
- 1315：額外處理器
- 1316：第一匯流排
- 1318：匯流排橋
- 1320：第二匯流排
- 1322：鍵盤及/或滑鼠
- 1324：聲頻 I/O
- 1327：通訊裝置
- 1328：儲存單元
- 1330：指令/碼及資料
- 1332：記憶體
- 1334：記憶體
- 1338：共處理器

- 1339 : 高性能介面
- 1350 : 點對點互連
- 1352, 1354 : P-P 介面
- 1370 : 第一處理器
- 1372, 1382 : 集成記憶體控制器 (IMC) 單元
- 1376, 1378 : 點對點 (P-P) 介面
- 1380 : 第二處理器
- 1386, 1388 : P-P 介面
- 1390 : 晶片組
- 1394, 1398 : 點對點介面電路
- 1396 : 介面
- 1400 : 系統
- 1414 : I/O 裝置
- 1415 : 傳統 I/O 裝置
- 1500 : SoC
- 1502 : 互連單元
- 1510 : 應用程式處理器
- 1520 : 共處理器
- 1530 : 靜態隨機存取記憶體 (SRAM) 單元
- 1532 : 直接記憶體存取 (DMA) 單元
- 640 : 顯示單元
- 1602 : 高階語言
- 1604 : x86 編譯器
- 1606 : x86 二元碼



1608 : 指令集編譯器

1610 : 指令集二元碼

1612 : 指令轉換器

1614 : 不具有至少一 x86 指令集核心之處理器

1616 : 具有至少一 x86 指令集核心之處理器

1354 : P-P 介面

## 七、申請專利範圍

1. 一種用於萃取之設備，包含：

指令執行邏輯電路，用以執行：

第一指令及第二指令，其中該第一指令和該第二指令兩者從個別第一和第二輸入向量的多個第一無重疊區段之一選擇第一族群的輸入向量元件，該第一族群具有第一位元寬度，該些多個第一無重疊區段之每一者具有如該第一族群之相同位元寬度；及

第三指令及第四指令，其中，該第三指令和該第四指令兩者從個別第三和第四輸入向量的多個第二無重疊區段之一選擇第二族群的輸入向量元件，該第二族群具有大於該第一位元寬度之第二位元寬度，該些多個第二無重疊區段之每一者具有如該第二族群之相同位元寬度；及

遮蔽層電路，用以依據該第一和第三指令之個別即刻運算元而遮蔽該第一和第三指令之該第一和第二族群於第一粒度，如此產生之個別所得為該第一和第三指令之個別所得，及依據該第二和第四指令之個別即刻運算元而遮蔽該第二和第四指令之該第一和第二族群於第二粒度，如此產生之個別所得為該第二和第四指令之個別所得。

2. 如申請專利範圍第 1 項之設備，其中該第二族群為該第一族群之大小的兩倍。

3. 如申請專利範圍第 2 項之設備，其中該第一族群為 128 位元而該第二族群為 256 位元。

4. 如申請專利範圍第 1 項之設備，其中該第一粒度

為 32 位元而該第二粒度為 64 位元。

5. 如申請專利範圍第 1 項之設備，其中該些向量元件為浮點資料值。

6. 如申請專利範圍第 1 項之設備，其中假如該元件之位置被遮蔽，則該遮蔽層電路容許目的地向量中之元件維持。

7. 如申請專利範圍第 1 項之設備，其中假如該元件之位置被遮蔽，則該遮蔽層電路以固定值覆寫目的地向量中之元件。

8. 一種用於萃取之方法，包含：

執行第一指令，包括：從第一輸入向量的多個第一無重疊區段之一選擇第一族群的輸入向量元件，該第一族群具有第一位元寬度，該些多個第一無重疊區段之每一者具有如該第一族群之相同位元寬度、及依據該第一指令之即刻運算元而遮蔽該選擇的第一族群於第一粒度；

執行第二指令，包括：從第二輸入向量的多個第二無重疊區段之一選擇第二族群的輸入向量元件，該第二族群具有該第一位元寬度，該第二輸入向量的該些多個第二無重疊區段之每一者具有如該第一族群之相同位元寬度、及依據該第二指令之即刻運算元而遮蔽該選擇的第二族群於第二粒度，該第一粒度較該第二粒度更細；

執行第三指令，包括：從第三輸入向量的多個第三無重疊區段之一選擇第三族群的輸入向量元件，該第三族群具有第二位元寬度，該些多個第三無重疊區段之每一者具

有如該第三族群之相同位元寬度，該第二位元寬度較該第一位元寬度更大、及依據該第三指令之即刻運算元而遮蔽該選擇的第三族群於該第二粒度；

執行第四指令，包括：從第四輸入向量的多個第四無重疊區段之一選擇第四族群的輸入向量元件，該第四族群具有該第二位元寬度，該第四輸入向量的該些多個第四無重疊區段之每一者具有如該第三族群之相同位元寬度、及依據該第四指令之即刻運算元而遮蔽該選擇的第四族群於該第一粒度。

9. 如申請專利範圍第 8 項之方法，其中該第一、第二、第三及第四指令係由指令執行管線內之相同執行單元所執行。

10. 如申請專利範圍第 8 項之方法，其中該第二族群為該第一族群之大小的兩倍。

11. 如申請專利範圍第 10 項之方法，其中該第一族群為 128 位元而該第二族群為 256 位元。

12. 如申請專利範圍第 8 項之方法，其中該第一粒度為 32 位元而該第二粒度為 64 位元。

13. 如申請專利範圍第 8 項之方法，其中該些向量元件為浮點資料值。

14. 如申請專利範圍第 8 項之方法，其中該遮蔽包括以下之任一者：假如該元件之位置被遮蔽則容許目的地向量中之元件維持；及假如該元件之位置被遮蔽則以固定值覆寫目的地向量中之元件。

15. 一種用於萃取之設備，包含：

指令執行邏輯電路，用以執行：

第一指令及第二指令，其中該第一指令和該第二指令兩者依據第一和第二個別即刻運算元而從個別第一和第二輸入向量的多個第一無重疊區段之一選擇第一族群的輸入向量元件，該第一族群具有第一位元寬度，該些多個第一無重疊區段之每一者具有如該第一族群之相同位元寬度；及

第三指令及第四指令，其中該第三指令和該第四指令兩者依據第三和第四個別即刻運算元而從個別第三和第四輸入向量的多個第二無重疊區段之一選擇第二族群的輸入向量元件，該第二族群具有大於該第一位元寬度之第二位元寬度，該些多個第二無重疊區段之每一者具有如該第二族群之相同位元寬度；及

遮蔽層電路，用以依據該第一和第三指令之個別即刻運算元而遮蔽該第一和第三指令之該第一和第二族群於第一粒度，如此產生之個別所得為該第一和第三指令之個別所得，及依據該第二和第四指令之個別即刻運算元而遮蔽該第二和第四指令之該第一和第二族群於第二粒度，如此產生之個別所得為該第二和第四指令之個別所得。

16. 如申請專利範圍第 15 項之設備，其中該第二族群為該第一族群之大小的兩倍。

17. 如申請專利範圍第 16 項之設備，其中該第一族群為 128 位元而該第二族群為 256 位元。

18. 如申請專利範圍第 15 項之設備，其中該第一粒度為 32 位元而該第二粒度為 64 位元。

19. 如申請專利範圍第 15 項之設備，其中該些向量元件為浮點資料值。

20. 如申請專利範圍第 15 項之設備，其中該遮蔽包括以下之任一者：假如該元件之位置被遮蔽則容許目的地向量中之元件維持；及假如該元件之位置被遮蔽則以固定值覆寫目的地向量中之元件。

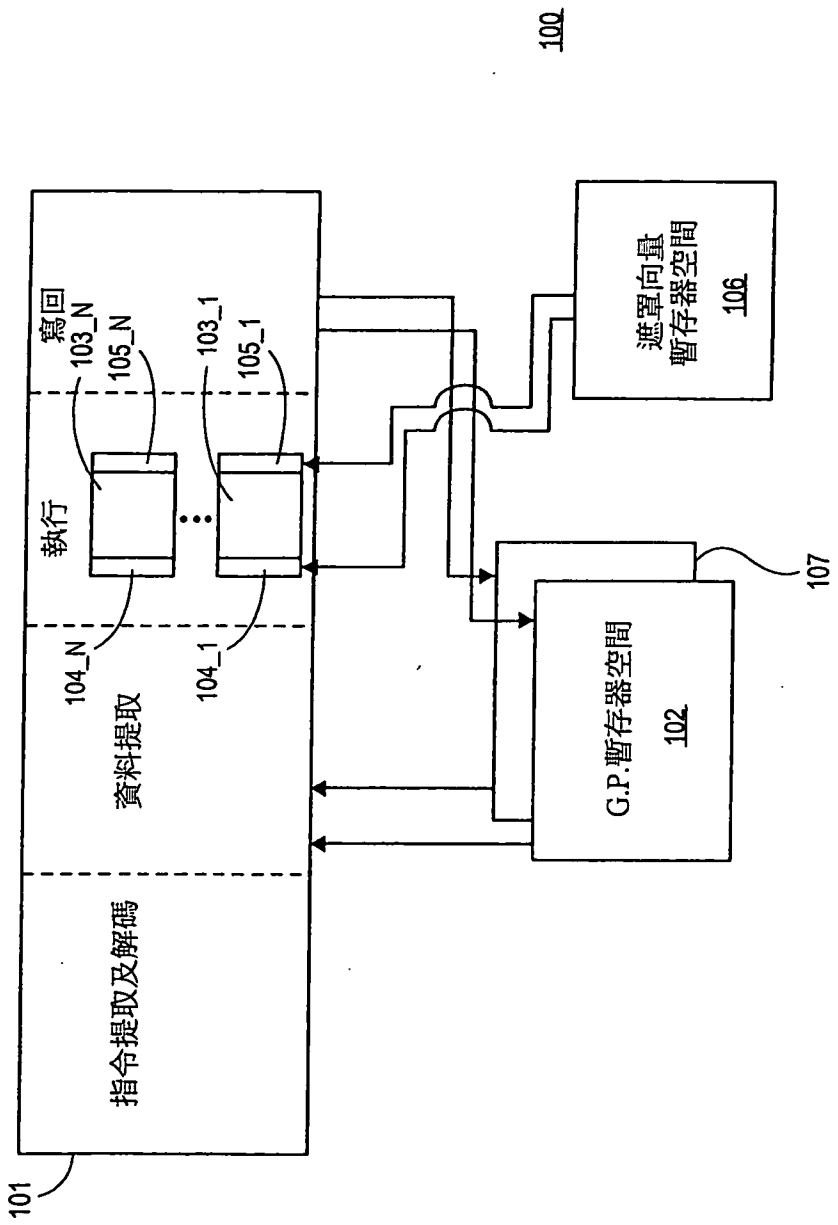


圖1

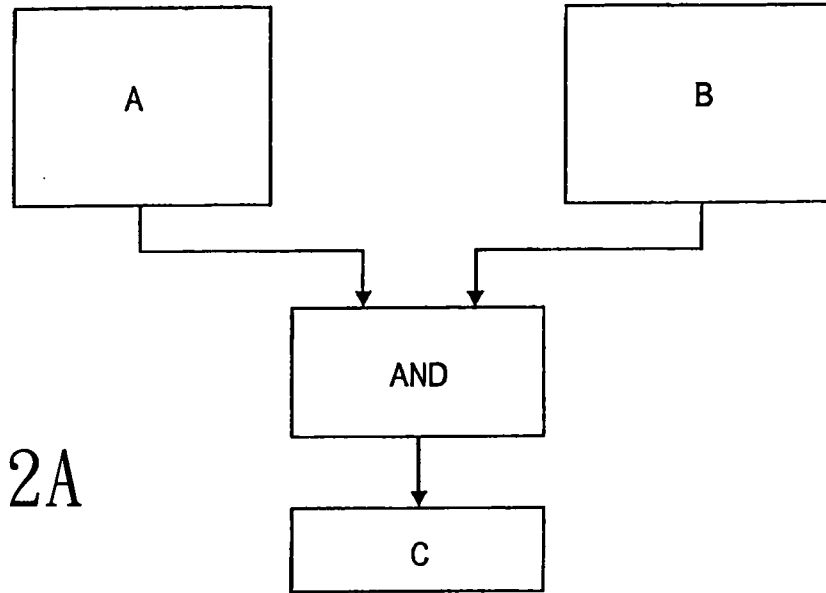


圖 2A

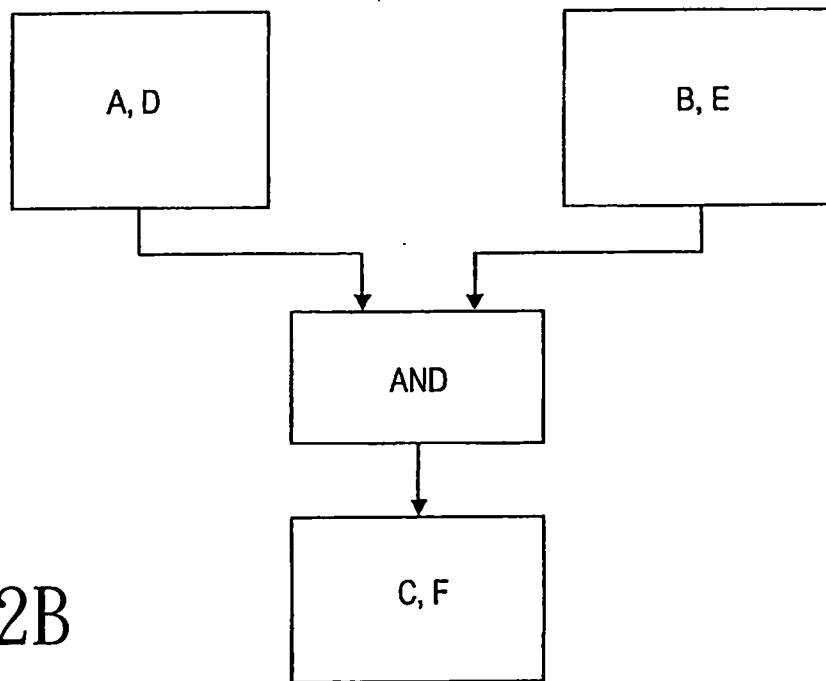
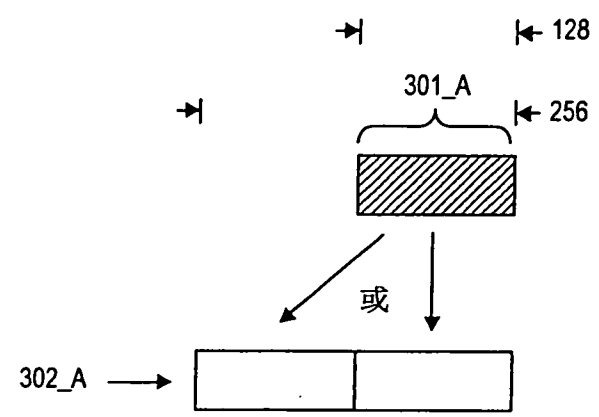


圖 2B



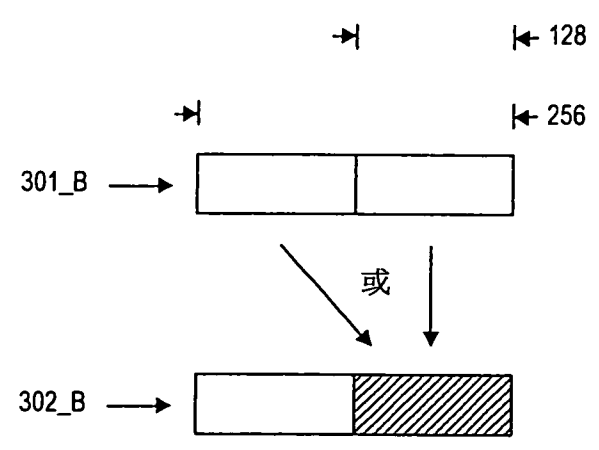
VINSERT

圖 3A



VEXTRACT

圖 3B



VPERMILPS

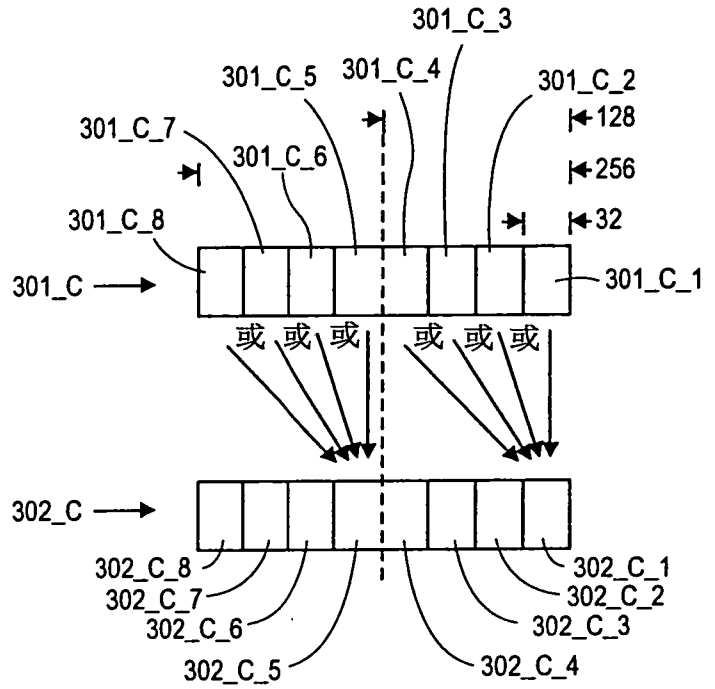


圖 3C

VPERMILPD

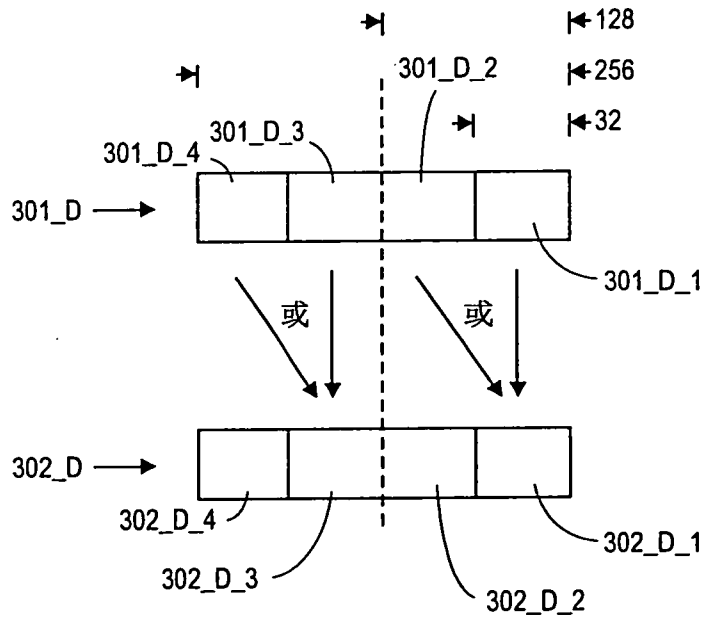


圖 3D

VPERM2F128

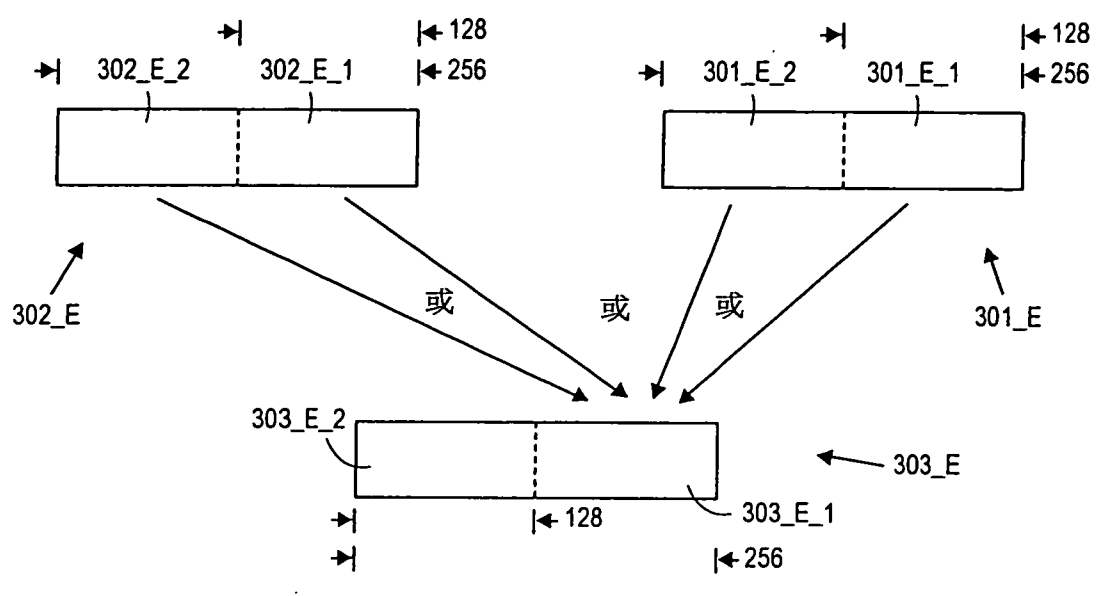


圖 3E

VINSERTF32X4

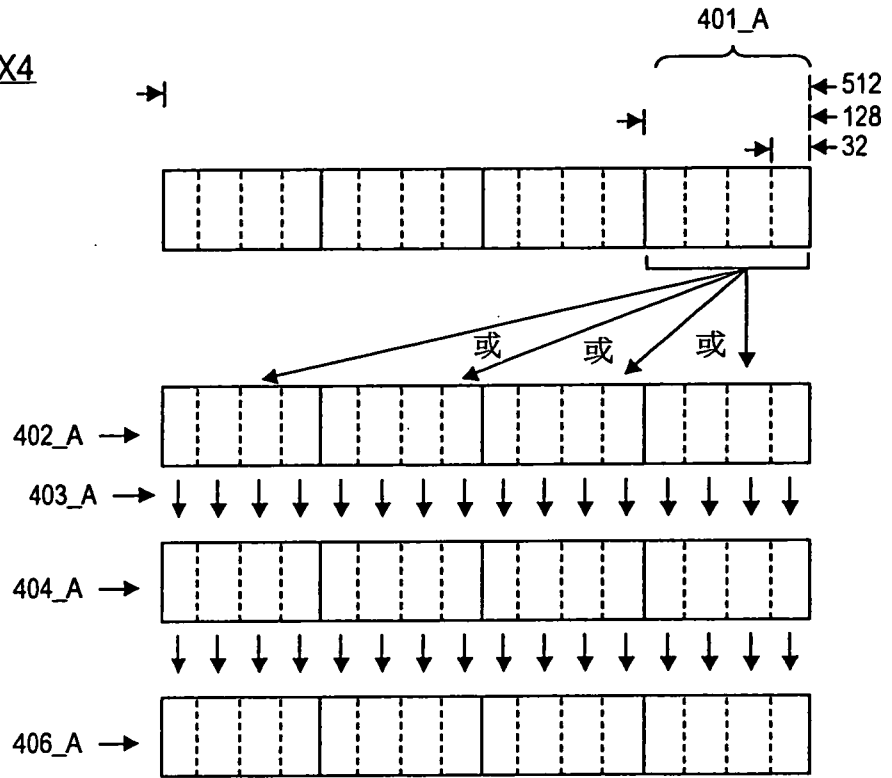


圖 4A

VINSERTF64X2

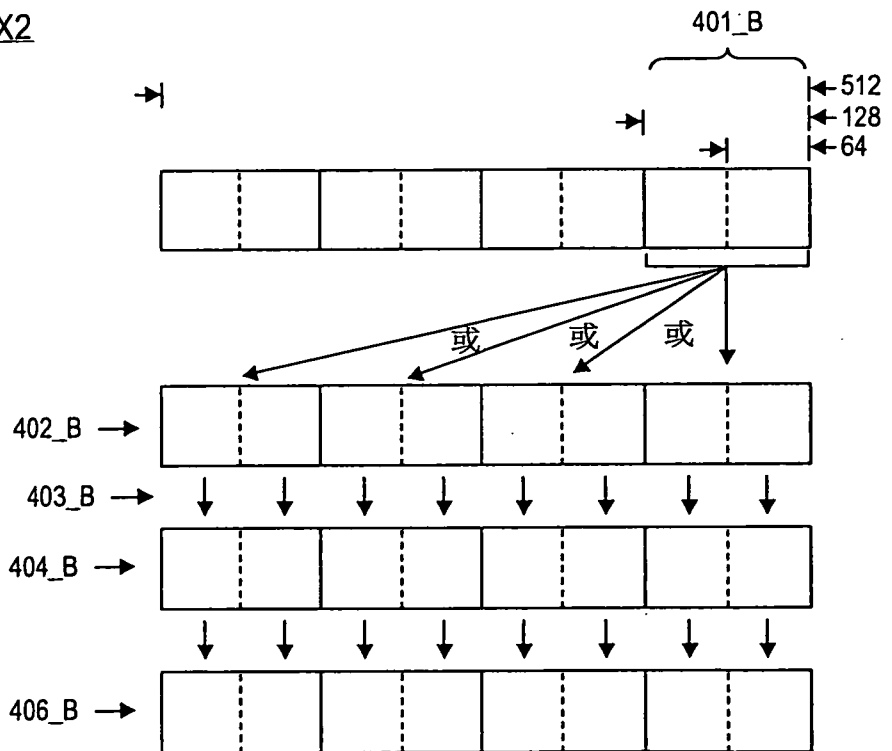


圖 4B

VINSERTF32X4

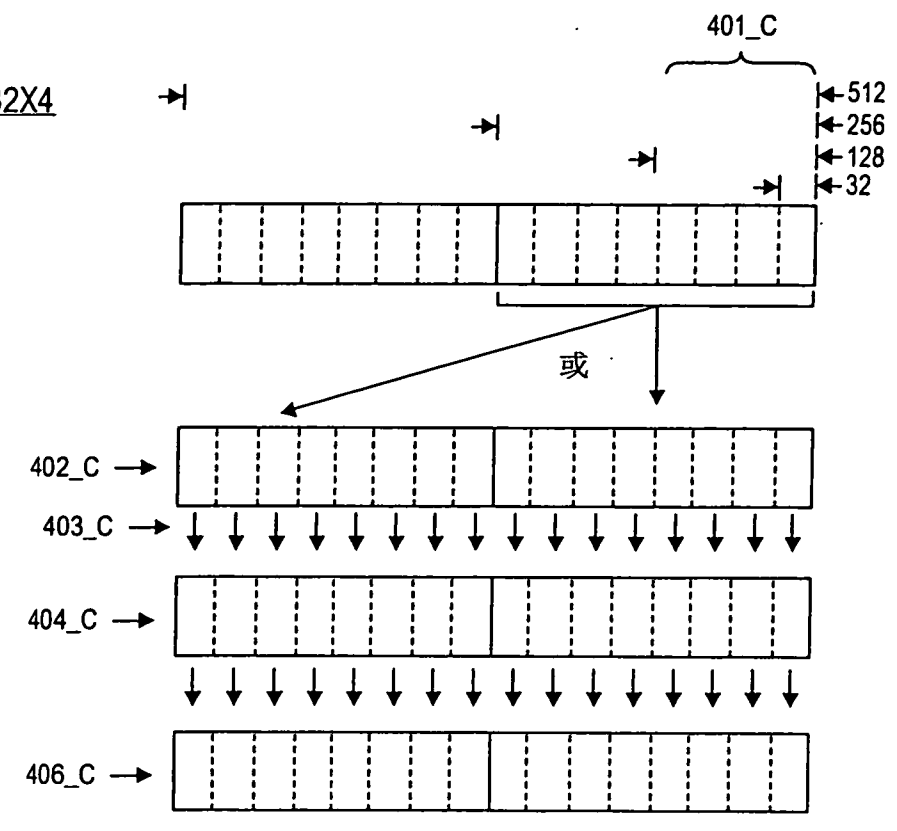


圖 4C

VINSERTF64X4

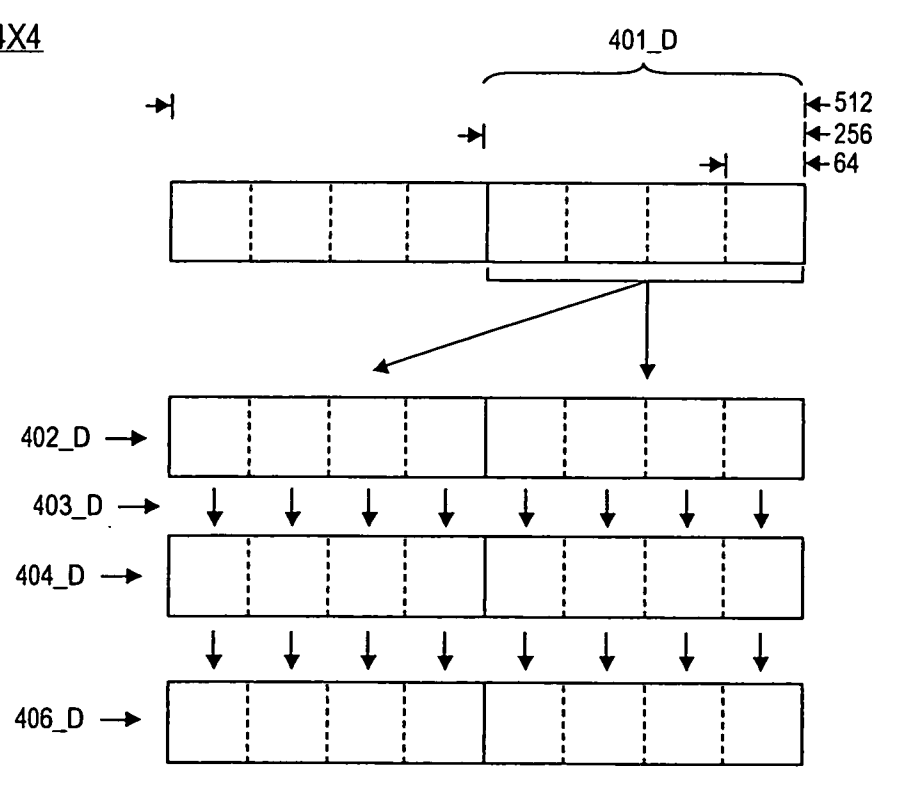


圖 4D

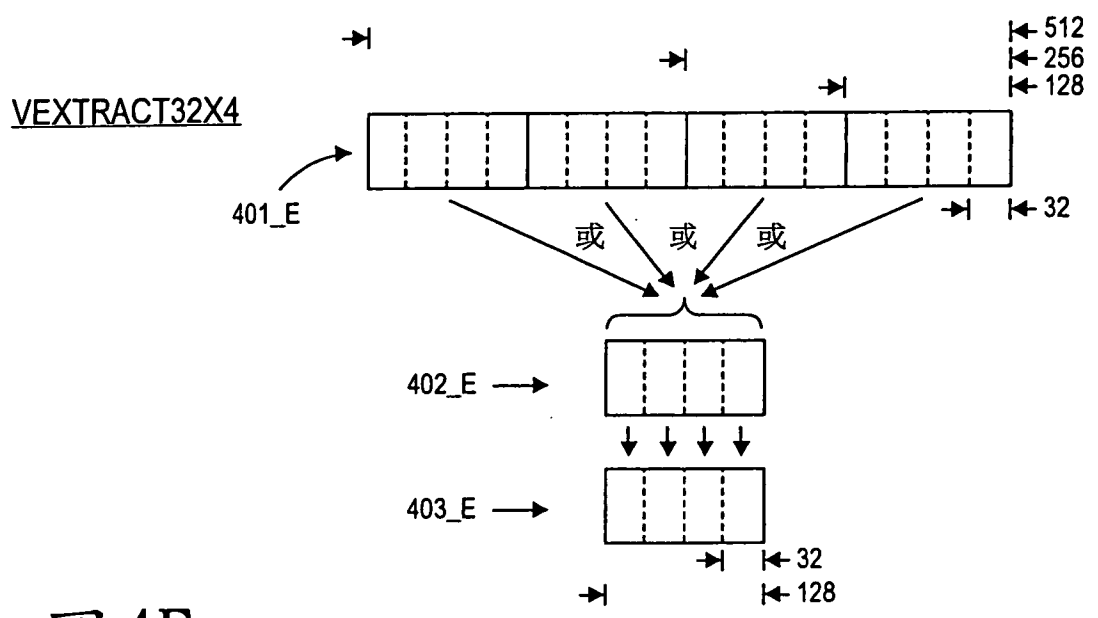


圖 4E

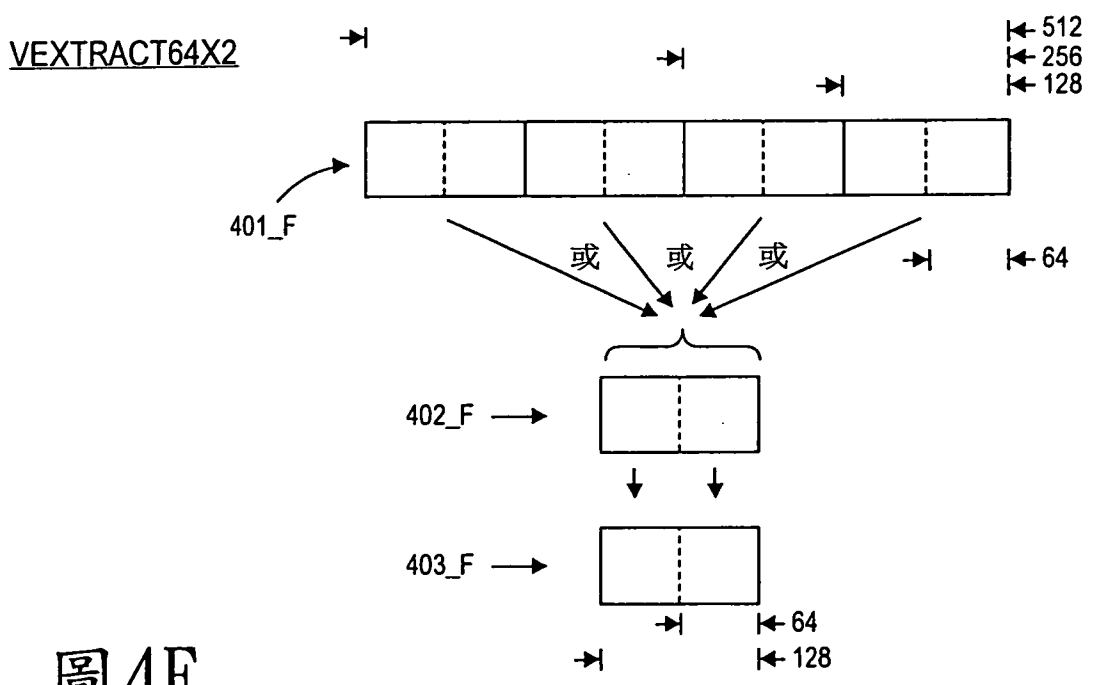


圖 4F

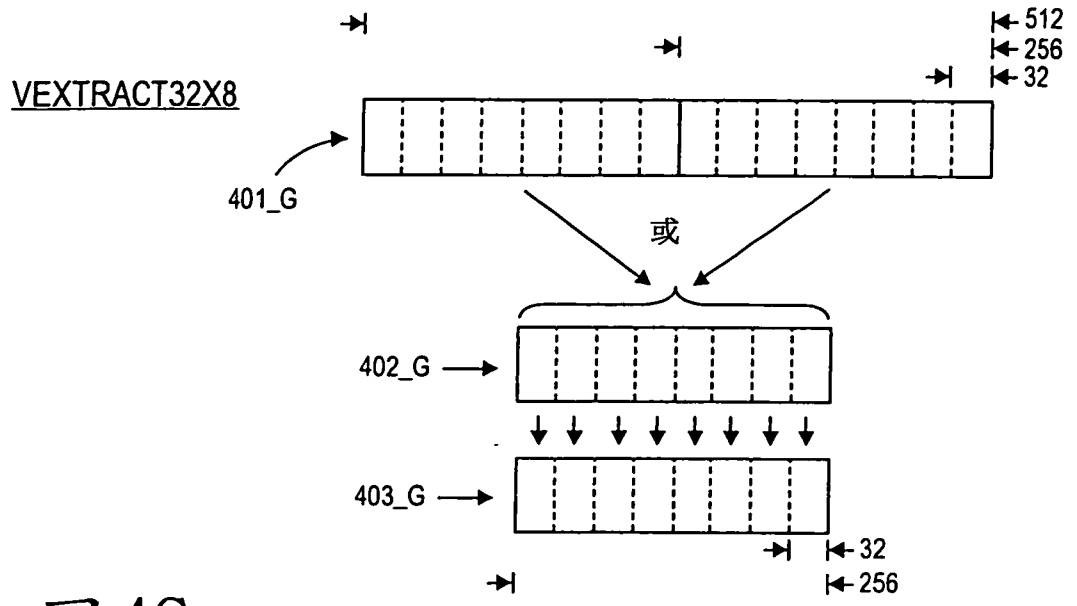


圖 4G

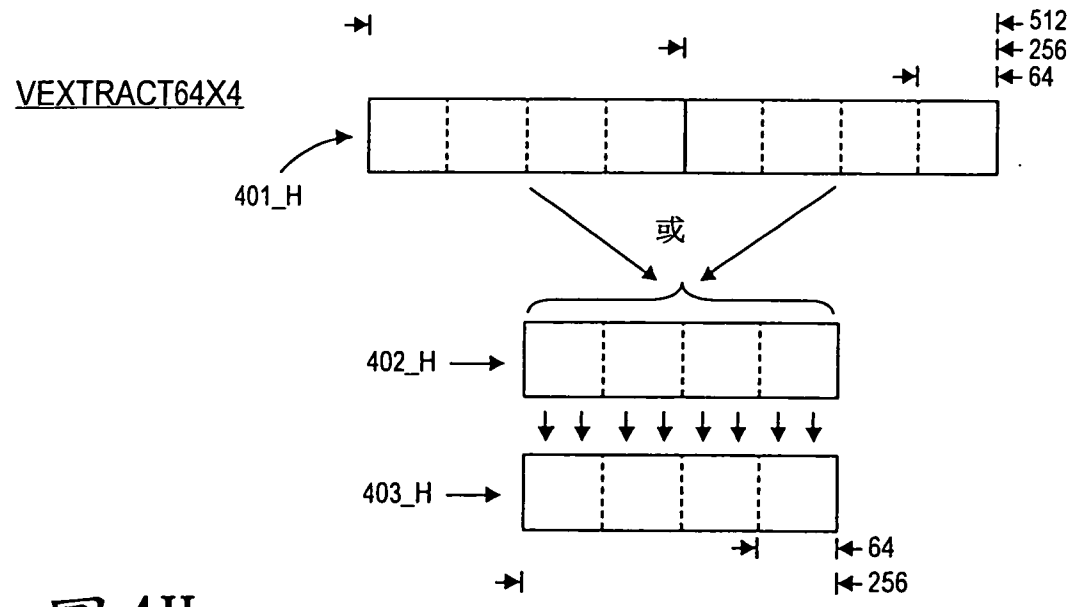


圖 4H

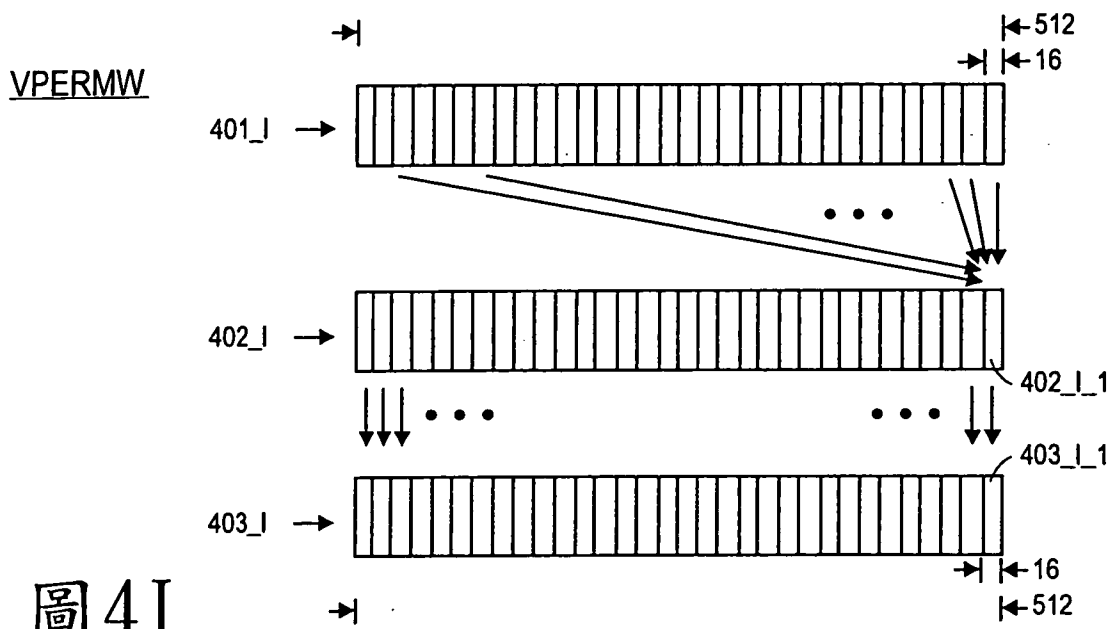


圖4I

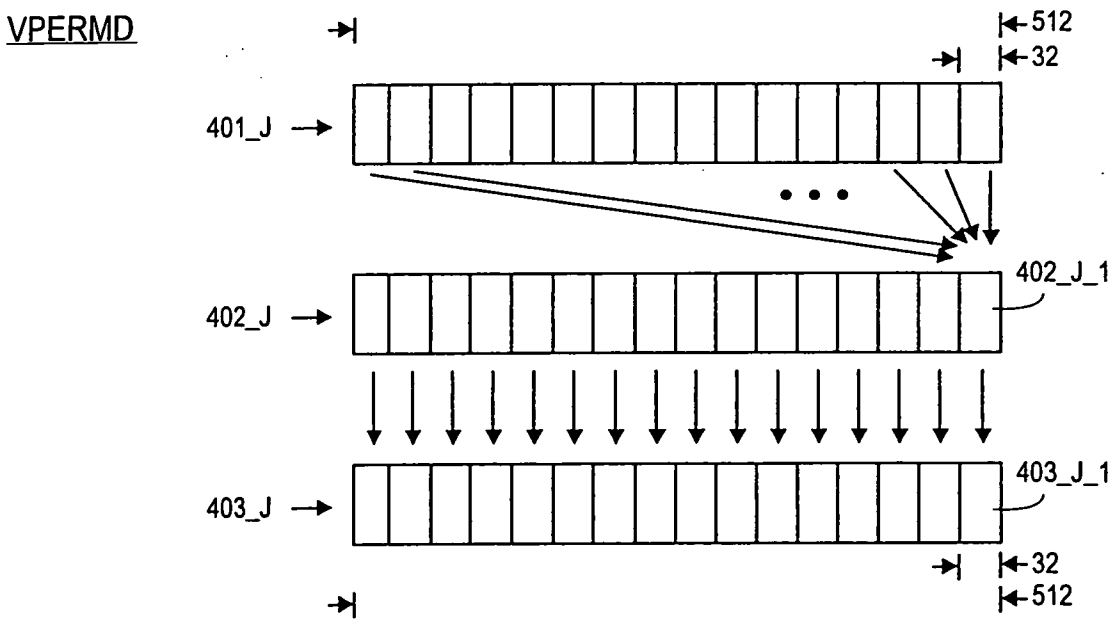


圖4J



VPERMQ

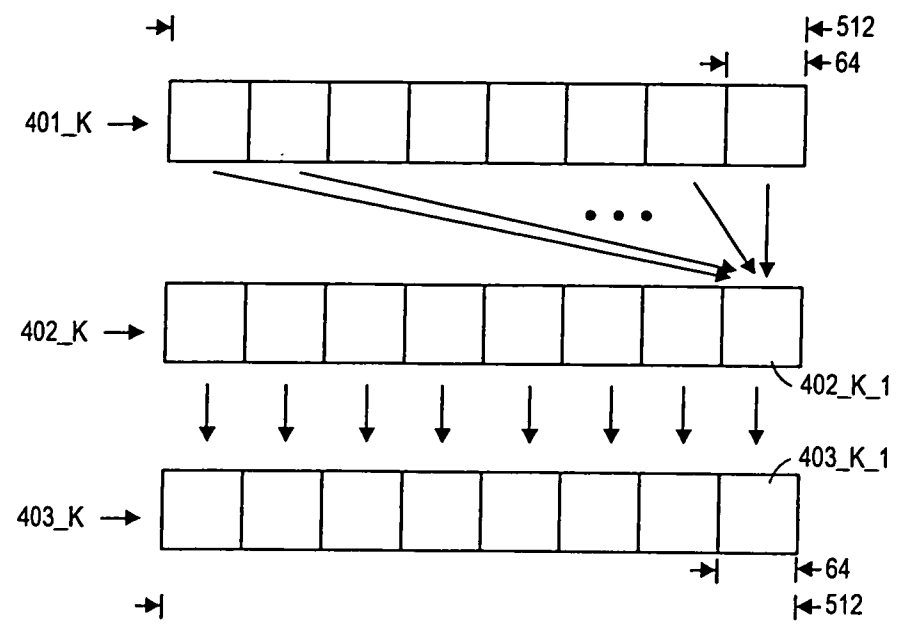


圖 4K

VPERMILPS

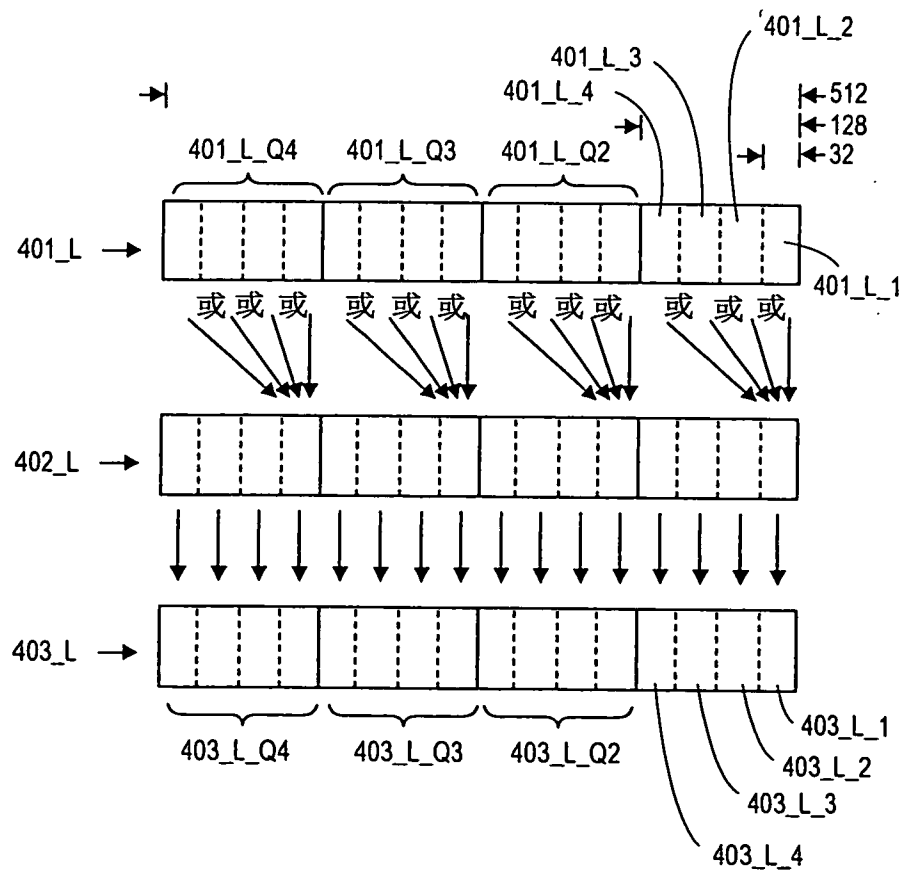


圖 4L

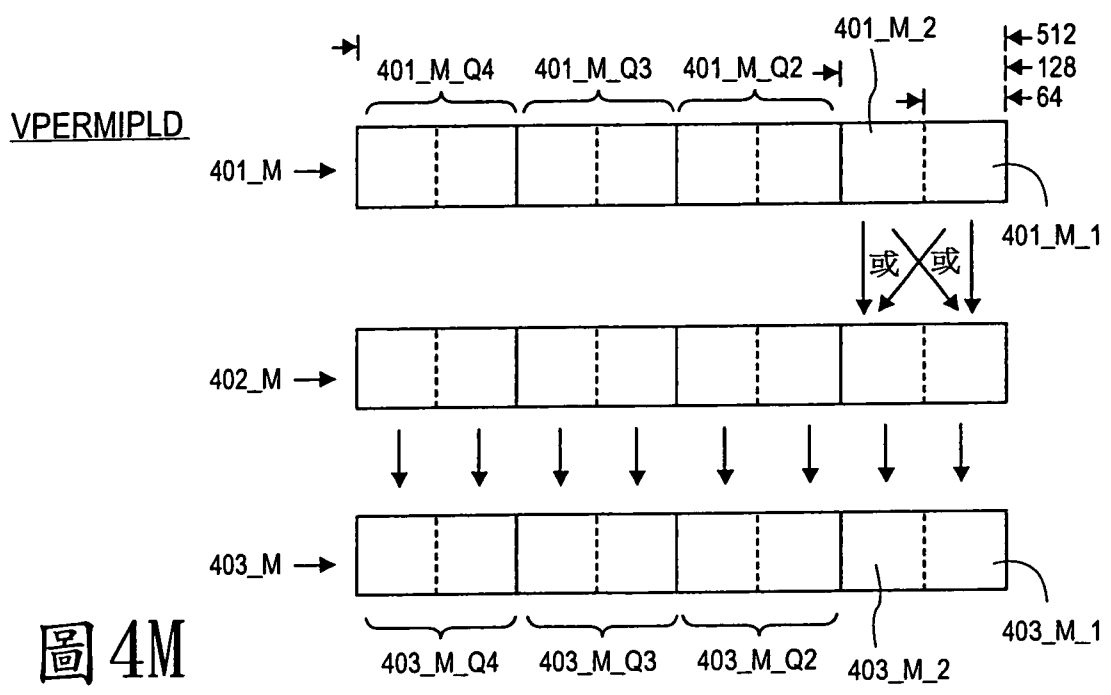


圖 4M

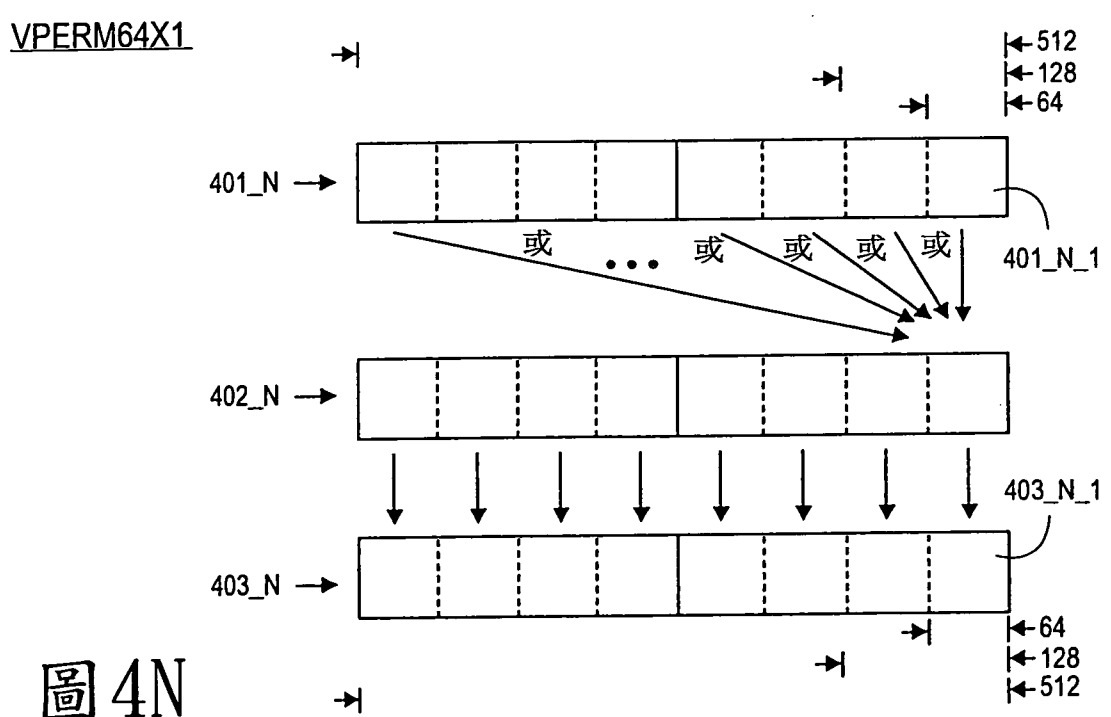


圖 4N

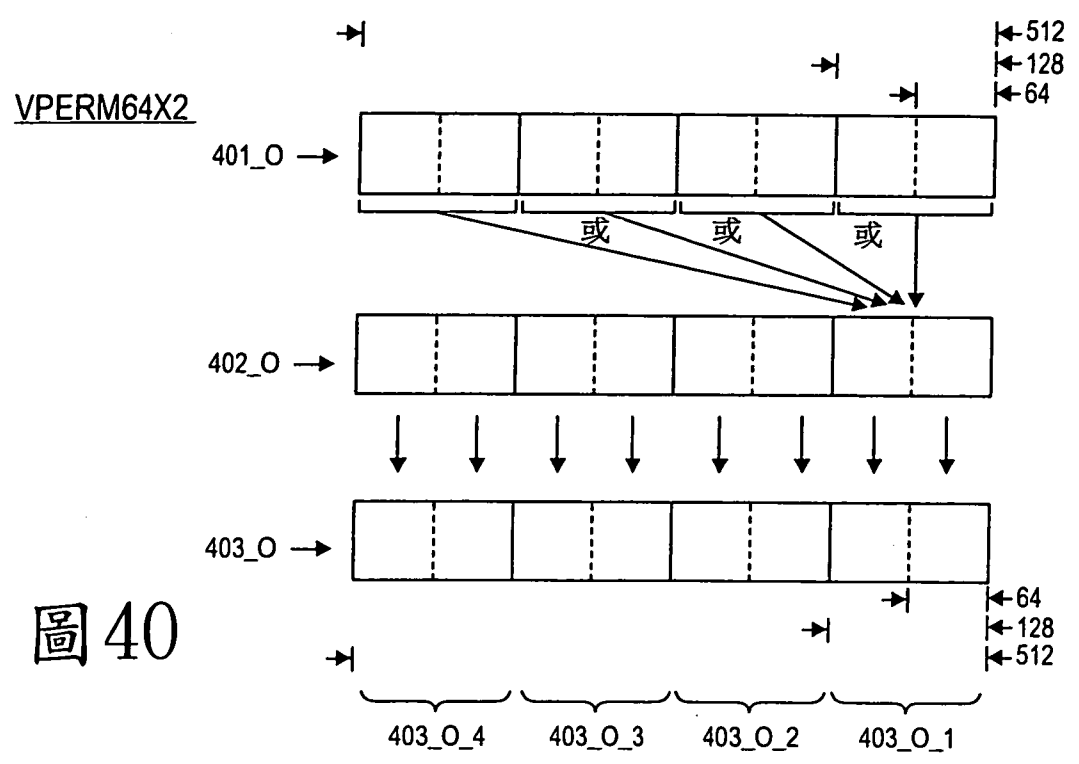


圖 40

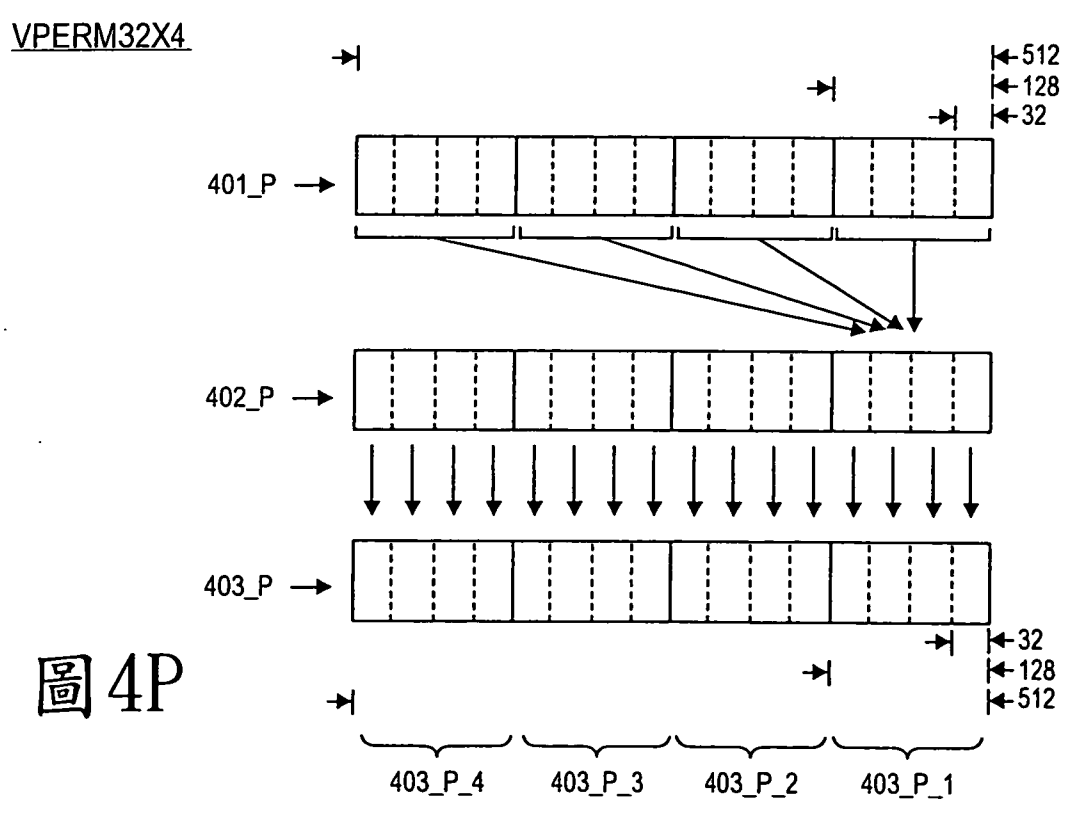


圖 4P

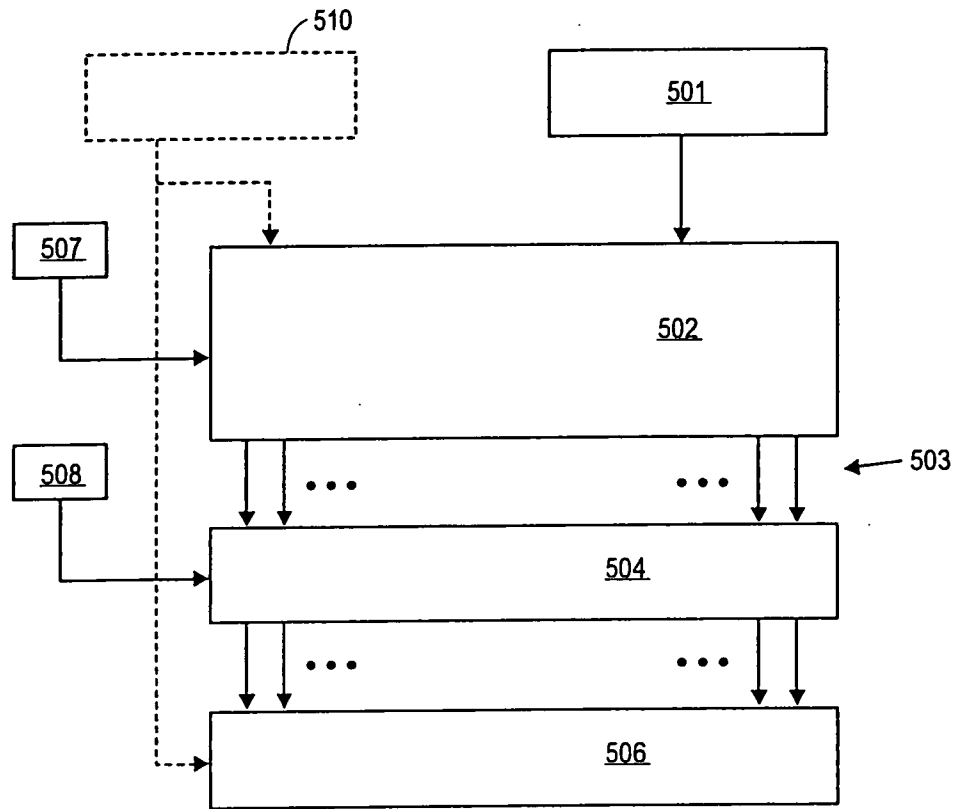


圖5A

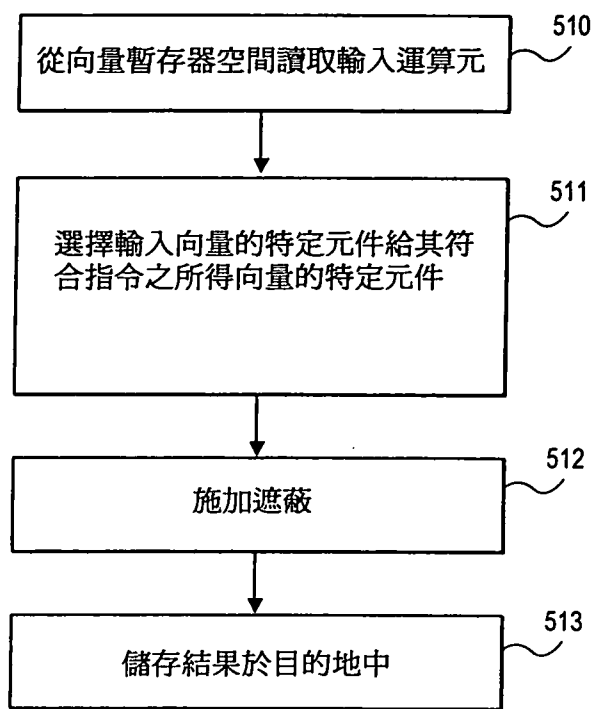


圖 5B

圖 6A

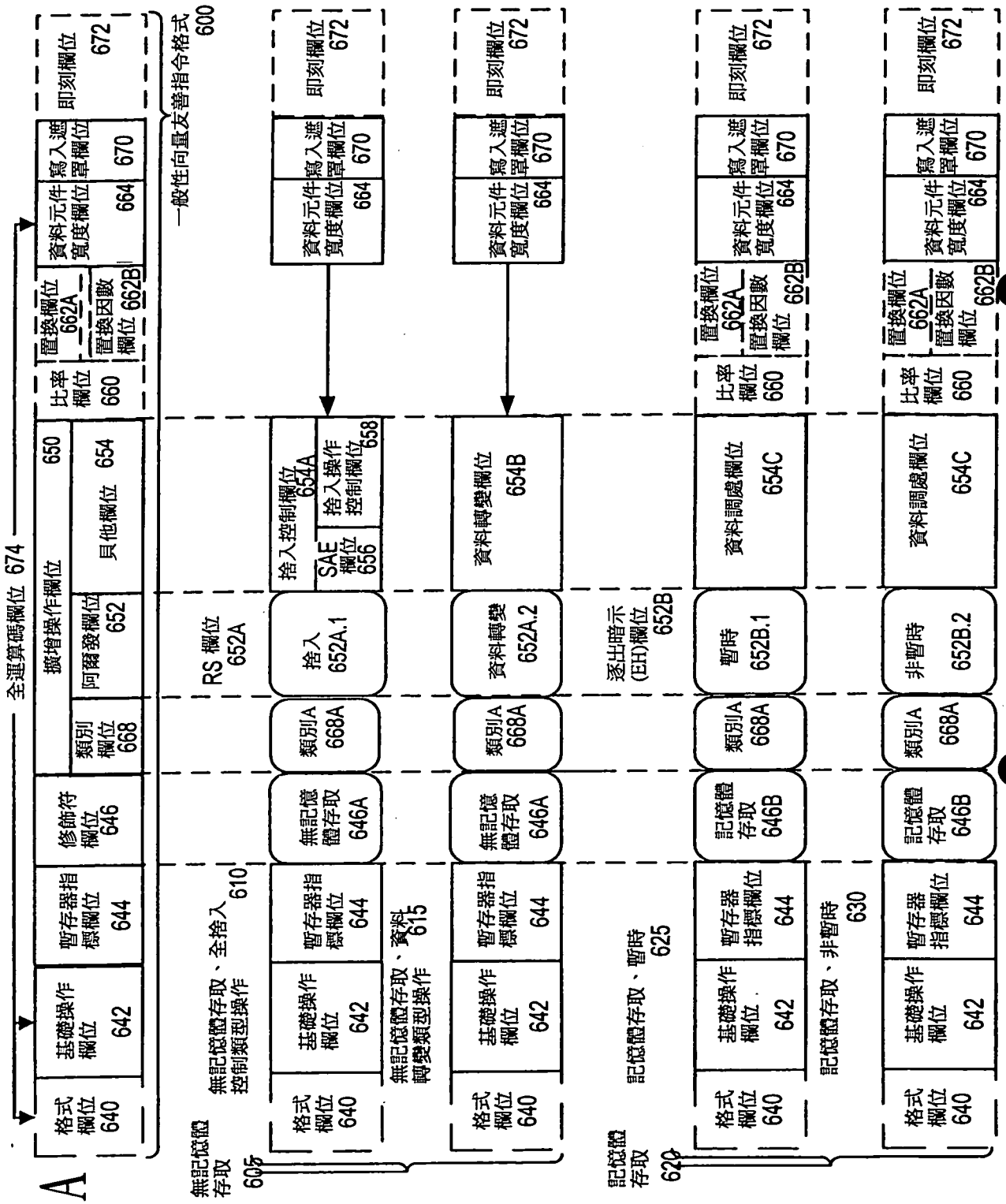


圖 6B

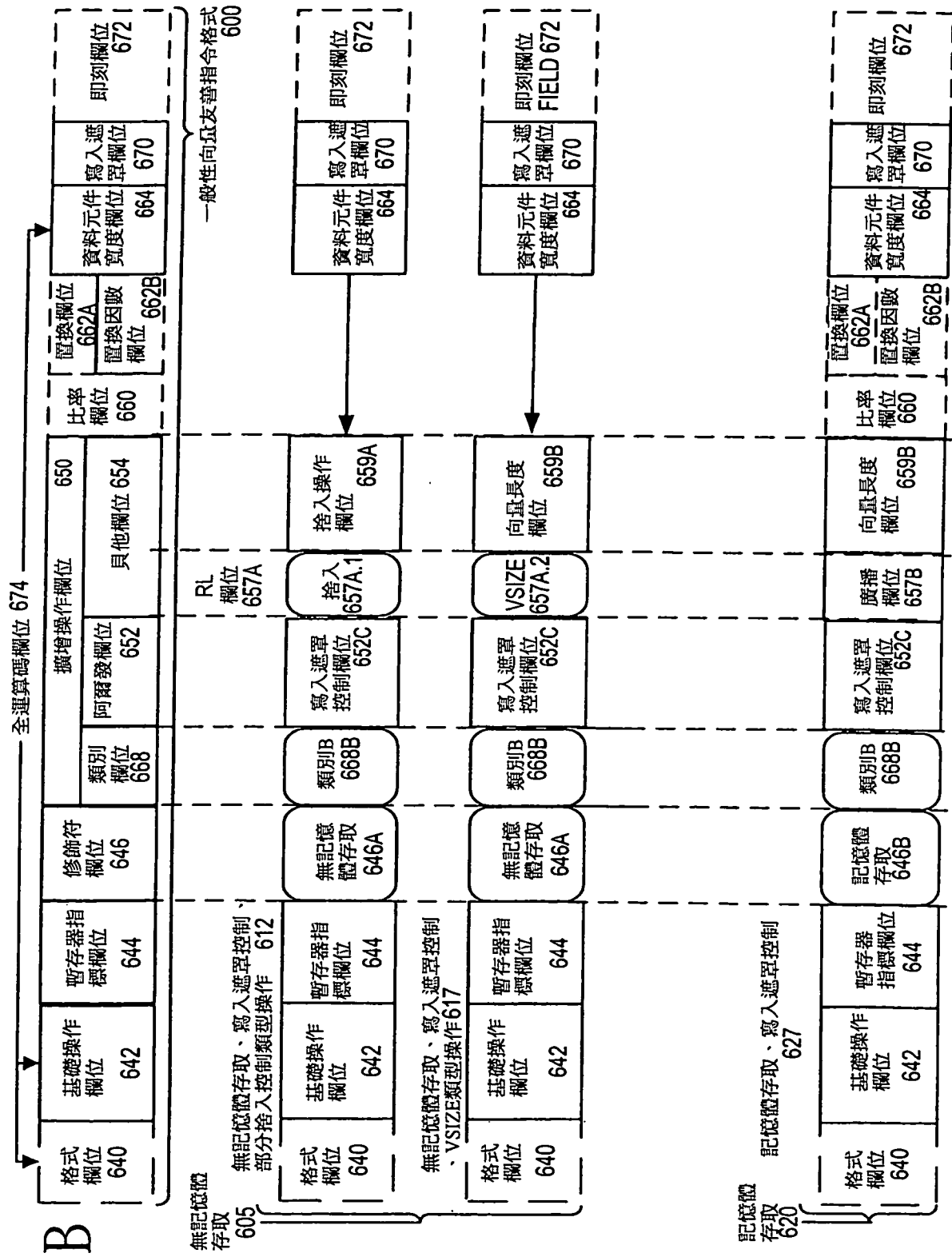


圖 7A

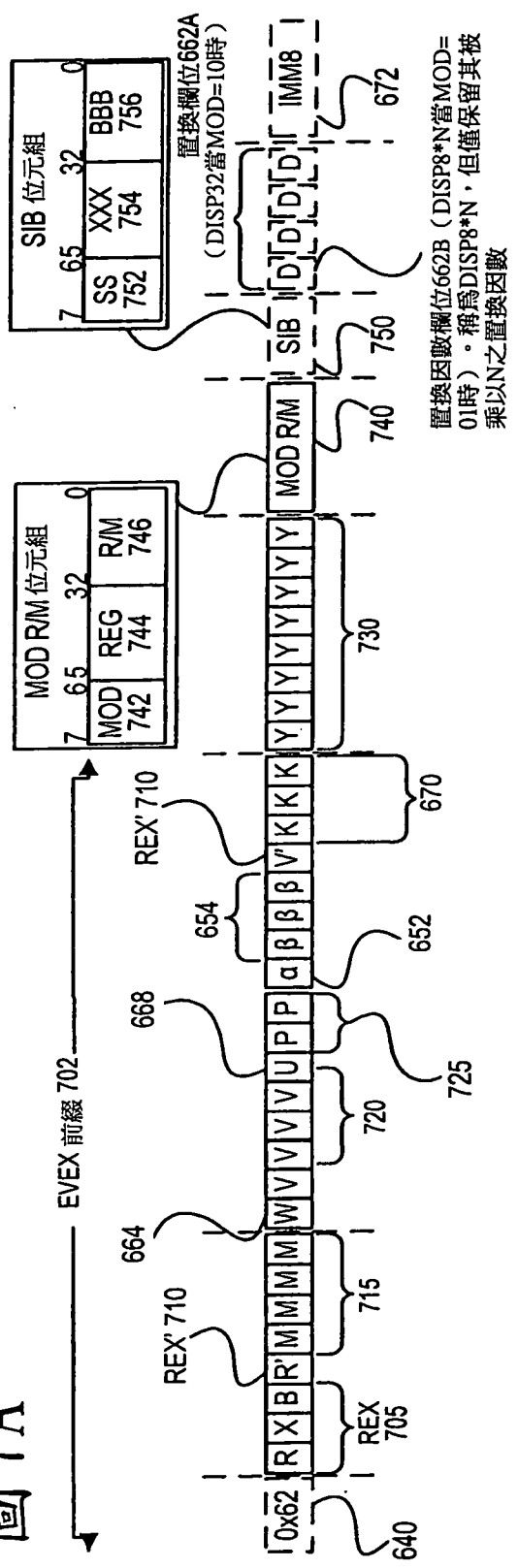


圖 7B

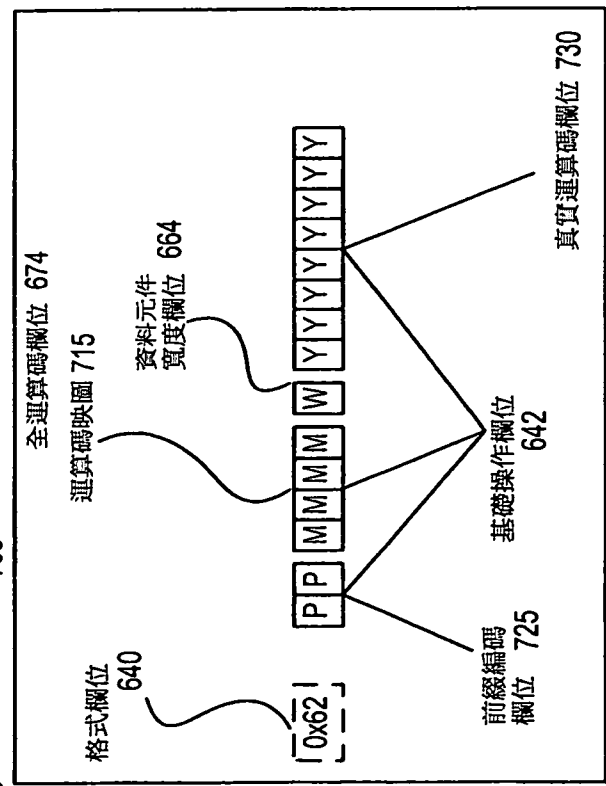


圖 7C

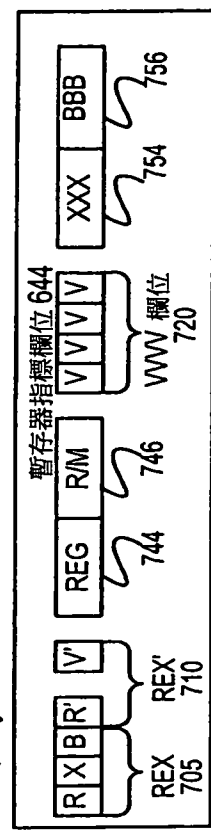




圖 7D

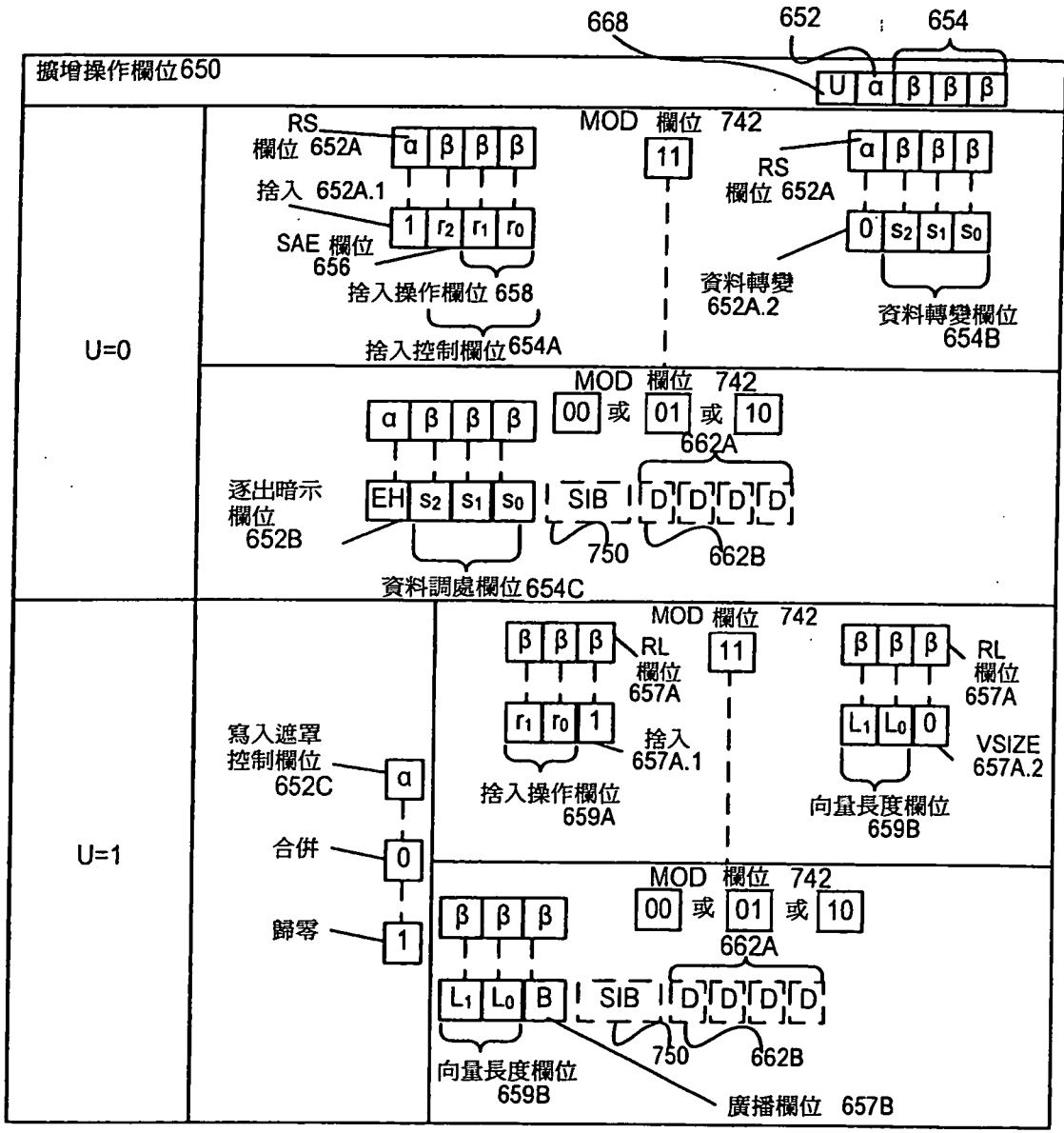


圖8

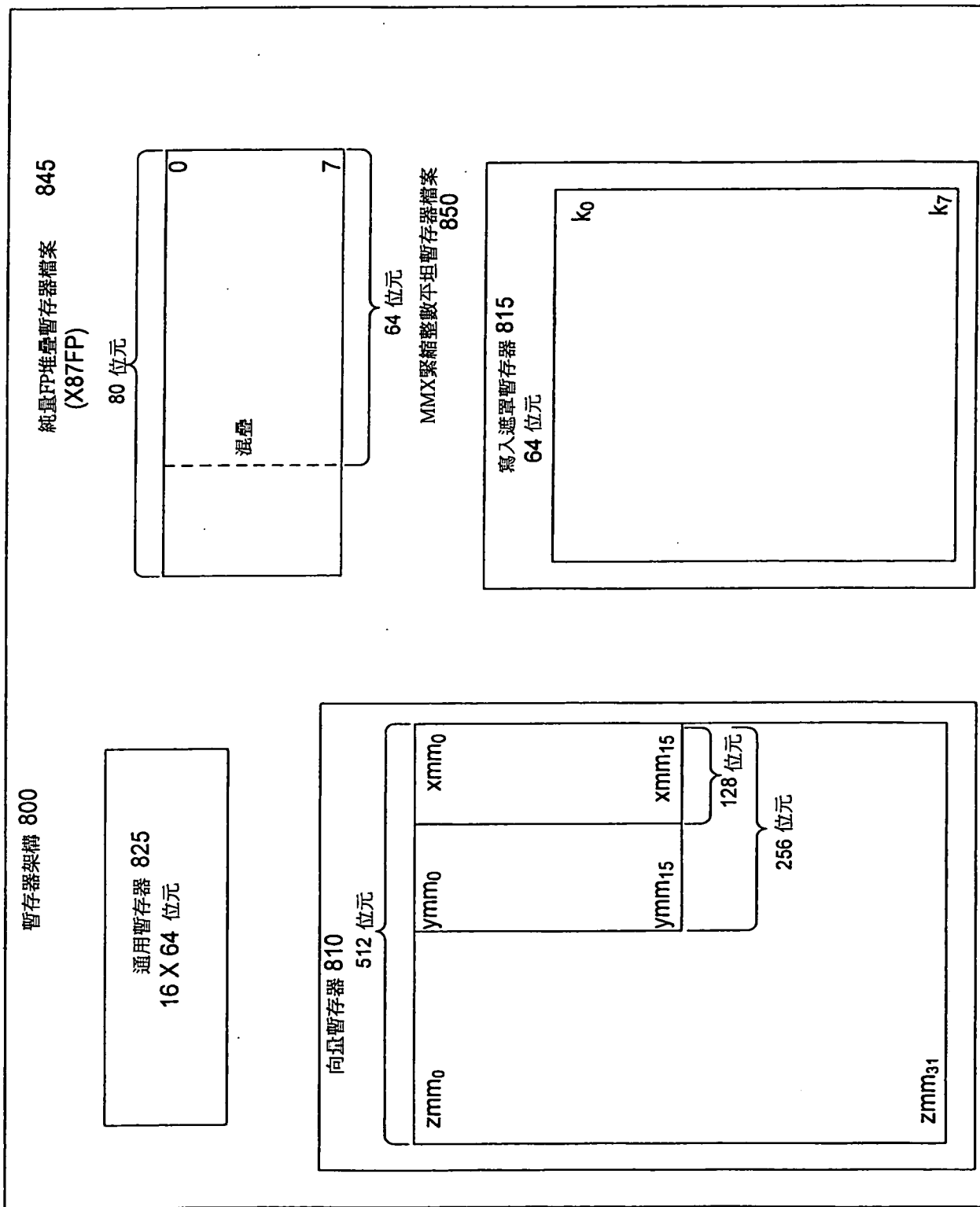
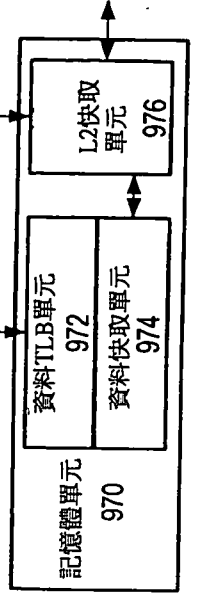
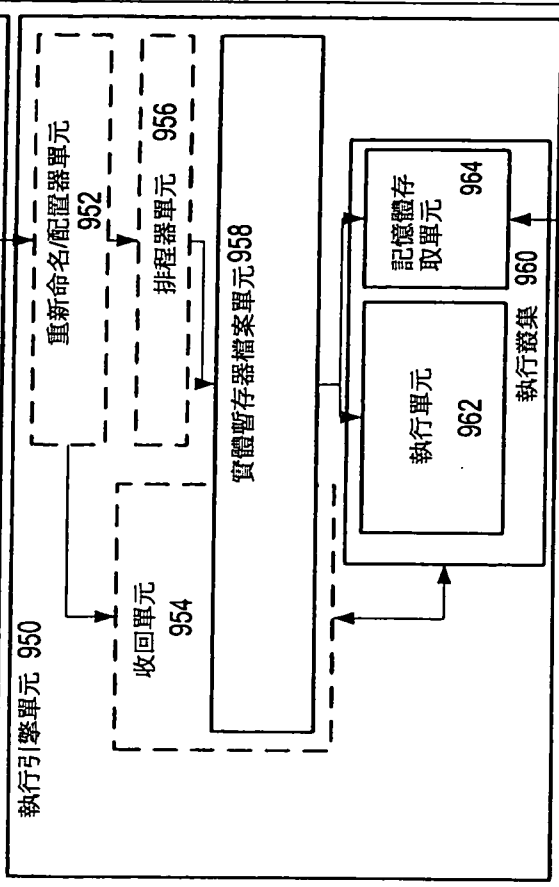
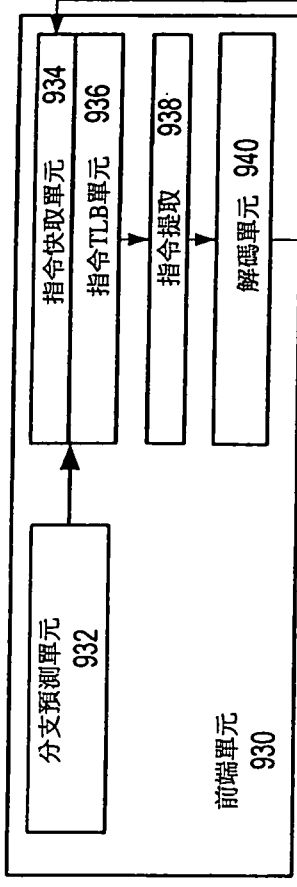
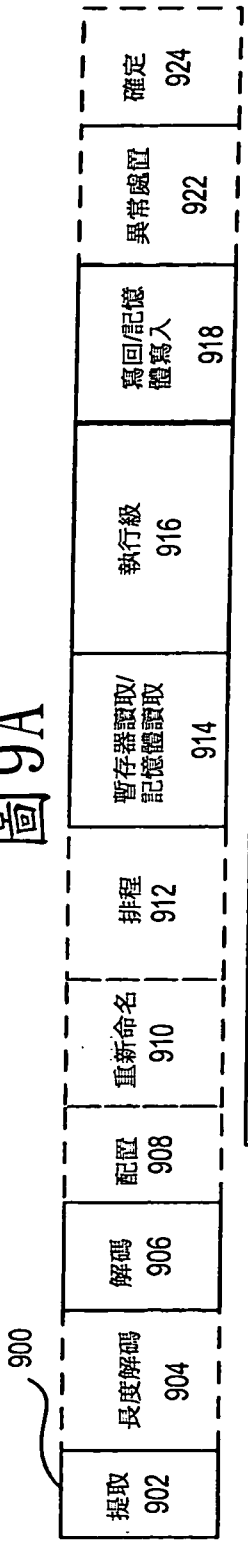


圖9A



990

圖9B

圖10A

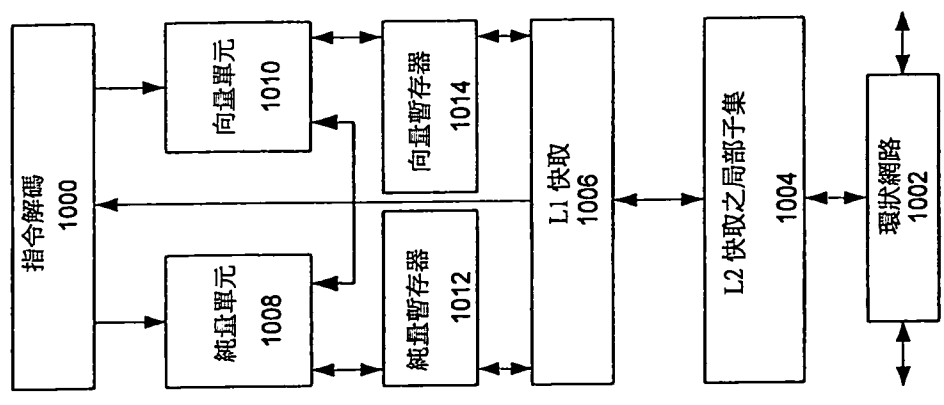
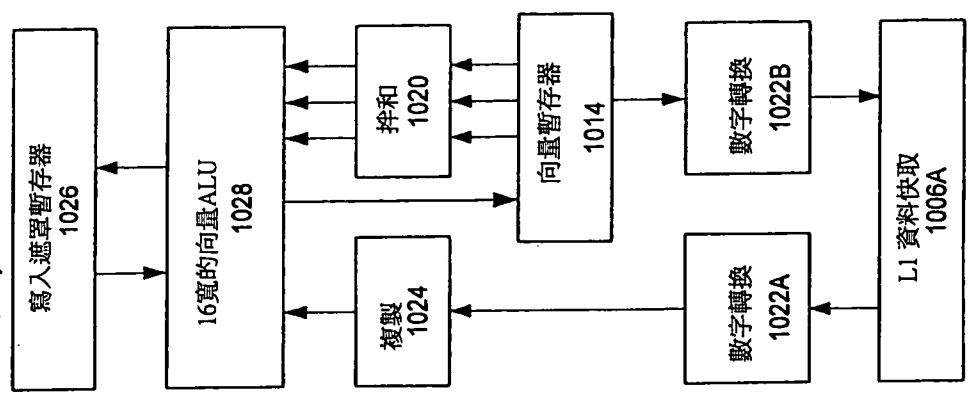


圖10B



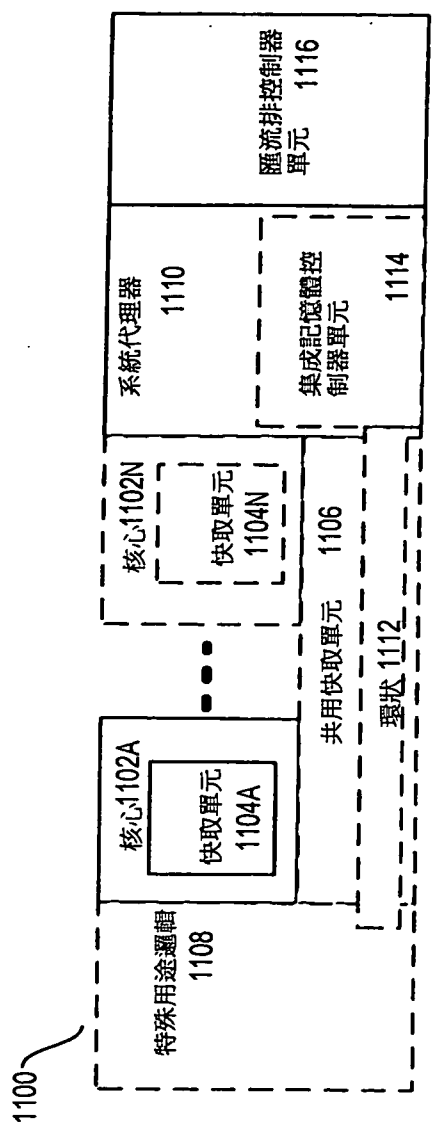


圖11

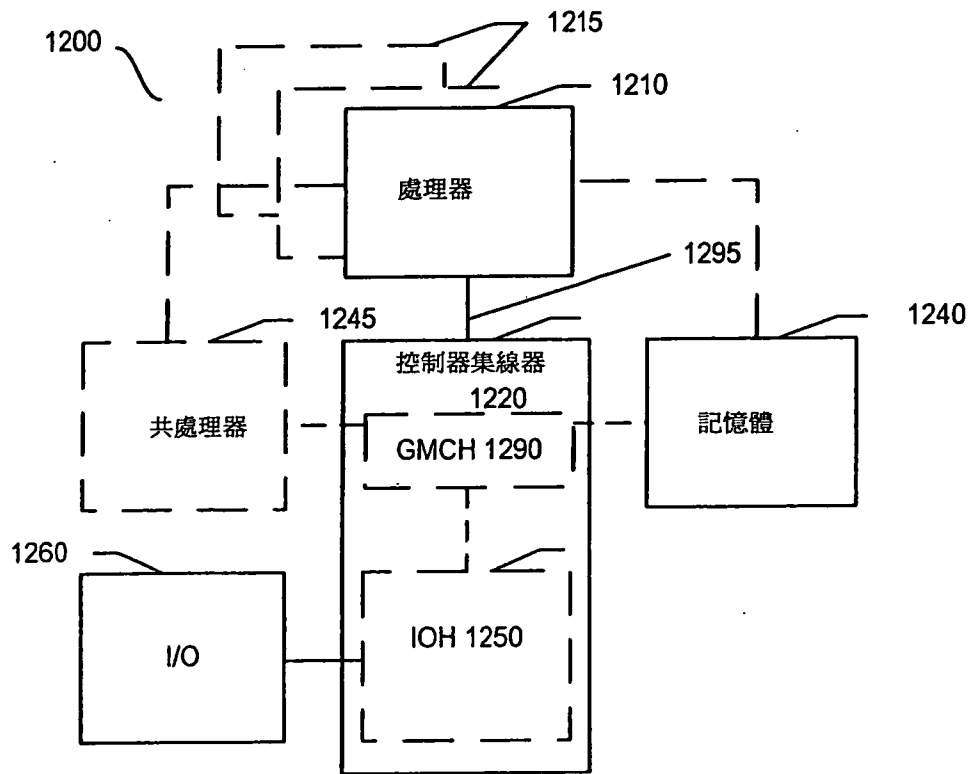


圖12

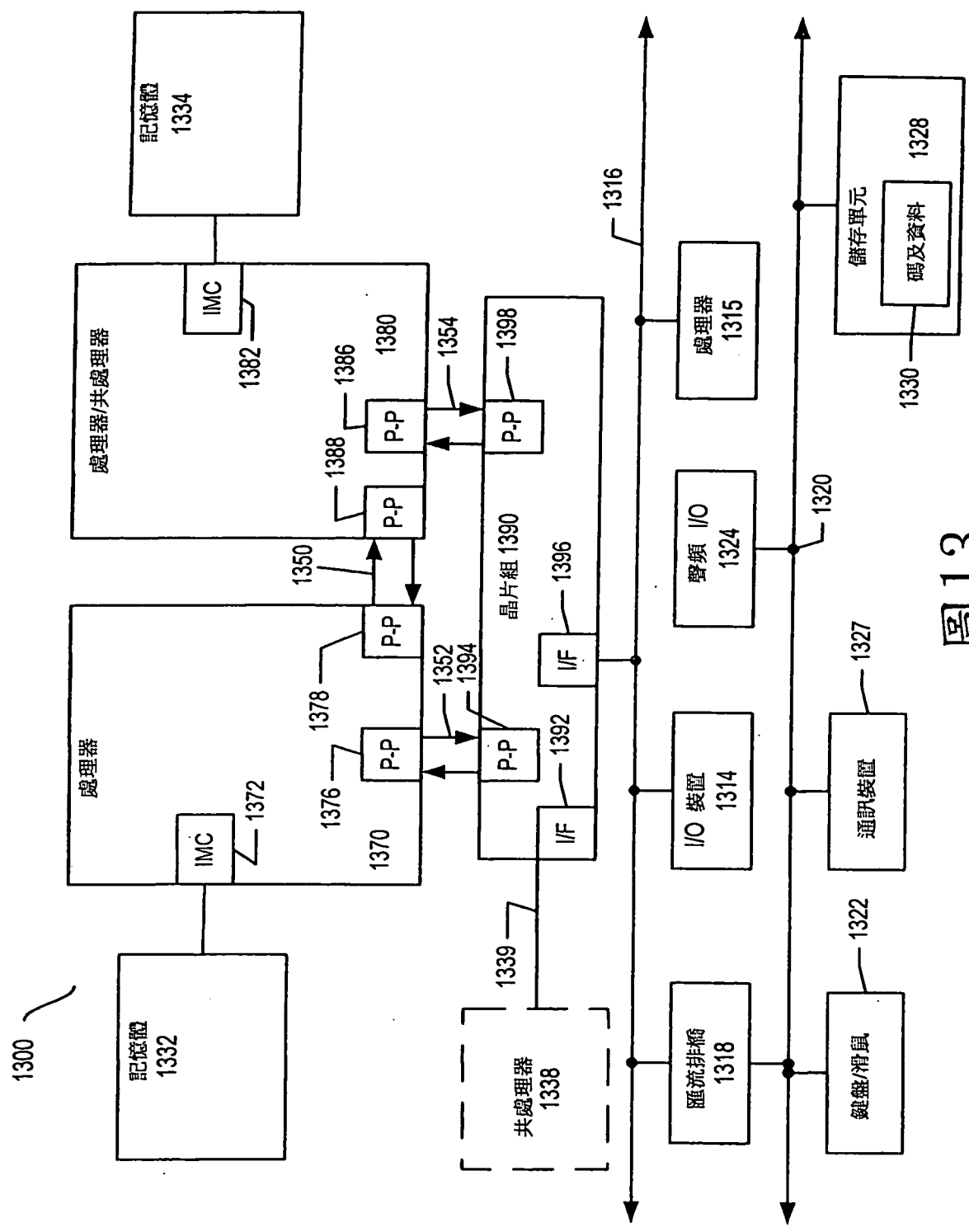


圖13

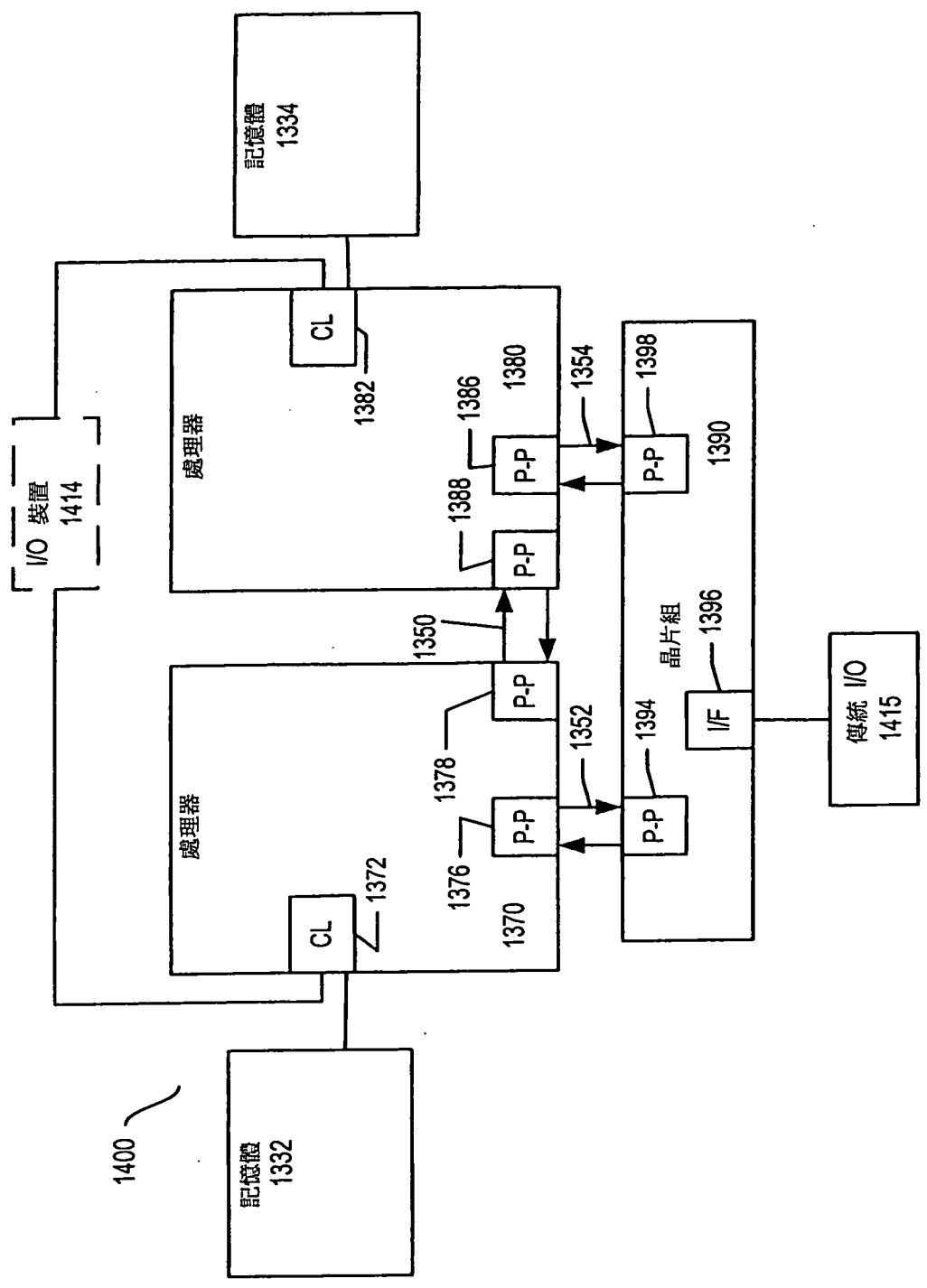


圖14



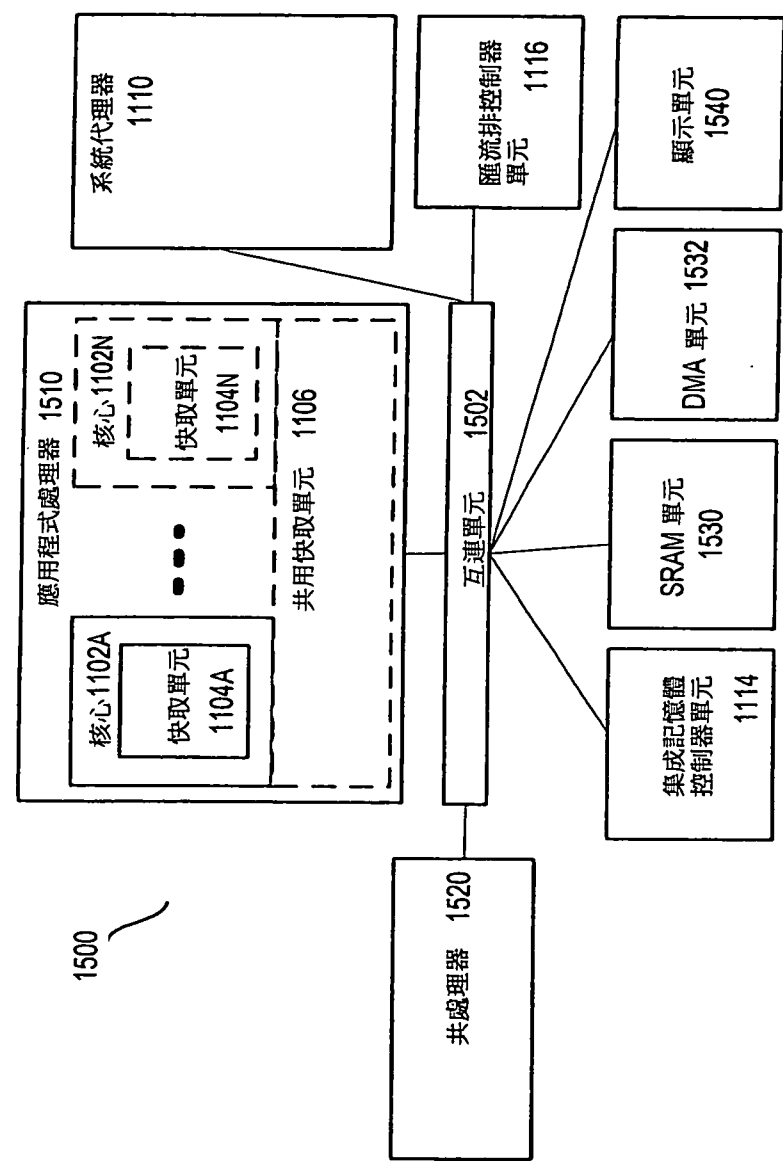


圖15

1500

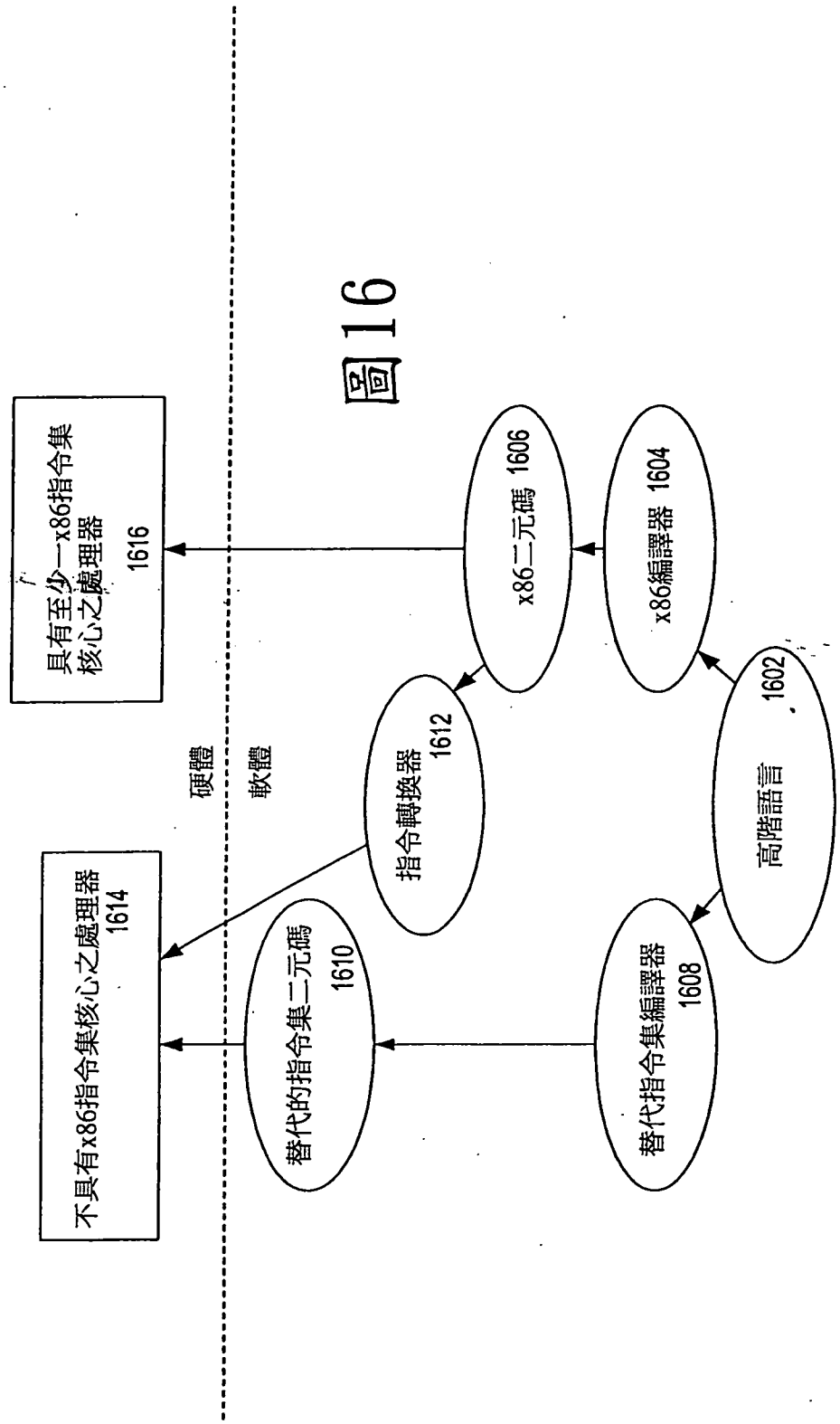


圖16