（此處由本局於收文時黏貼條碼）

842529

發明專利說明書

200413947

（本申請書格式、順序及粗體字，請勿任意更動，※記號部分請勿填寫）

※申請案號：92131106

※申請日期：92 年 11 月 06 日　　※IPC 分類：G06F 15/80 G06F 17/10

壹、發明名稱：

(中) 使用單一指令多重資料（ＳＩＭＤ）暫存器之小矩陣的有效乘法

(外) Efficient multiplication of small matrices using SIMD registers

貳、申請人：(共 1 人)

1.姓　名：（中）英特爾股份有限公司
（英）INTEL CORPORATION
代表人：（中）1.大衛　賽門
（英）1.SIMON, DAVID
地　址：（中）美國加州聖大克拉瑞密遜學院路二二〇〇號
（英）2200 Mission College Blvd., Santa Clara, CA 95052, USA
國籍：（中英）美國　U.S.A.

參、發明人：(共 1 人)

1.姓　名：（中）威廉　梅西二世
（英）MACY, JR., WILLIAM
地　址：（中）美國加州巴洛艾托梅爾維勒大道一五一號
（英）151 Melville Avenue, Palo Alto, CA 94301, U.S.A.

肆、聲明事項：

◎本案申請前已向下列國家（地區）申請專利□主張國際優先權：
【格式請依：受理國家（地區）；申請日；申請案號數　順序註記】
1.美國　；2002/12/20；10/327,445 ☑有主張優先權

（此處由本局於收文時黏貼條碼）

842529

發明專利說明書

200413947

（本申請書格式、順序及粗體字，請勿任意更動，※記號部分請勿填寫）

※申請案號：92131106

※申請日期：92 年 11 月 06 日　　※IPC 分類：G06F 15/80 G06F 17/10

壹、發明名稱：

(中) 使用單一指令多重資料（ＳＩＭＤ）暫存器之小矩陣的有效乘法

(外) Efficient multiplication of small matrices using SIMD registers

貳、申請人：(共 1 人)

1.姓　名：（中）英特爾股份有限公司
（英）INTEL CORPORATION
代表人：（中）1.大衛　賽門
（英）1.SIMON, DAVID
地　址：（中）美國加州聖大克拉瑞密遜學院路二二〇〇號
（英）2200 Mission College Blvd., Santa Clara, CA 95052, USA
國籍：（中英）美國　U.S.A.

參、發明人：(共 1 人)

1.姓　名：（中）威廉　梅西二世
（英）MACY, JR., WILLIAM
地　址：（中）美國加州巴洛艾托梅爾維勒大道一五一號
（英）151 Melville Avenue, Palo Alto, CA 94301, U.S.A.

肆、聲明事項：

◎本案申請前已向下列國家（地區）申請專利□主張國際優先權：
【格式請依：受理國家（地區）；申請日；申請案號數　順序註記】
1.美國　；2002/12/20；10/327,445 ☑有主張優先權

(1)

玖、發明說明

【發明所屬之技術領域】

本發明係有關矩陣算術。更明確言之，本發明提供使用 SIMD 暫存器之矩陣之有效乘法之例。

【先前技術】

算術操縱普通 mxn 矩陣爲一般資料處理工作。一 mxn 矩陣由 m 列及 n 行構成。被乘數矩陣 c 之大小爲 nxm，及乘數矩陣 a 爲 mxp。結果矩陣 b 爲 nxp。b 中之值由 c 之各行中之值乘 a 之各行中之值之乘積之和計算，使用 $b_{ij}=\Sigma_m C_{ik}*a_{kj}$，其中，第一下標指列及第二下標指行。故此，由 c 之列 i 及 a 之行 j 之內乘積計算 b 之列 i 及行 j 中之一元素之值。乘積 m*n*p 之總數及相加之總數爲 (m-1)*n*p。

爲最佳結果，已使用矩陣乘法實施來執行乘法，加法，及資料排序步驟，使用最少之指令數。由於 c 爲一係數矩陣及 a 爲一資料矩陣，故已發展各種技術，此等利用預儲存 c 元素之能力，其方式適於有效實施矩陣乘法。然而，儲存元素之此彈性不提供給矩陣 a 之資料。a 之資料通常依邏輯順序儲存，此不知任何資料處理演算法。

矩陣乘法使用於應用程式中，諸如坐標及色變換，造影演算法，及許多科學計算工作。矩陣乘法爲一計算密集之運算，此可由微處理器之單指令多資料(SIMD)暫存器協助執行，此支持普通 SIMD 矩陣乘法進行，使用 SIMD 指

(2)

令來安排資料，並執行矩陣乘法，遵循由矩陣乘法等式指示之計算順序：

$b_{ij}=\Sigma_m C_{ik}*a_{kj}$

其中：

$b(x)=c(x)*a(x)$

相當於

b_0	b_0	b_0	b_0
b_1	b_1	b_1	b_1
b_2	b_2	b_2	b_2
b_3	b_3	b_3	b_3

=

c_0	c_0	c_0	c_0
c_1	c_1	c_1	c_1
c_2	c_2	c_2	c_2
c_3	c_3	c_3	c_3

*

a_0	a_0	a_0	a_0
a_1	a_1	a_1	a_1
a_2	a_2	a_2	a_2
a_3	a_3	a_3	a_3

由被乘數矩陣 c 乘之各列乘乘數矩陣 a 之各行之內乘積(點乘積)計算結果矩陣 b 之元素。b 之第一元素爲：

$b_{00}=(c_{00}*a_{00})+(c_{01}*a_{10})+(c_{02}*a_{20})+(c_{03}*a_{30})$

此爲 c 之第一列及 a 之第一行之乘積及和。

其次

$b_{01}=(c_{00}*a_{01})+(c_{01}*a_{11})+(c_{02}*a_{21})+(c_{03}*a_{31})$

爲 c 之第一列及 a 之第二行之乘積及和。繼續計算，直至完成第一列之結果。使用 c 之次列計算 b 之次列，開始爲：

$b_{10}=(c_{10}*a_{00})+(c_{11}*a_{10})+(c_{12}*a_{20})+(c_{13}*a_{30})$。

由適當改變(XOR 取代加法)，同樣用於模組乘法及普通乘法。

使用 SIMD 指令之矩陣乘法之普通實施依乘數矩陣 a 之元素儲存於記憶體中之順序儲存其於 SIMD 暫存器中，

(3)

並依列順序儲存被乘數矩陣 c 之元素，以 c 之行數重複各列。a 之各元素依其儲存於記憶體中之順序儲存於暫存器中。例如，在 4 行矩陣中，c 之第一列之各元素重複 4 次，因爲 c 有 4 行。如 c 之大小小於 SIMD 暫存器，則來自 c 之其他列之元素亦可儲存於 SIMD 暫存器中。如 c 之大小大於 SIMD 暫存器，則需要額外之暫存器來儲存該列之資料。

使用 SIMD 暫存器中所儲存之資料之矩陣乘法由 c 之元素乘 a 中之元素開始 $-c_{00}*a_{00}$，$c_{01}*a_{10}...c_{03}*a_{30}$。其次，需計算同一暫存器中相鄰之每一列之乘積之和。如使用乘-加(MAC)指令，當計算乘法時，計算乘積之一些此等和。普通計算 b_{00}，隨後計算 b_{01}。矩陣 c 之次列載入於 c 值之暫存器中，以計算矩陣 b 之次列之元素。

在運算中，可需要模組乘積之精確重大之資料重排，俾此等可計算 b 之元素(在 Galois 場算術運算中，由例如 XOR 提供加法運算)。而且，如結果並未配入一暫存器中，則在其可儲存前，該等結果需在暫存器之間交換。二問題導致重大計算工作，此影響矩陣乘法進行之速度。

【實施方式】

圖 1 大體顯示一電腦系統 10，具有一處理器 12 及記憶系統 13(此可爲任何可存取記憶體，包含外部快取記憶體及內部 RAM，及/或處理器內之部份記憶體)，用以執行可由外部提供於軟體中之指令，作爲電腦程式產物並儲存

(4)

於資料儲存單元 18 中。

電腦系統 10 之處理器 12 亦支持內部記憶暫存器 14，包含單指令多資料(SIMD)暫存器 16。暫存器 14 在意義上並不限於特定型式之記憶電路。而是，一實施例之暫存器需能儲存並提供資料，並執行此處所述之功能。在一實施例，暫存器 14 包含多媒體暫存器，例如 SMID 暫存器 16 用以儲存多媒體資訊。在一實施例，多媒體暫存器各儲存高至 128 位元之包裝資料。多媒體暫存器可爲專用之多媒體暫存器，或用以儲存多媒體資訊及其他資訊之暫存器。在一實施例，當執行多媒體操作時，多媒體暫存器儲存多媒體資料，及當執行浮點操作時，儲存浮點資料。

本發明之電腦系統 10 可包含一或更多 I/O(輸入/輸出)裝置 15，包含顯示裝置，諸如監視器。I/O 裝置亦可包含一輸入裝置，諸如鍵盤，及一遊標控制器，諸如滑鼠，軌跡球，或軌跡墊。而且，I/O 裝置亦可包含網路連接器，俾該電腦系統 10 爲本地區網路(LAN)或寬地區網路(WAN)之一部份，I/O 裝置 15，聲音記錄及/或回放之裝置，諸如聲音數位化器連接至微音器，用以記錄語音辨認用之聲音輸入。I/O 裝置 15 亦可包含聲音數位化裝置，此可用以捕聲音影像，硬拷貝裝置，諸如印表機，及 CD-ROM 裝置。

在一實施例，可由資料儲存單元 18 讀出之電腦程式產物包含一機器或電腦可讀出之媒體，其上儲有指令，此可用以規劃(即訂定操作)一電腦(或其他電子裝置)，以依

(5)

本發明執行一程序。資料儲存單元 18 之電腦可讀出媒體包含，但不限於軟碟，光碟，小巧碟，僅讀記憶體(CD-ROM)，及磁光碟，僅讀記憶體(ROM)，隨機進出記憶體(RAM)，可抹消可程式僅讀記憶體(EPROM)，可電抹消可程式僅讀記憶體(EEPROM)，磁或光卡，快閃記憶體等。

故此，電腦可讀出之媒體包含任何型式之媒體/機器可讀出之媒體，適於儲存電子指令。而且，本發明亦可作爲電腦程式產物下載。如此，該程式可自遠處電腦(例如伺服器)轉移至申請之電腦(例如客戶)。程式可經由具體表現成載波之資料信號，或經由通訊鏈之其他傳播媒體(例如數據機，網路連接等)轉移。

電腦系統 10 可爲通用電腦，具有一處理器，帶有適當之暫存器結構，或可經構造供特定用途之用，或可爲埋置之應用程式。在一實施例，本發明之方法具體表現於機器可執行之指令，著眼於電腦系統之控制操作，且更明確言之，處理器及暫存器之操作。指令可用以使由指令規劃之通用或特殊用途處理器執行本發明之步驟。或且，本發明之步驟可由特定硬體組成件執行，此含有硬線邏輯用以執行該等步驟，或由程式電腦組成件及定製之硬體組成件之任何組合執行。

應明瞭知道本藝之人士使用各種術語及技術來說明通訊，議定，應用，實施，機構等。一種技術爲以演算法或數學表示式說明一技術之實施。即是，雖該技術可例如以在電腦執行程式來實施，但該技術可更宜且簡明地以公式

，演算法，或數學表示式來溝通。

故此，精於本藝之人士認識表示 A+B=C 之一方塊爲加函數，其在硬體及/或軟體中之實施爲取二輸入(A 及 B)，並產生一和輸出(C)。如此，應明瞭使用公式，演算法，或數學表示式作說明在至少硬體及/或軟體(諸如電腦系統，在此，可實作及實施本發明之技術，作爲一實施例)中具有物理實施例。

圖 2 顯示用以依本發明乘諸如圖 3 所示之一矩陣之程序。如顯示於圖 2，資料先由記錄及載入於記憶體(在本例中，標示如方塊 21 之暫存器)中加以組織，供有效矩陣乘法之用。被乘數矩陣 c 之每一對角線載入於不同之暫存器中。使用位置鄰近右行之矩陣之一拷貝，使具有一元素在並非底列之最右行之對角線延伸至次列中之元素。一對角線之次元素在次列中。對角線在暫存器中複製數次，其次數等於乘數矩陣 a 之行數。一對角線中之元素數等於 c 中之行數。乘數矩陣 a 之資料依行順序載入於暫存器中，順序資料儲存於記憶體中。在暫存器中之 a 之每一行中之每一相乘及相加元素之間移位一元素(方塊 22)。一行之最後元素移位或轉動至該行之前方。被乘數矩陣 c 之對角線由乘數矩陣 a 之行(可調整其長度)乘(方塊 23)，及其乘積加於乘積之和中，作爲結果矩陣 b 之行(方塊 24)。

如 a 之一行之元素數與 c 之一行之數不同，則調整來自 SIMD 暫存器中之 a 之一行之元素數，俾等於 c 之一行之元素數。決定選擇乘數矩陣 a 之何元素之一方法爲先相

(7)

互上下堆疊乘數矩陣 a 之拷貝，俾各行對齊，且一拷貝之頂列在底列及另一拷貝下方。此有效延伸每一行。由於自延伸行所取之元素數等於被乘數矩陣 c 之一對角線中之元素數。在每一相乘及相加運算後，由向下移位延伸之行一元素，選擇元素用於其次相乘及相加運算。如一被乘數對角線之長度大於一乘數行，則選擇一行中之相等值，且如被乘數對角線之長度小於乘數行，則不選擇一行中之所有值。

雖以上實例使用內部處理器暫存器，但應明瞭並非恆需載入內部處理器暫存器，以執行 SIMD 操作。用於相乘或其他之運算元可儲存於記憶體中，而非先載入於暫存器中。一些構造，諸如 RISC 構造先載入暫存器，但 Intel 構造可具有運算元在記憶體中。使用暫存器及跡憶器運算元之比較爲 pmaddwdxmmo,xmm1andpmaddwdxmmo,[eax] 如儲存於暫存器 eax 位址中之資料與 xmm1 中之資料相同，則在 xmmo 中產生相同結果。如暫存器中之代碼用完且記憶體進出快速，則需要使用記憶體運算元。

圖 3 顯示依有關圖 2 一般討論之程序之模組乘法 30。在本例中，模組乘法爲一 Galois 場算術，在此，使用 XOR 於相加值，而無進位(例如，二進位相加而無進位，故 1+1=0，0+0=0，0+1=1，及 1+0=1，且普通由 XOR 計算結果)。如顯示於圖 3。決定正方矩陣 b(x)=c(x)xa(x)之乘法 30。圖 4 顯示決定暫存器資料載入型樣 40，用於圖 3 所示矩陣之乘法。如見之於圖 4 之一暫存器排序設計 40

(8)

中，用於次步驟之暫存器中之資料爲粗體式。實線指示複製矩陣之界線。在第一步驟，a之各行由c之對角線乘。在第二步驟，a之各行移位，並由c之次對角線乘，如箭頭所示。

圖5顯示由圖4所示之移位造成暫存器中之資料之順序50。如有關圖5之時間步驟(A)所見，暫存器依儲存於記憶體中之順序，保持c之主對角線及矩陣a之資料。在圖5之時間步驟(B)，由使用一位元組穿梭操作，由轉動元素實施移位各行。注意a中之各行可上移位，且c中之選擇對角線可選擇至左方而非右方。

圖6另顯示用以乘4x4矩陣a及c之運算60。每一時間步驟之資料依以上有關圖4及5所述排序。在每一時間步驟C，D，E，及F，計算a及c之模組乘積。乘積由XOR加於其他步驟之乘積。

以下假碼片段提供矩陣乘法之一實例實施。

(1)LDOADR3,MEMORY；c矩陣對角線1

(2)LDOADR4,MEMORY；c矩陣對角線2

(3)LDOADR5,MEMORY；c矩陣對角線3

(4)LDOADR6,MEMORY；c矩陣對角線4

(5)LDOADR7,MEMORY；資料穿梭型樣

(6)LDOADR0,MEMORY；自記憶體載入a資料(第一型樣)

(7)MOVER1,RO；拷貝第一資料型樣

(8)MODMULR0,R3；由對角線1(主對角線)乘a資料

(9)

(9)SHUFFLER1,R7；產生第二 a 資料型樣轉動行

(10)MOVER2,R1；拷貝第二 a 資料型樣

(11)MODMULR1,R4；由對角線 2 乘第二 a 資料型樣

(12)XORR0,R1；加第二型樣於第一

(13)SHUFFLER2,R7；產生第三 a 資料型樣轉動行

(14)MOVER1,R2；拷貝第三 a 資料型樣

(15)MODMULR2,R5；由對角線 3 乘第三 a 資料型樣

(16)XORR0,R2；加第三型樣

(17)SHUFFLER1,R6；產生第四 a 資料型樣轉動行

(18)MODMULR1,R6；由對角線 4 乘第四資料型樣

(19)XORR0,R1；加第四型樣

(20)STOREMEMORY,R0；儲存輸出矩陣

指令 9 至 12 表示本方法之基本操作。乘數 a 矩陣之各行在指令 9 中轉動。其結果在指令 10 中拷貝，因爲此由指令 11 中之乘法覆寫，及乘積在指令 12 中加於乘積之和中。

非正矩陣亦可接受本發明之程序之一實施例。例如，考慮圖 7 之矩陣乘法 70。在此，被乘數矩陣 c 之對角線中之元素數不等於乘數矩陣 a 之一行中之元素數，及被乘數矩陣 c 之對角線大於乘數矩陣 a 之行。在本例中，3x2 矩陣 c 乘 2x4 矩陣 a 之模組乘法。圖 8 說明本例中用以選擇及排序資料於 SIMD 中之方法。c 之第一對角線爲 c_{00}，c_{110}，c_{20}。此對角線由 a 之延伸行之首 3 值乘。由於 a 之行長度僅爲 2，故 a 矩陣依順序 80 相互堆疊，如顯示於

圖 8，以有效延伸行之長度。觀察此之另一方法爲，一旦到達一行之末端時，此捲回或轉回至第一值。圖 9 顯示 c 之第一對角線及 a 之延伸行之值之資料排列。注意在右方之 a 之首 3 值爲 a_{00}，a_{10}，a_{00}，故 a_{00} 重複。c 之次對角線爲 c_{01}，c_{10}，c_{21}，及 a 之次行爲 a_{10}，a_{00}，a_{10}，此由每一延伸行中向下移位一元素選擇，如顯示於圖 8。圖 9 另顯示用以乘矩陣 a 及 c 之運算。每一時間步驟之資料順序如以上有關圖 7 及 8 所述。在每一時間步驟，計算 a 及 c 之模組乘積。各乘責由 XOR 加於其他步驟之乘積中。

圖 10 顯示模組乘法 100，具有被乘數矩陣 c 對角線短於乘數矩陣 a，使用 2x3 矩陣 c 及 3x4 矩陣 a。如顯示於圖 11，順序選擇 110 設定 c 之第一對角線爲 c_{00} 及 c_{11}。此對角線由 a 之延伸行之首二值 a_{00} 及 a_{10} 乘。a 之行長度爲 3，但僅選擇行之二值。圖 12 顯示暫存器中各值之資料排列 120。有三對暫存器，具有來自矩陣 a 及 c 之值，此等相乘一起，因爲矩陣 c 具有三對角線。僅 a 之第一行之首 2 值 a_{00} 及 a_{10} 儲存於第一暫存器中。在次對暫存器中，c 之對角線爲 c_{01} 及 c_{12}，及由向下移位選擇 a 中之次值。例如，來自第一行之值爲 a_{10} 及 a_{20}。第三對暫存器保持第三對角線及 a 之向下移位之行之次值。在此情形，來自第一行之值爲 a_{20} 及 a_{00}。

如所明瞭，圖 3-12 之以上說明敘述無需相乘/累計(MAC)指令之算術運算。代之者，說明 Galois 場算術，使用模組乘法及 XOR 於加法。如被乘數之一列及乘數之一

行之元素之乘積由與原矩陣元素相同之資料型式表示，則僅普通算術及 Galois 場算術間之不同爲用於相加及相乘之方法。所有型樣保持相同。如結果所需之資料型式在大小上大於原資料者，則矩陣元素之資料型式在矩陣相乘之前增加(通常大小加倍)。在此情形，儲存恆定之被乘數矩陣資料，成爲較大資料型式。例如，儲存位元組大小係數，成爲 16 位元整數。乘數矩陣之資料型式在圖 3-12 所示之計算前改變。通常使用 SIMD 解包操作，以改變資料型式。此增加是時所需之暫存器數，但否則，圖 3-12 所述之操作在 Galois 場或普通算術方面不變。

如可用 MAC 指令，可如有關以下圖 13-15 所示進行矩陣乘法。雖 MAC 指令可用於任何形式之算術(包括 Galois 場算術)，但在普通固定點算術之情形，一 MAC 電腦 2 產生，相加此等乘積，且通常寫入結果，成爲原被乘數及乘數之大小之二倍之資料型式(普通位元組至 16 位元字及 16 位元字至雙倍 32 位元字)。在 Galois 場算術之情形，MAC 電腦 2 使用模組乘法產生，使用 XOR 運算加乘積，及寫入同資料型式之結果。代表 Galois 場算術之和或乘積所需之數元數與代表原資料所需之位元數相同。普通算術用之 MAC 大部份見之於所有 SIMD 指令集(即在 Intel 架構指令集中之 madd)。故此，圖 13 顯示具有正矩陣之乘法 130，並使用適當之 MAC 指令。如顯示於圖 14，排序 140 以體式指示連續步驟用之暫存器中之資料。實線指示複製矩陣之界線。注意在正矩陣乘法中，元素爲二

值，及每一移位爲二值。在正乘法情形中，在矩陣 c 之一對角線中之值之數爲矩陣 a 之一行之二倍，如顯示於圖 14(本實例中排序 8 值)。複製 a 矩陣之每一行，如顯示於圖 15a 及 b 之暫存器排序 150 中。故此，a 矩陣之首二行保持於一暫存器中，及次二行保持於另一暫存器中。正矩陣乘法之資料排序方法與模組乘法相同，唯在正矩陣情形，各元素爲二值。移位二值至次步驟之資料順序，並複製各乘數行。乘-加運算施加於 a 及 c 中之相鄰值。此運算乘 a 及 c 中之值，並相加相鄰之乘積。乘-加結果儲存於原資料大小之二倍之空間中。例如，在步驟(1)中，madd 運算計算 a_{00} 及 c_{00} 之乘積及 a_{10}，及 c_{01} 之乘積，並相加二乘積。同樣，，在步驟(2)中，madd 運算計算 a_{20} 及 c_{02} 之乘積及 a_{30} 及 c_{03} 之乘積，並相加二乘積。madd 運算之結果相加，以提供矩陣乘法之結果 b_{00}。

使用 16 位元字及 128 位元暫存器之正矩陣乘法之假碼顯示如下：

(1)LOADR5,MEMORY；係數對角線 1

(2)LOADR5,MEMORY；係數對角線 2

(3)LOADR5,MEMORY；資料穿梭型樣

(4)LOADR5,MEMORY；載入記憶體之資料(第一型樣)

(5)MOVER2,RO；拷貝第一資料型樣

(6)UNPACKLDQR0,R0；複製資料行 1 及 2

(7)MOVER1,R0；拷貝行 1 及 2

(8)MADDRO,R5；乘累計 1 及 2

(13)

(9)SHUFFLER1,R7；產生第二資料型樣

(10)MADDR1,R6；乘累計型樣 2 行 1 及 2

(11)ADDWRO,R1；結果行 1 及 2

(12)STOREMEMORY,RO；儲存結果行 1 及 2

(13)UNPACKHDOR2,R2；複製行 3 及 4

(14)MOVER3,R2；拷貝行 3 及 4

(15)MADDR2,R5；乘累計行 3 及 4

(16)SHUFFLER3,R7；產生第二資料型樣

(17)MADDR3,R6；乘累計型樣 2 行 3 及 4

(18)ADDWR2,R3；結果行 3 及 4

(19)STOREMEMORY,R2；儲存結果行 3 及 4 由二乘-加運算，一穿梭，及乘-加結果之一加法產生每一結果。結果爲 16 位元，故 16 結果需要二 128 位元暫存器。

雖本發明特別可用於由 SIMD 指令實施位元組資料之矩陣乘法，但本發明並不限於此乘法。可使用較大資料型式，僅需減少一暫存器中可儲存之元素數，且較大之矩陣具有較多之元素需儲存。如被乘數矩陣 c 之對角線，或乘數矩陣 a 之行並不配合於一 SIMD 暫存器中，則此等可延伸至額外暫存器。在使用較大暫存器之一些情形，一行中資料之轉動可需要暫存器間交換資料。

如所明瞭，說明書中所提"一實施例"，"一些實施例"，"或其他實施例"意爲有關實施例中所述一特定特色，結構，或特性包含於至少一些實施例中，但並非必需包含於本發明之所有實施例中。各種顯示"一實施例"，或"一些

實施例"並非需均指同一實施例。

如說明書說明"可"或"能"包含一組成件，特色，結構，或特性，此並非必需包含該特定組成件，特色，結構，或特性。如說明書或申請專利提及"一"元素，此並非意爲僅一個該元素。如說明書或申請專利提及"一額外"元素，此並不排除一個以上之額外元素。

受益於本說明之精於本藝之人士可明瞭在本發明範圍內可作與以上說明及附圖不同之許多其他改變。故此，包含其任何增補之以下申請專利界定本發明之範圍。

【圖式簡單說明】

自以下本發明之實施例之詳細說明及附圖，可更完全明瞭本發明，然而，此不應限制本發明於所述之特定實施例，而是僅供說明及瞭解之用。

圖 1 概要顯示支持 SIMD 暫存器之一計算系統；

圖 2 爲用以記錄供有效矩陣乘法用之資料之程序；

圖 3 顯示通類 4x4 模組矩陣乘法；

圖 4 顯示記錄供暫存器基礎之乘法用之資料；

圖 5 顯示依圖 4 記錄後之暫存器；

圖 6 顯示依圖 4 及 5 記錄後之矩陣乘法；

圖 7 顯示模組矩陣乘法，在此，被乘數矩陣 c 之一對角線中之元素數不等於乘數矩陣之一行中之元素數；

圖 8 顯示記錄供暫存器基礎之乘法用之資料；

圖 9 顯示在依圖 7 及 8 記錄後之矩陣乘法；

(15)

圖 10 顯示模組矩陣乘法，在此，被乘數矩陣 c 對角線小於乘數 a，使用 2x3 矩陣 c 及 3x4 矩陣 a；

圖 11 顯示記錄供暫存器基礎之乘法用之資料；

圖 12 顯示在依據圖 10 及 11 記錄後之矩陣乘法；

圖 13 顯示具有正矩陣之模組矩陣乘法；

圖 14 顯示記錄供暫存器基礎之乘法用之資料；及

圖 15 顯示在依據圖 13 及 14 記錄後之矩陣乘法。

元件對照表

10：電腦系統

12：處理器

13：記憶系統

14：內部記憶暫存器

15：I/O 裝置

16：單指令多資料暫存器

18：資料儲存單元

伍、中文發明摘要

發明之名稱：使用單一指令多重資料（SIMD）暫存器之小矩陣的有效乘法

說明一種矩陣乘法之例，其減少在 SIMD 處理器上的計算時間。此矩陣乘法需要將要被乘數矩陣 c 之每一對角線載入於處理器之不同的暫存器中，並依行順序乘數矩陣 a 載入於至少一暫存器中。暫存器中之乘數矩陣 a 之每一行中的乘法及加法元素藉由移位一元素來作選擇性移位，一行的最後一個元素移位至該行的前方。被乘數矩陣 c 之對角線被乘以乘數矩陣 a 的諸各行，且爲結果矩陣的諸行其乘積加到乘積的和中。

陸、英文發明摘要

發明之名稱：

EFFICIENT MULTIPLICATION OF SMALL MATRICES USING SIMD REGISTERS

An example of a matrix multiplication method that reduces calculation times on SIMD processors is described. The matrix multiplication requires loading each diagonal of the multiplicand matrix **c** into a different register of a processor, and loading a multiplier matrix **a** into at least one register in column order. Multiplication and addition elements in each column of multiplier matrix **a** in the register are selectively shifted to by shifting one element, with the last element of a column shifted to the front of the column. Diagonals of the multiplicand **c** matrix are multiplied by columns of the multiplier **a** matrix, with their product being added to the sum of products for columns of a result matrix.

(1)

拾、申請專利範圍

1.一種矩陣乘法方法，包含：

將被乘數矩陣 c 之每一對角線載入於處理器可存取記憶體中，

依行順序而將乘數矩陣 a 載入於處理器可存取記憶體中，

藉由移位一元素而將乘數矩陣 a 之每一行的元素移位入暫存器中，且一行的最後一個元素移位至該行的前方，及藉由將乘數矩陣 a 的諸行乘上被乘數矩陣 c 的對角線，且爲結果矩陣之各行將其乘積加到乘積的和中。

2.如申請專利範圍第 1 項所述之方法，其中，處理器可存取之記憶體爲一 SIMD 暫存器。

3.如申請專利範圍第 2 項所述之方法，另包含載入一對角線於處理器之多個 SIMD 暫存器中。

4.如申請專利範圍第 1 項所述之方法，其中，乘數 a 矩陣在與被乘數 c 矩陣之對角線相乘之前，藉由相互上下堆疊乘數矩陣 a 之拷貝來調整其長度，俾諸行對齊，且一拷貝之頂列係在一底列及任何其他拷貝之下，以延伸每一行。

5.如申請專利範圍第 1 項所述之方法，其中，被乘數矩陣 c 之對角線較乘數矩陣 a 之行短。

6.如申請專利範圍第 1 項所述之方法，其中，被乘數矩陣 c 之對角線較乘數矩陣 a 之行長。

7.如申請專利範圍第 1 項所述之方法，其中，移位該

(2)

等元素另包含依預定順序，以 c 之一對角線乘上 a 之各行；及移位並以 c 之次一對角線乘上 a 之各行。

8.如申請專利範圍第 1 項所述之方法，其中，移位該等元素另包含使用一位元組穿梭操作來轉動諸元素。

9.如申請專利範圍第 1 項所述之方法，其中，各元素爲一位元組。

10.如申請專利範圍第 1 項所述之方法，其中，乘諸對角線另包含 MAC 運算之應用。

11.一種包含一儲存媒體之物件，該儲存媒體具有指令儲存於其上，該物件當由一機器予以執行時將導致：

將被乘數矩陣 c 之每一對角線載入於處理器可存取記憶體中，

依行順序而將乘數矩陣 a 載入於處理器可存取記憶體中，

藉由移位一元素而將乘數矩陣 a 的每一行的元素移位入暫存器中，且一行的最後一個元素移位至該行的前方，及

藉由將乘數 a 矩陣的諸行乘上被乘數 c 矩陣的對角線且爲結果矩陣之各行將其乘積加到乘積的和中。

12.如申請專利範圍第 11 項所述之包含具有指令儲存於其上之儲存媒體的物件，其中，處理器可存取之記憶體爲一 SIMD 暫存器。

13.如申請專利範圍第 12 項所述之包含具有指令儲存於其上之儲存媒體的物件，其中，載入一對角線於處理器

(3)

之多個 SIMD 暫存器中。

14.如申請專利範圍第 11 項所述之包含具有指令儲存於其上之儲存媒體的物件，其中，乘數 a 矩陣在與被乘數 c 矩陣之對角線相乘之前，藉由相互上下堆疊乘數矩陣 a 之拷貝來調整其長度，俾諸行對齊，且一拷貝之頂列係在一底列及任何其他拷貝之下，以延伸每一行。

15.如申請專利範圍第 11 項所述之包含具有指令儲存於其上之儲存媒體的物件，其中，被乘數矩陣 c 之對角線較乘數 a 矩陣之行短。

16.如申請專利範圍第 11 項所述之包含具有指令儲存於其上之儲存媒體的物件，其中，被乘數矩陣 c 之對角線較乘數 a 矩陣之行長。

17.如申請專利範圍第 11 項所述之包含具有指令儲存於其上之儲存媒體的物件，其中，移位乘法及加法元素另包含依預定順序，以 c 之一對角線乘上 a 之各行；及移位並以 c 之次一對角線乘上 a 之各行。

18.如申請專利範圍第 11 項所述之包含具有指令儲存於其上之儲存媒體的物件，其中，移位乘法及加法之元素另包含使用一位元組穿梭操作來轉動諸元素。

19.如申請專利範圍第 11 項所述之包含具有指令儲存於其上之儲存媒體的物件，其中，乘諸對角線另包含 MAC 運算之應用。

20.如申請專利範圍第 11 項所述之包含具有指令儲存於其上之儲存媒體的物件，其中，各元素爲一位元組。

(4)

21.一種矩陣乘法系統，包含：

一處理器，具有暫存器，其將被乘數矩陣 c 之每一對角線載入於處理器可存取記憶體中，且依行順序而將乘數矩陣 a 載入於處理器可存取記憶體中，及

控制邏輯，藉由移位一元素而將乘數矩陣 a 之每一行之相乘及相加元素移位入暫存器中，且一行的最後一個元素移位至該行的前方，及藉由乘數矩陣 a 之各行乘上被乘數矩陣 c 之各對角線，且爲結果矩陣之各行將其乘積加到乘積的和中。

22.如申請專利範圍第 21 項所述之系統，其中，處理器可存取記憶體爲一 SIMD 暫存器。

23.如申請專利範圍第 22 項所述之系統，另包含載入一對角線於處理器之多個 SIMD 暫存器中。

24.如申請專利範圍第 21 項所述之系統，其中，乘數矩陣 a 在與被乘數矩陣 c 之對角線相乘之前，藉由相互上下堆疊乘數矩陣 a 之拷貝來調整其長度，俾諸行對齊，且一拷貝之頂列係在一底列係及任何其他拷貝之下，以延伸每一行。

25.如申請專利範圍第 21 項所述之系統，其中，被乘數矩陣 c 之對角線較乘數 a 矩陣之行短。

26.如申請專利範圍第 21 項所述之系統，其中，被乘數矩陣 c 之對角線較乘數 a 矩陣之行長。

27.如申請專利範圍第 21 項所述之系統，其中，移位乘法及加法元素之控制邏輯另包含依預定順序，以 c 之一

(5)

對角線乘上 a 之各行；及移位並以 c 之次一對角線乘上 a 之各行。

28.如申請專利範圍第 21 項所述之系統，其中，移位乘法及加法元素之控制邏輯另包含使用一位元組穿梭操作來轉動各元素。

29.如申請專利範圍第 21 項所述之系統，其中，各元素爲一位元組。

30.如申請專利範圍第21項所述之系統，其中，乘諸對角線另包含MAC運算之應用。

圖1

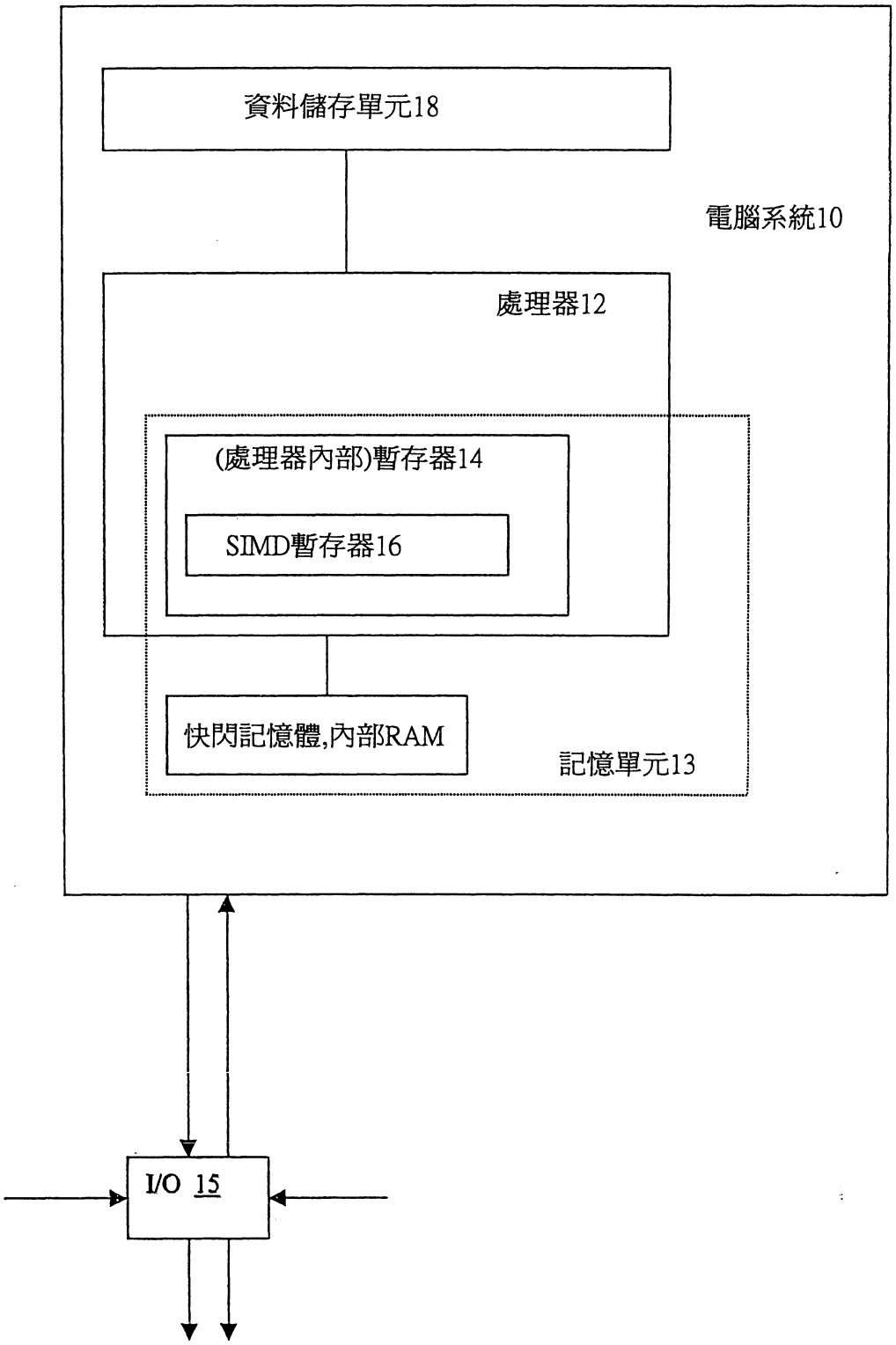

資料儲存單元18
電腦系統10
處理器12
(處理器內部)暫存器14
SIMD暫存器16
快閃記憶體,內部RAM
記憶單元13
I/O 15

圖2

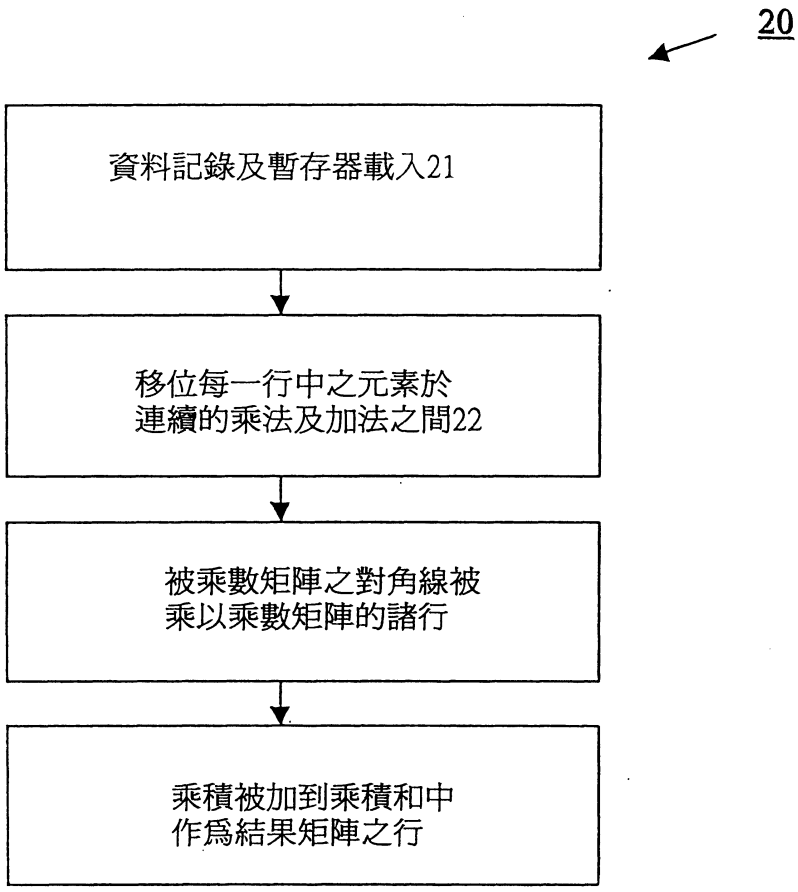

20
資料記錄及暫存器載入21
移位每一行中之元素於
連續的乘法及加法之間22
被乘數矩陣之對角線被
乘以乘數矩陣的諸行
乘積被加到乘積和中
作爲結果矩陣之行

圖3

30

$$b(x) = c(x) \otimes a(x)$$

$$\begin{bmatrix} b_{00} & b_{01} & b_{02} & b_{03} \\ b_{10} & b_{11} & b_{12} & b_{13} \\ b_{20} & b_{21} & b_{22} & b_{23} \\ b_{30} & b_{31} & b_{32} & b_{33} \end{bmatrix} = \begin{bmatrix} c_{00} & c_{01} & c_{02} & c_{03} \\ c_{10} & c_{11} & c_{12} & c_{13} \\ c_{20} & c_{21} & c_{22} & c_{23} \\ c_{30} & c_{31} & c_{32} & c_{33} \end{bmatrix} \otimes \begin{bmatrix} a_{00} & a_{01} & a_{02} & a_{03} \\ a_{10} & a_{11} & a_{12} & a_{13} \\ a_{20} & a_{21} & a_{22} & a_{23} \\ a_{30} & a_{31} & a_{32} & a_{33} \end{bmatrix}$$

圖4

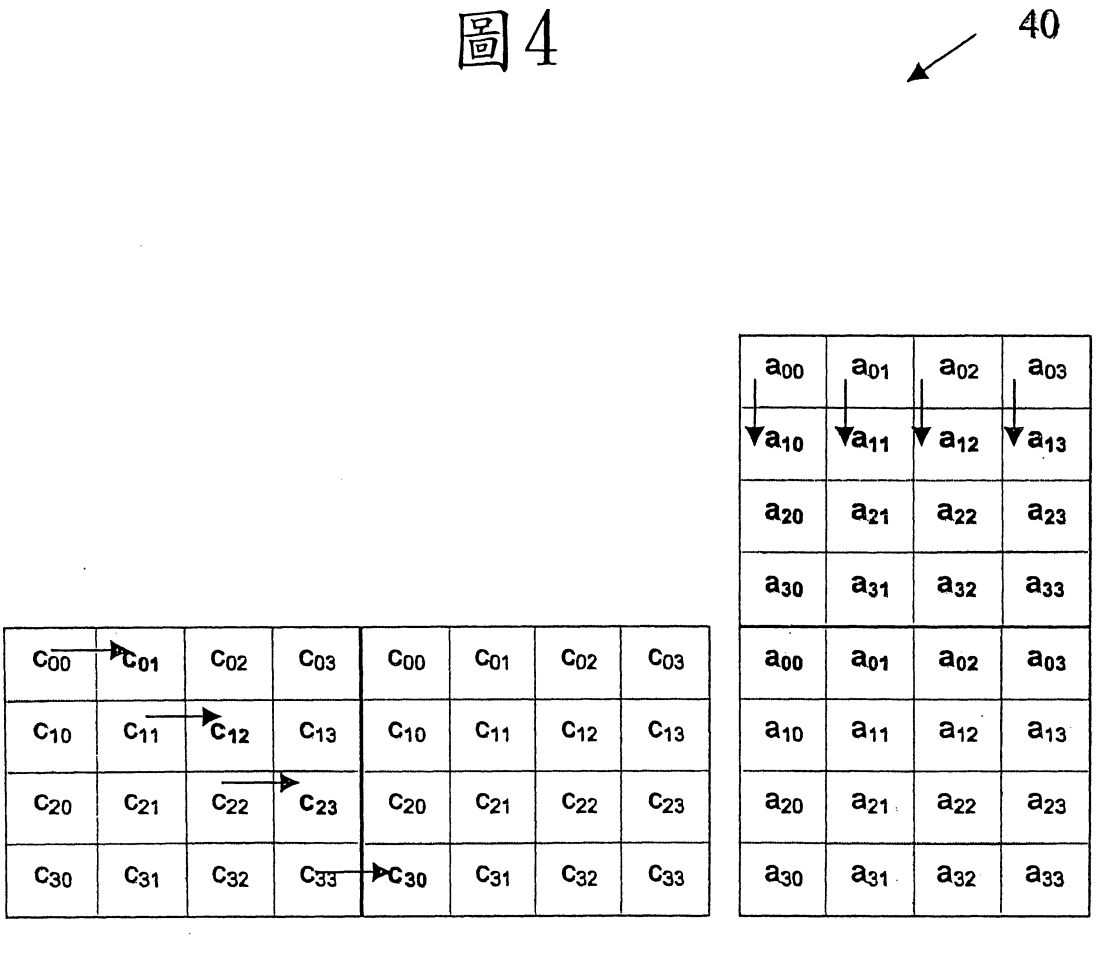

40
a00 a01 a02 a03
a10 a11 a12 a13
a20 a21 a22 a23
a30 a31 a32 a33
c00 c01 c02 c03
c10 c11 c12 c13
c20 c21 c22 c23
c30 c31 c32 c33
c00 c01 c02 c03
c10 c11 c12 c13
c20 c21 c22 c23
c30 c31 c32 c33
a00 a01 a02 a03
a10 a11 a12 a13
a20 a21 a22 a23
a30 a31 a32 a33

圖5

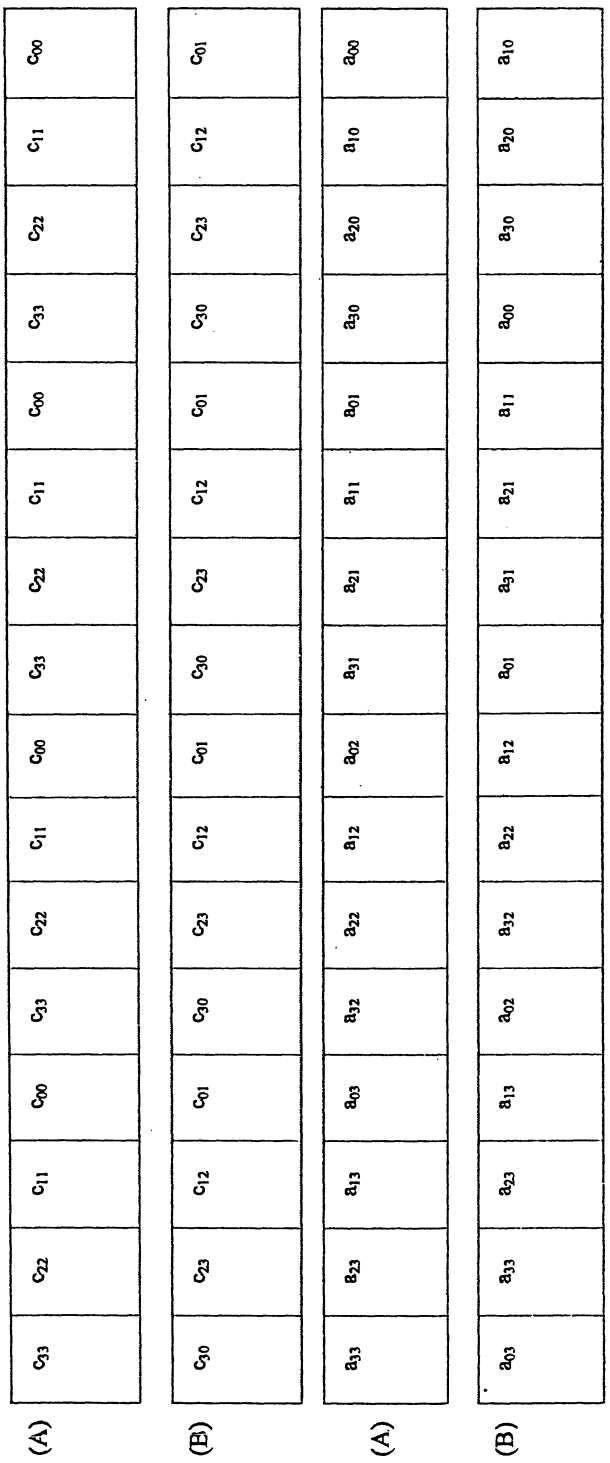
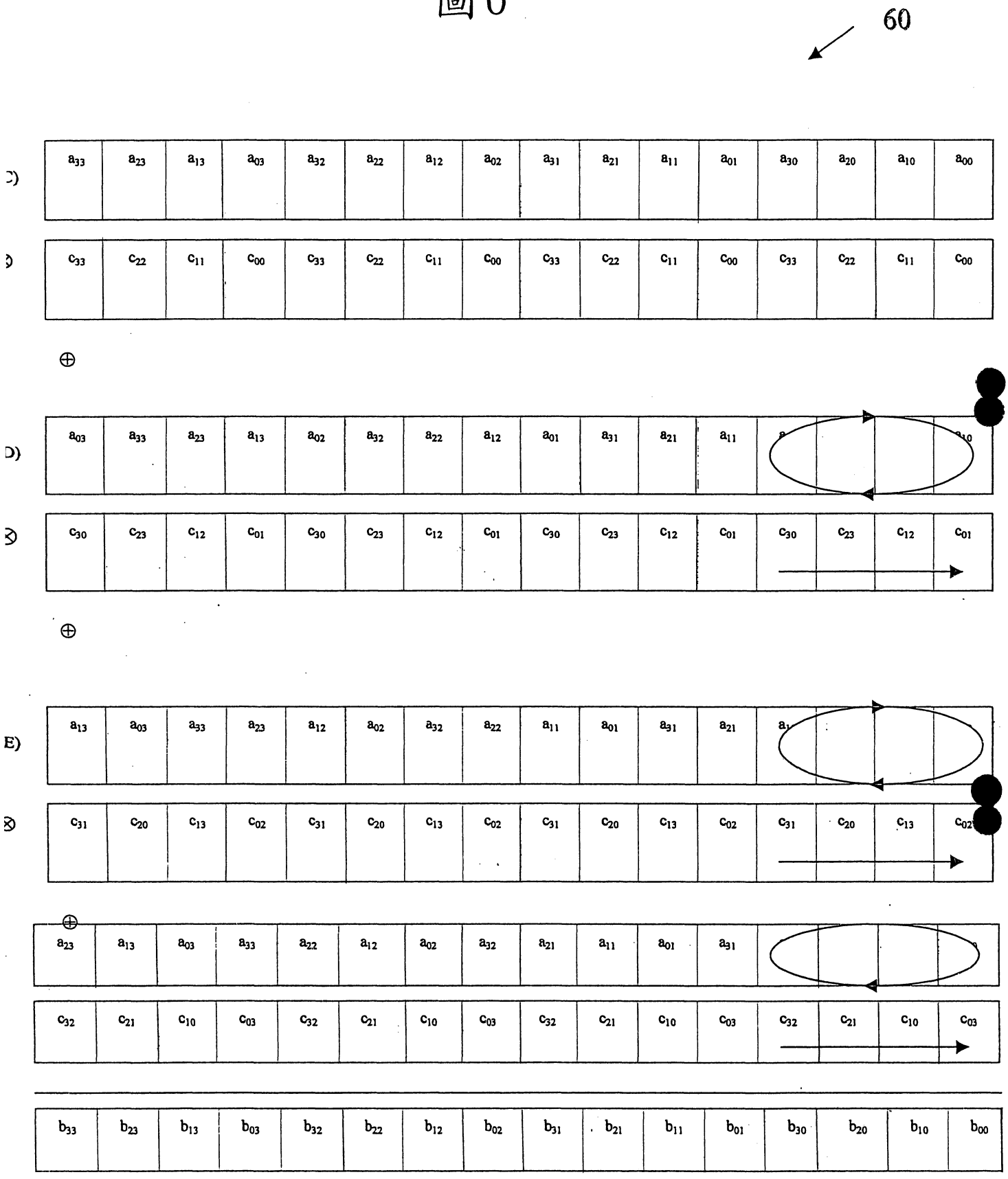


圖6


60
C)
a_{33} a_{23} a_{13} a_{03} a_{32} a_{22} a_{12} a_{02} a_{31} a_{21} a_{11} a_{01} a_{30} a_{20} a_{10} a_{00}
c_{33} c_{22} c_{11} c_{00} c_{33} c_{22} c_{11} c_{00} c_{33} c_{22} c_{11} c_{00} c_{33} c_{22} c_{11} c_{00}
⊕
D)
a_{03} a_{33} a_{23} a_{13} a_{02} a_{32} a_{22} a_{12} a_{01} a_{31} a_{21} a_{11}
c_{30} c_{23} c_{12} c_{01} c_{30} c_{23} c_{12} c_{01} c_{30} c_{23} c_{12} c_{01} c_{30} c_{23} c_{12} c_{01}
⊕
E)
a_{13} a_{03} a_{33} a_{23} a_{12} a_{02} a_{32} a_{22} a_{11} a_{01} a_{31} a_{21}
c_{31} c_{20} c_{13} c_{02} c_{31} c_{20} c_{13} c_{02} c_{31} c_{20} c_{13} c_{02} c_{31} c_{20} c_{13}
⊕
a_{23} a_{13} a_{03} a_{33} a_{22} a_{12} a_{02} a_{32} a_{21} a_{11} a_{01} a_{31}
c_{32} c_{21} c_{10} c_{03} c_{32} c_{21} c_{10} c_{03} c_{32} c_{21} c_{10} c_{03} c_{32} c_{21} c_{10} c_{03}
b_{33} b_{23} b_{13} b_{03} b_{32} b_{22} b_{12} b_{02} b_{31} b_{21} b_{11} b_{01} b_{30} b_{20} b_{10} b_{00}

70

圖7

b_{00}	b_{01}	b_{02}	b_{03}
b_{10}	b_{11}	b_{12}	b_{13}
b_{20}	b_{21}	b_{22}	b_{23}

=

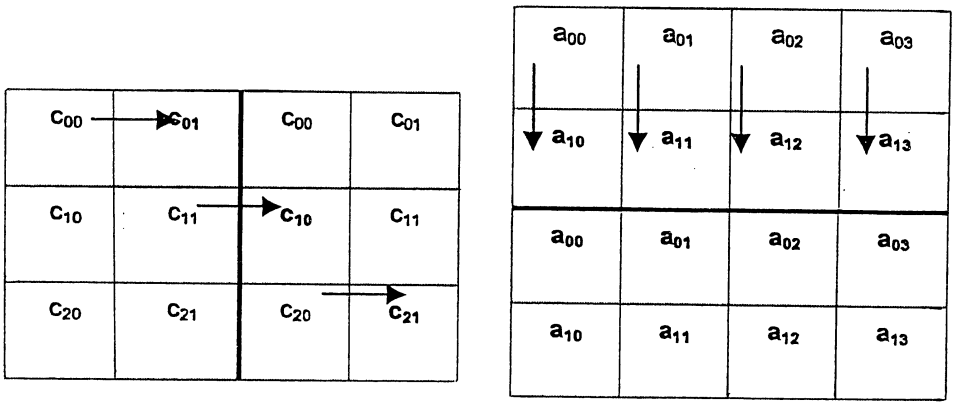
c_{00}	c_{01}
c_{10}	c_{11}
c_{20}	c_{21}

$\otimes$

a_{00}	a_{01}	a_{02}	a_{03}
a_{10}	a_{11}	a_{12}	a_{13}

80

圖8


c00
c01
c00
c01
c10
c11
c10
c11
c20
c21
c20
c21
a00
a01
a02
a03
a10
a11
a12
a13
a00
a01
a02
a03
a10
a11
a12
a13

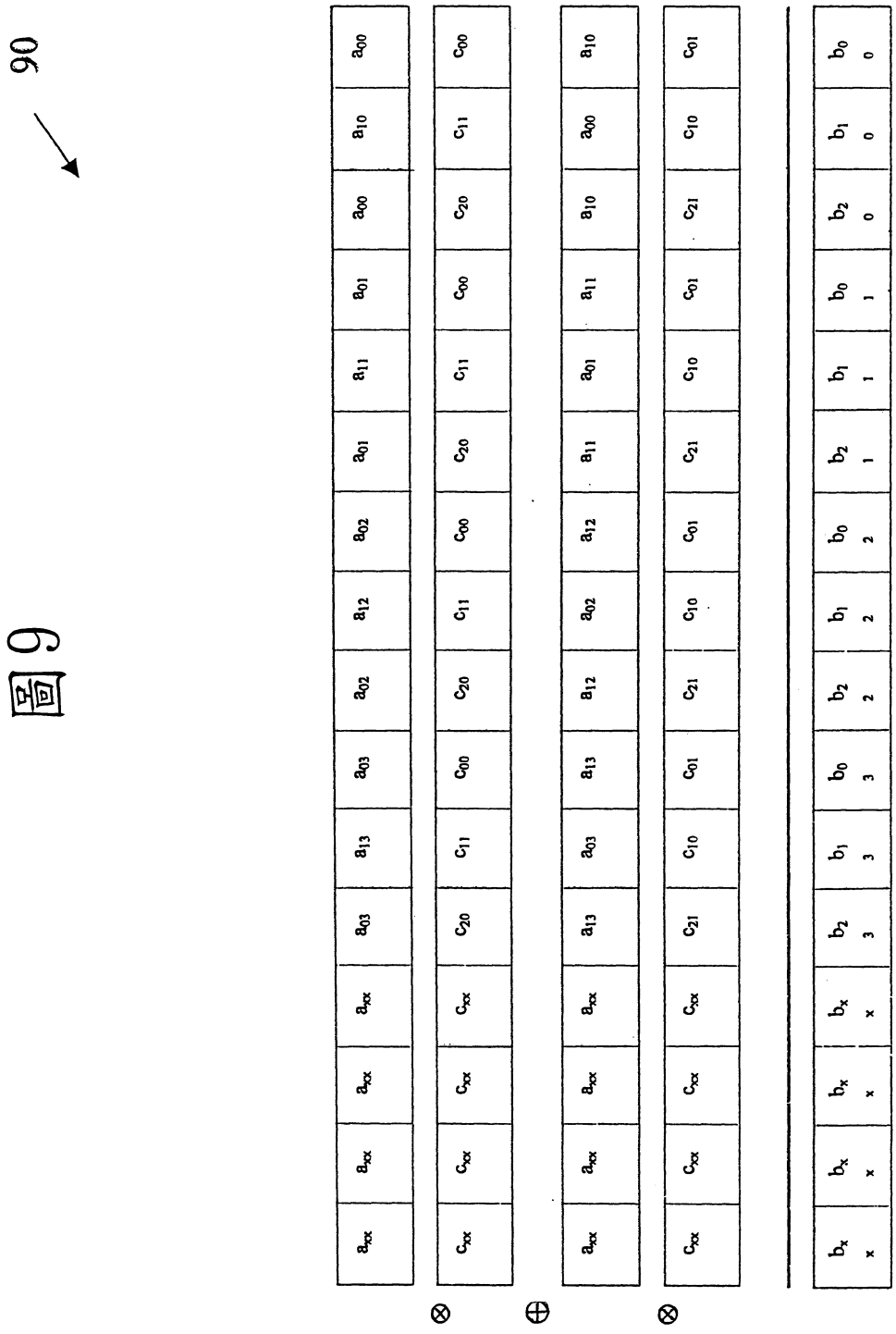


圖 9

圖10

100

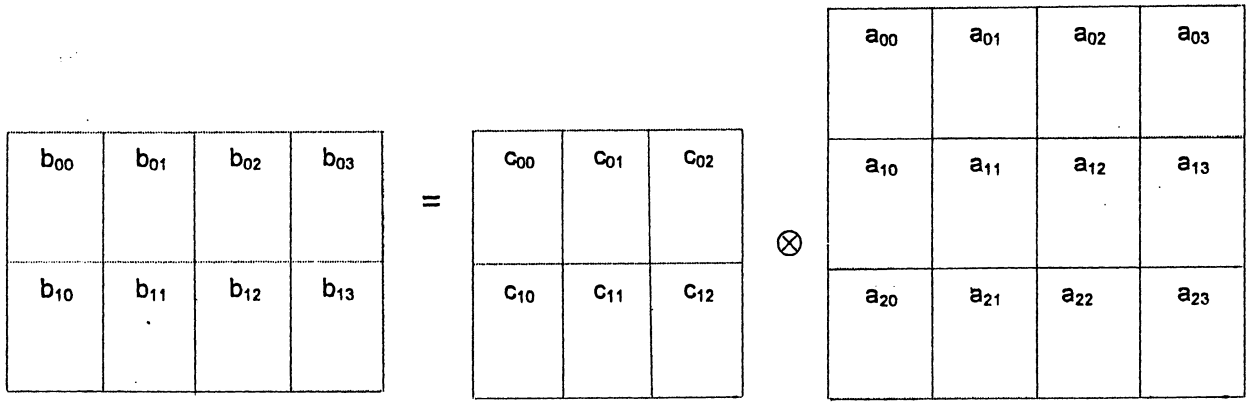

b00 b01 b02 b03
b10 b11 b12 b13
=
c00 c01 c02
c10 c11 c12
⊗
a00 a01 a02 a03
a10 a11 a12 a13
a20 a21 a22 a23

圖11

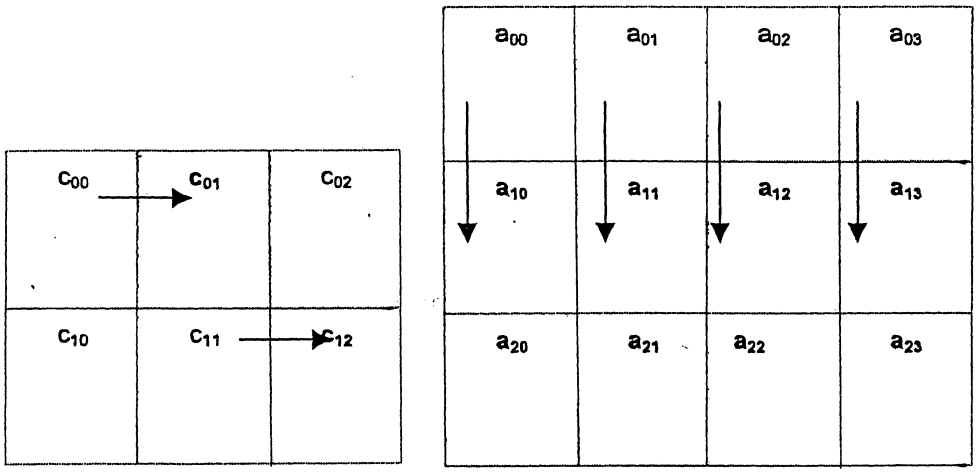

110
a_{00} a_{01} a_{02} a_{03}
a_{10} a_{11} a_{12} a_{13}
a_{20} a_{21} a_{22} a_{23}
c_{00} c_{01} c_{02}
c_{10} c_{11} c_{12}

圖12

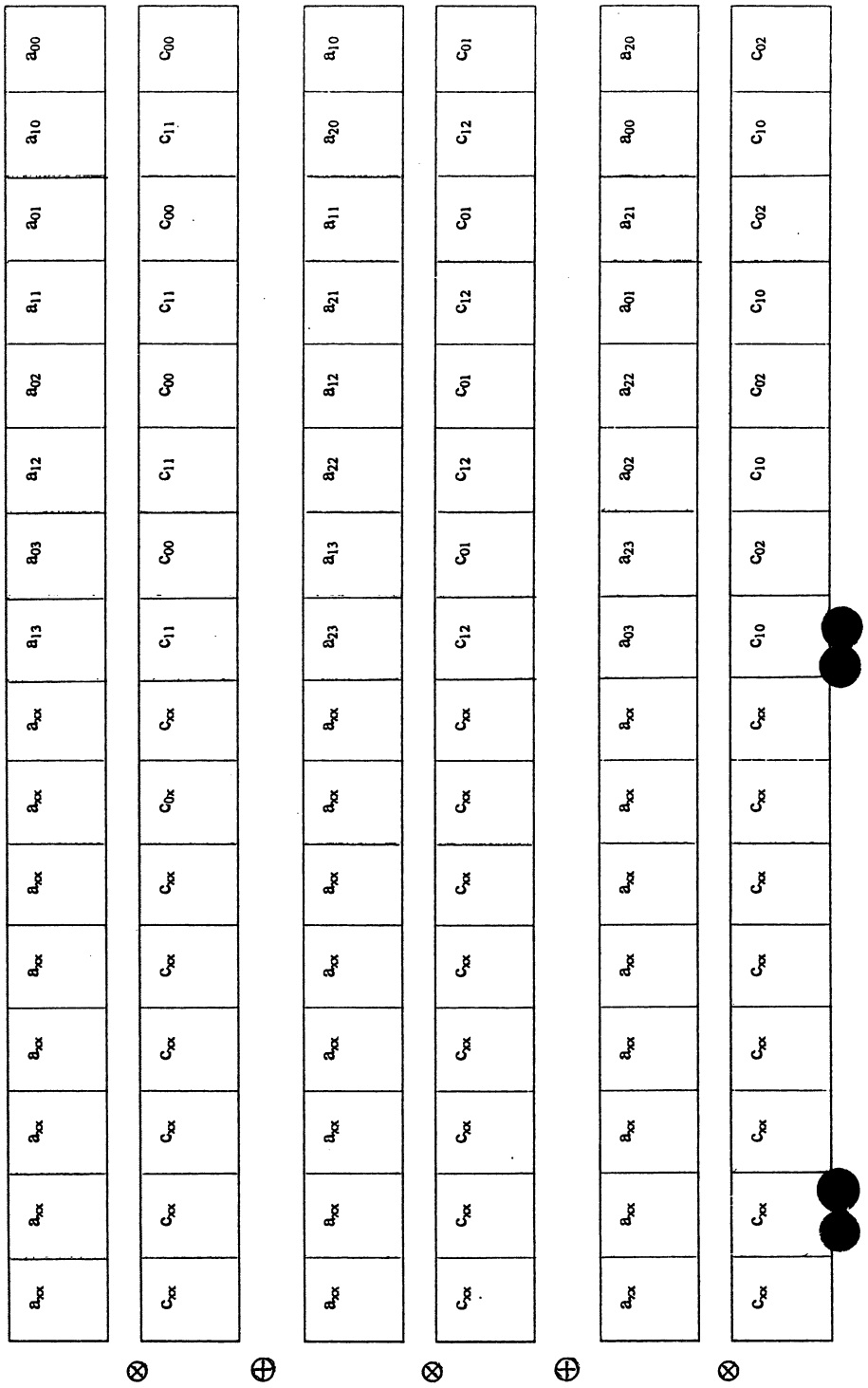
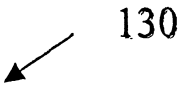

120
a_{00} a_{10} a_{01} a_{11} a_{02} a_{12} a_{03} a_{13} a_{0x} a_{0x} a_{0x} a_{0x} a_{0x} a_{0x} a_{0x}
c_{00} c_{11} c_{00} c_{11} c_{00} c_{11} c_{00} c_{11} c_{0x} c_{0x} c_{0x} c_{0x} c_{0x} c_{0x} c_{0x}
⊗
a_{10} a_{20} a_{11} a_{21} a_{12} a_{22} a_{13} a_{23} a_{0x} a_{0x} a_{0x} a_{0x} a_{0x} a_{0x} a_{0x}
c_{01} c_{12} c_{01} c_{12} c_{01} c_{12} c_{01} c_{12} c_{0x} c_{0x} c_{0x} c_{0x} c_{0x} c_{0x} c_{0x}
⊕
⊗
a_{20} a_{00} a_{21} a_{01} a_{22} a_{02} a_{23} a_{03} a_{0x} a_{0x} a_{0x} a_{0x} a_{0x} a_{0x} a_{0x}
c_{02} c_{10} c_{02} c_{10} c_{02} c_{10} c_{02} c_{10} c_{0x} c_{0x} c_{0x} c_{0x} c_{0x} c_{0x} c_{0x}
⊕
⊗

圖13



$$b(x) = c(x) * a(x)$$

$$\begin{bmatrix} b_{00} & b_{01} & b_{02} & b_{03} \\ b_{10} & b_{11} & b_{12} & b_{13} \\ b_{20} & b_{21} & b_{22} & b_{23} \\ b_{30} & b_{31} & b_{32} & b_{33} \end{bmatrix} = \begin{bmatrix} c_{00} & c_{01} & c_{02} & c_{03} \\ c_{10} & c_{11} & c_{12} & c_{13} \\ c_{20} & c_{21} & c_{22} & c_{23} \\ c_{30} & c_{31} & c_{32} & c_{33} \end{bmatrix} * \begin{bmatrix} a_{00} & a_{01} & a_{02} & a_{03} \\ a_{10} & a_{11} & a_{12} & a_{13} \\ a_{20} & a_{21} & a_{22} & a_{23} \\ a_{30} & a_{31} & a_{32} & a_{33} \end{bmatrix}$$

140

圖14

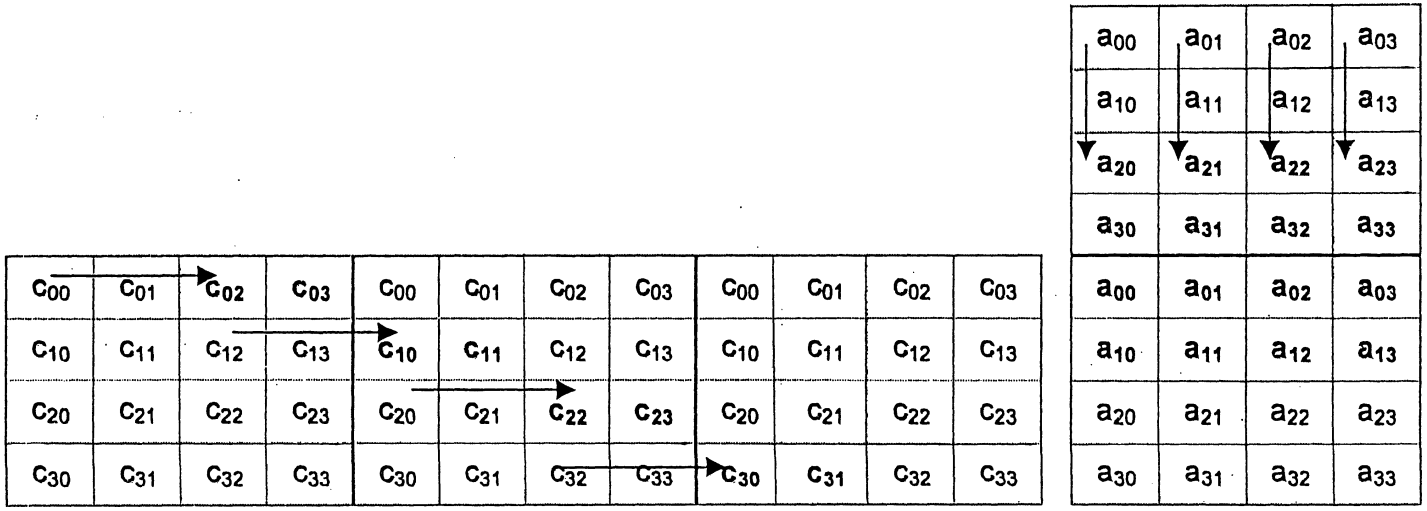
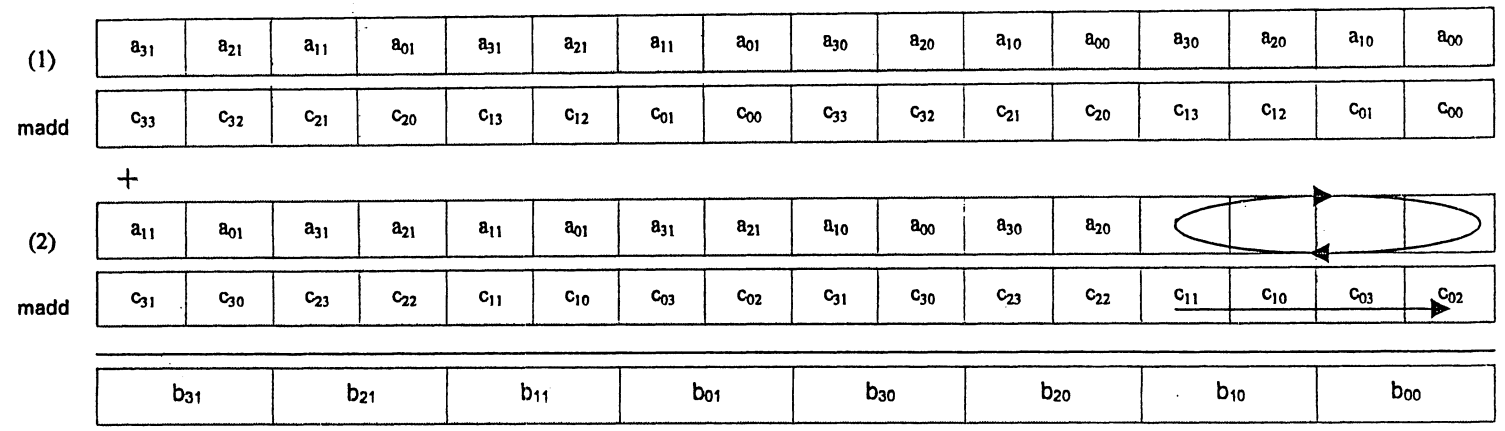
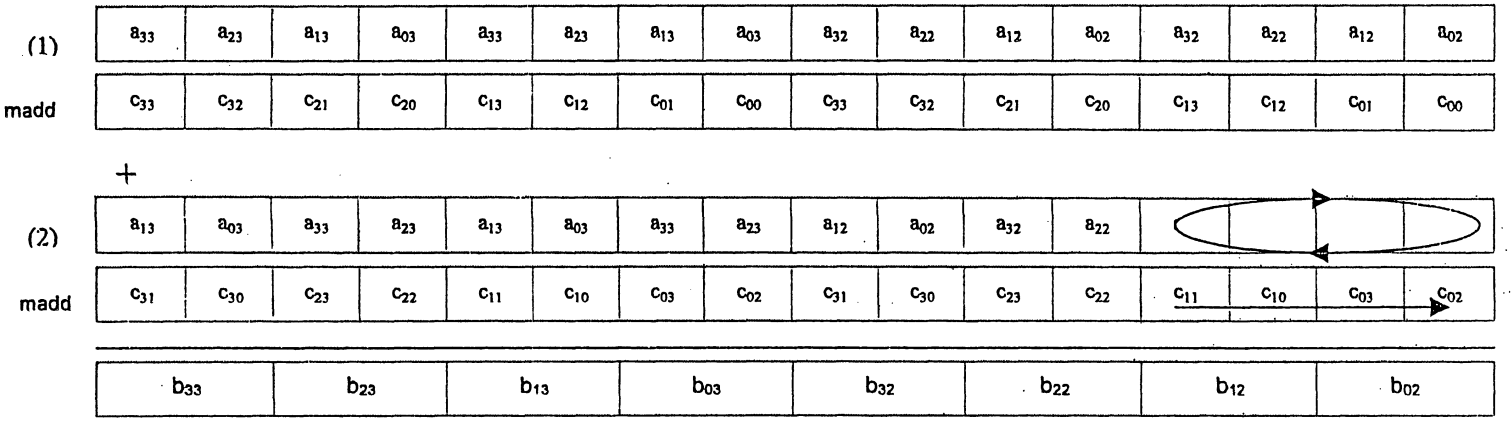

c00 c01 c02 c03 c00 c01 c02 c03 c00 c01 c02 c03
c10 c11 c12 c13 c10 c11 c12 c13 c10 c11 c12 c13
c20 c21 c22 c23 c20 c21 c22 c23 c20 c21 c22 c23
c30 c31 c32 c33 c30 c31 c32 c33 c30 c31 c32 c33
a00 a01 a02 a03
a10 a11 a12 a13
a20 a21 a22 a23
a30 a31 a32 a33
a00 a01 a02 a03
a10 a11 a12 a13
a20 a21 a22 a23
a30 a31 a32 a33

圖15

150



矩陣a之首二行之正規乘法用之暫存器中之資料之順序



矩陣a之次二行之正規乘法用之暫存器中之資料之順序

柒、指定代表圖：

（一）、本案指定代表圖為：第 1 圖

（二）、本代表圖之元件代表符號簡單說明：

10：電腦系統

12：處理器

13：記憶系統

14：內部記憶暫存器

15：I/O 裝置

16：單指令多資料暫存器

18：資料儲存單元

捌、本案若有化學式時，請揭示最能顯示發明特徵的化學式：

無