



(19)中華民國智慧財產局

(12)發明說明書公開本

(11)公開編號：TW 201007472 A1

(43)公開日：中華民國 99 (2010) 年 02 月 16 日

(21)申請案號：098119101

(22)申請日：中華民國 98 (2009) 年 06 月 08 日

(51)Int. Cl. : G06F15/80 (2006.01)

(30)優先權：2008/08/15 美國 12/192,513

(71)申請人：飛思卡爾半導體公司(美國) FREESCALE SEMICONDUCTOR, INC. (US)
美國

(72)發明人：摩伊爾 威廉 C MOYER, WILLIAM C. (US)

(74)代理人：陳長文

申請實體審查：無 申請專利範圍項數：21 項 圖式數：13 共 54 頁

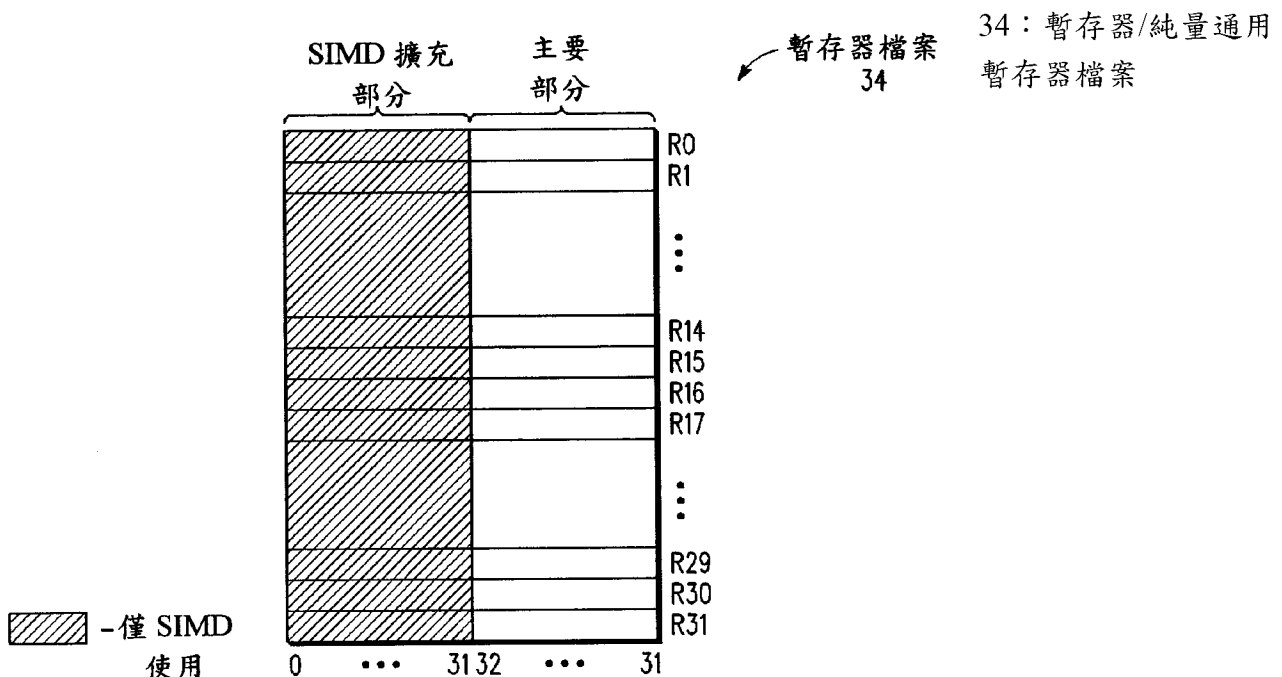
(54)名稱

在單一指令多資料之資料處理器中的擴充尋址模式之規定

PROVISION OF EXTENDED ADDRESSING MODES IN A SINGLE INSTRUCTION MULTIPLE DATA (SIMD) DATA PROCESSOR

(57)摘要

由一 N 位元處理器(14)執行一具有更新之第一記憶體存取指令包含存取複數個暫存器(34)中之至少一個源暫存器，其中該存取包含存取一第一暫存器，其中該複數個暫存器中之每一暫存器包含一 N 個位元之主要部分及一 M 個位元之擴充部分，其中該第一暫存器之該主要部分包含一第一位址運算元。該第一指令之該執行進一步包含：使用該第一位址運算元形成一記憶體存取位址；使用該記憶體存取位址作為一用於一記憶體存取之位址；產生一已更新之位址運算元；及將該已更新之位址運算元寫入至該第一暫存器之該主要部分。該產生包含存取該至少一個源暫存器中之一源暫存器之一擴充部分以獲得修改資訊且在該產生一已更新之位址運算元中使用該修改資訊。





(19)中華民國智慧財產局

(12)發明說明書公開本

(11)公開編號：TW 201007472 A1

(43)公開日：中華民國 99 (2010) 年 02 月 16 日

(21)申請案號：098119101

(22)申請日：中華民國 98 (2009) 年 06 月 08 日

(51)Int. Cl. : G06F15/80 (2006.01)

(30)優先權：2008/08/15 美國 12/192,513

(71)申請人：飛思卡爾半導體公司(美國) FREESCALE SEMICONDUCTOR, INC. (US)
美國

(72)發明人：摩伊爾 威廉 C MOYER, WILLIAM C. (US)

(74)代理人：陳長文

申請實體審查：無 申請專利範圍項數：21 項 圖式數：13 共 54 頁

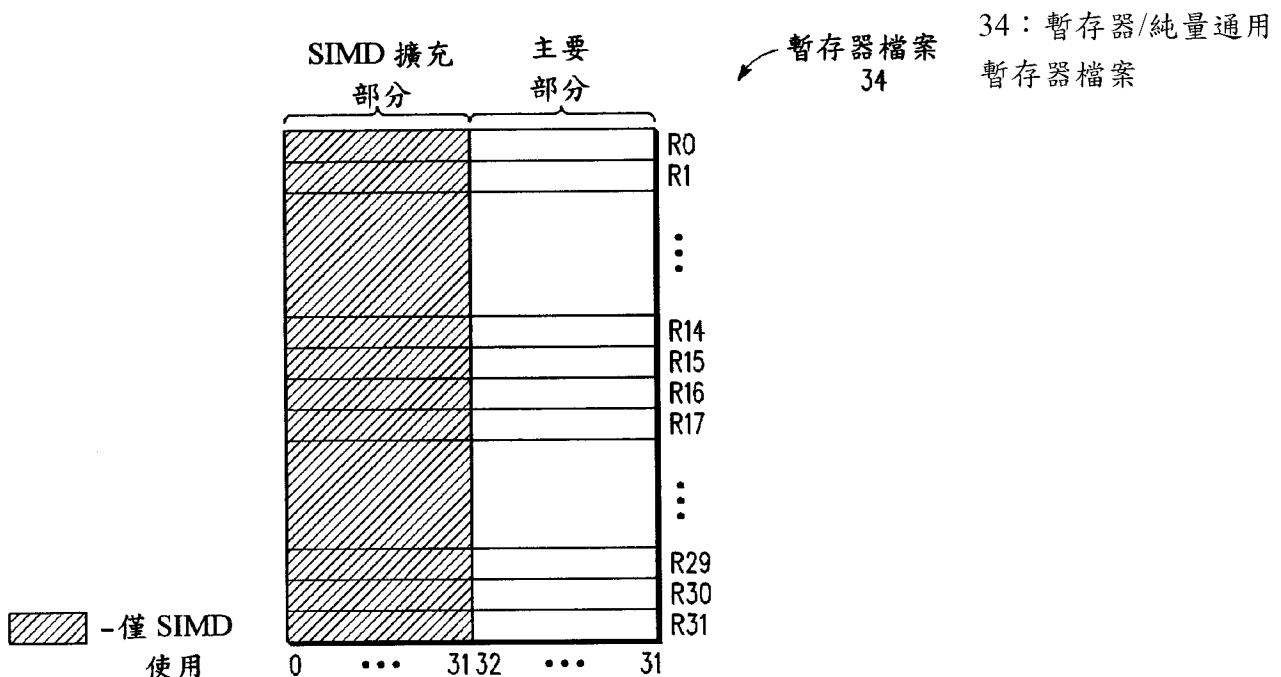
(54)名稱

在單一指令多資料之資料處理器中的擴充尋址模式之規定

PROVISION OF EXTENDED ADDRESSING MODES IN A SINGLE INSTRUCTION MULTIPLE DATA (SIMD) DATA PROCESSOR

(57)摘要

由一 N 位元處理器(14)執行一具有更新之第一記憶體存取指令包含存取複數個暫存器(34)中之至少一個源暫存器，其中該存取包含存取一第一暫存器，其中該複數個暫存器中之每一暫存器包含一 N 個位元之主要部分及一 M 個位元之擴充部分，其中該第一暫存器之該主要部分包含一第一位址運算元。該第一指令之該執行進一步包含：使用該第一位址運算元形成一記憶體存取位址；使用該記憶體存取位址作為一用於一記憶體存取之位址；產生一已更新之位址運算元；及將該已更新之位址運算元寫入至該第一暫存器之該主要部分。該產生包含存取該至少一個源暫存器中之一源暫存器之一擴充部分以獲得修改資訊且在該產生一已更新之位址運算元中使用該修改資訊。



六、發明說明：

【發明所屬之技術領域】

概言之，本揭示內容係關於資料處理器，且更特定而言，係關於在SIMD暫存器擴充中提供擴充尋址模式。

本申請案已於2008年8月15日在美國作為專利申請案第12/192,513號予以申請。

【先前技術】

可藉由允許並行執行對一向量之多個元素之運作達成資料處理系統中增加之效能。當前可用之一種類型之處理器係一利用向量暫存器來執行向量作業之向量處理器。然而，儘管可允許較高效能，但與使用純量通用暫存器之處理器相比向量處理器亦具有增加之複雜性及成本。亦即，向量處理器內之一向量暫存器檔案通常包含N個向量暫存器，其中每一向量暫存器包含一組用於保持M個元素之M個暫存器。另一類型之已知處理器係一單一指令多資料(SIMD)純量處理器(亦將其稱作一「短向量機」)，該處理器在使用一純量通用暫存器(GPR)時允許有限向量處理。因此，雖然與向量處理器相比每一作業之元素數目係有限的，但減少硬體係所需的。

在SIMD處理器上執行之諸多不同應用程式需要專用尋址，例如循環尋址或位元反轉尋址。然而，載入及儲存指令由於(舉例而言)較大位移通常需要大量有限作業碼空間來解碼。因此，通常未留下支援任何額外功能性之作業碼空間。因此，在不增加作業碼空間之情況下之額外尋址控

制之說明係合意的。

【發明內容】

在一個實施例中，提供支援不同尋址模式之使用之 SIMD 資料處理指令。舉例而言，在一個實施例中，一指數形式向量載入或儲存指令規定兩個源通用暫存器，其中一個源通用暫存器 (GPR) 之主要部分係用以提供一基底位址且另一源 GPR 之主要部分係用以提供一索引值 (亦即，一偏移值)。提供基底位址之源 GPR 之擴充部分或提供索引值之源 GPR 之擴充部分係用以提供額外尋址控制資訊，該額外尋址控制資訊允許向量載入或儲存指令之所增加之功能性。在一個實施例中，一位移形式向量載入或儲存指令規定一個源 GPR，其中該一個源 GPR 之主要部分係用以提供一基底位址且該一個源 GPR 之擴充部分係用以提供額外尋址控制資訊。以此方式，可在不增加作業碼大小之情況下增加尋址模式功能性。

【實施方式】

如本文中所使用，術語「匯流排 (bus)」係用以指示複數個信號或導體，該等信號或導體可用以傳送一個或多個不同類型之資訊，例如資料、位址、控制或狀態。如本文中所論述之導體可參照係一單個導體、複數個導體、單向導體或雙向導體予以圖解闡釋或闡述。然而，不同實施例可改動該等導體之實施方案。舉例而言，可使用若干單獨單向導體而非雙向導體，且反之亦然。同樣，可以一串行地或以一時間多工方式傳送多個信號之單個導體代替複數個

導體。同樣地，可將攜載多個信號之若干單個導體分離成攜載此等信號之子組之各種不同導體。因此，存在用於傳送信號之諸多選項。

當提及分別將一信號、狀態位元或類似裝置轉譯成其邏輯真或邏輯假狀態時，本文中使用的術語「斷定(assert)」或「設定(set)」及「否定(negate)」(或「去斷定(deassert)」或「清除(clear)」)。若邏輯真狀態係一邏輯位準1，則邏輯假狀態係一邏輯位準0。且若邏輯真狀態係一邏輯位準0，則邏輯假狀態係一邏輯位準1。

可將本文中所闡述之每一信號設計為正或負邏輯，其中可由一信號名稱上方之橫號或該名稱之後之一星號(*)表示負邏輯。在一負邏輯信號之情況下，該信號係有效低，其中該邏輯真狀態對應於一邏輯位準0。在一正邏輯信號之情況下，該信號係有效高，其中該邏輯真狀態對應於一邏輯位準1。應注意，可將本文中所闡述之信號中之任一者設計為負或正邏輯信號。因此，在替代實施例中，可將闡述為正邏輯信號之彼等信號實施為負邏輯信號，且可將闡述為負邏輯信號之彼等信號實施為正邏輯信號。

同樣，如本文中所使用，一字(w)包含4個位元組，一半字(h)包含2個位元組，且一雙字(d)包含8個位元組。然而，在替代實施例中，可將一字界定為2個位元組且將一雙字界定為4個位元組。在一數字之前之符號「\$」或「0x」指示該數字係以其十六進製或以16為基數形式表示。在一數字之前之符號「%」指示該數字係以其二進制

或以2為基數形式表示。

圖1以方塊圖形式圖解闡釋根據本發明之一個實施例之一資料處理系統10。由於資料處理系統10能夠執行SIMD指令，因此亦可將系統10稱作一SIMD資料處理系統。資料處理系統10包含：一記憶體12、一處理器14、一輸入/輸出(I/O)16、其他周邊器件18及一系統匯流排20。記憶體12經由導體22雙向耦合至系統匯流排20，I/O 16經由導體24雙向耦合至系統匯流排20，其他周邊器件18經由導體26雙向耦合至系統匯流排20，且處理器14經由導體58雙向耦合至系統匯流排20。在一個實施例中，其他周邊器件18可包含一個或多個周邊器件，其中每一器件可係任一類型之周邊器件，例如一通用異步接發器(UART)、一即時時鐘(RTC)、一鍵盤控制器、其他記憶體等。其他周邊器件18中之某些器件或全部器件能夠經由導體62傳遞資料處理系統10外部之資訊。I/O 16可包含任一類型之I/O電路，該電路經由(舉例而言)導體60接收或提供資料處理系統10外部之資訊。記憶體12可係任一類型之記憶體，例如一唯讀記憶體(ROM)、一隨機讀取記憶體(RAM)、非揮發性記憶體(例如，快閃)等。資料處理系統10可包含除所圖解闡釋之彼等元件以外之元件，或可包含比圖解闡釋之彼等元件更多或更少之元件。舉例而言，資料處理系統10可包含任一數目之記憶體或處理器。

處理器14可係任一類型之處理器，例如一微處理器、微控制器、數位信號處理器等。在一個實施例中，可將處理

器14稱為一處理器核心。在另一實施例中，處理器14可係一多重處理器資料處理系統中之諸多處理器中之一者。此外，雖然未如此圖解闡釋，但處理器14可係一管線處理器。亦應注意，亦可將處理器14稱作一SIMD處理器。在圖1中所圖解闡釋之實施例中，處理器14包含：一控制單元28、一指令單元30、若干執行單元32、一純量通用暫存器(GPR)檔案34、一匯流排介面單元(BIU)36及一載入/儲存單元38。控制單元28經由導體40雙向耦合至指令單元30，經由導體42雙向耦合至執行單元32，經由導體46雙向耦合至純量檔案34，且經由導體48雙向耦合至載入/儲存單元38。執行單元32經由導體44雙向耦合至純量暫存器檔案34，且純量暫存器檔案34經由導體50雙向耦合至載入/儲存單元38。BIU 36經由導體54雙向耦合至指令單元30且經由導體52雙向耦合至載入/儲存單元38。處理器14能夠經由耦合至導體58之導體56與系統匯流排20雙向通信。應注意，處理器14可包含比所圖解闡釋之電路更多之電路，其中額外電路亦可耦合至導體58。亦即，導體56可經由導體58中之全部或一部分與系統匯流排20通信。亦應注意，可將處理器14之全部或一部分稱作處理電路。

在運作中，指令單元30經由BIU 36及系統匯流排20自一記憶體(例如，記憶體12)提取指令，且自控制單元28接收控制資訊及將該控制資訊提供至控制單元28。指令單元30可係如此項技術中已知之任一類型之指令單元，且可如此項技術中已知方式運作，且因此本文中將不再更加詳細地

闡述。因此，指令單元30將指令提供至控制單元28，控制單元28控制經由(舉例而言)執行單元32及載入/儲存單元38對此等所接收指令之執行，執行單元及載入/儲存單元兩者能夠按需直接或經由控制單元28與純量暫存器檔案34通信。舉例而言，控制單元28經由載入/儲存單元38及BIU 36能夠按需將來自記憶體(例如，記憶體12)之資料載入至純量暫存器檔案34內之暫存器以用於執行指令且能夠按需將來自純量暫存器檔案34內之暫存器之資料儲存至記憶體(例如，記憶體12)以用於執行指令。舉例而言，在一個實施例中，載入/儲存單元38可基於經由導體48自控制單元28提供之控制資訊經由導體50直接與純量暫存器檔案34通信(以讀取及寫入資料)。執行單元32可使用儲存於純量暫存器檔案34內之資料執行算術、邏輯、移位或其他作業且將結果儲存至純量暫存器檔案34內之暫存器，此係用於執行由控制單元28自指令單元30接收之指令所需。舉例而言，執行單元32可包含算術邏輯單元(ALU)、浮動點單元等。純量暫存器檔案34可經由導體46將控制資訊或資料提供至控制單元28或自控制單元28接收該控制資訊或資料。

除對在理解本文中所闡述之各種實施例中所需之彼等部分以外，本文中將不再更加詳細地闡述處理器14之作業。亦應注意，可按需修改具有儲存於一純量通用暫存器檔案中之運算元之資料處理系統之現有設計以執行向量載入及儲存指令。

圖2中圖解闡釋純量通用暫存器檔案34之一方塊圖。在

所圖解闡釋之實施例中，純量暫存器檔案34包含32個通用暫存器(GPR)。然而，在替代實施例中，純量暫存器檔案可包含任一數目之GPR。如本文中所使用，一純量暫存器指示一具有一一維映射且因此僅保持一個資料列之暫存器。純量暫存器檔案34中之每一GPR包含一主要部分(其包含每一GPR之N個下部位元)及一SIMD擴充部分(其包含每一GPR之M個上部位元)。在圖解闡釋之實施例中，N係32個位元且M係32個位元，以使得每一GPR係一能夠儲存一用於SIMD作業之64位元量之64位元暫存器。在本文中所闡述之實施例中，純量暫存器檔案34中之每一GPR包含8個位元組，其中當用於SIMD作業時，每一GPR可儲存(舉例而言)2個字大小之向量元素、4個半字大小之向量元素或8個位元組大小之向量元素。

在所圖解闡釋之實施例中，當執行非SIMD指令時或當形成用於載入或存儲指令(其中包含向量載入及向量儲存指令)之有效位址(EA)值時，GPR檔案34之主要部分係用於其中不使用GPR檔案34之SIMD擴充部分之非SIMD作業。此外，當執行非SIMD作業時，保存儲存於一GPR之一擴充部分中之任何值。在一個實施例中，用於主要部分之位元數目N係由處理器14之架構確定。舉例而言，可將處理器14視為一N位元處理器。一N位元處理器係指一其原始(例如，未經擴充)整數資料類型係N個位元之處理器。通常，一N位元處理器中之指標係N個位元；因此，一指標值(儲存於一GPR之一主要部分內)能夠存取 2^N 個位元組。

亦即，一N位元處理器具有一N位元之位址空間。因此，在一個實施例中，用於GPR之主要部分之位元數目N係由處理器之類型或處理器之指令集架構確定。因此，GPR之經擴充部分之位元數目M可擴充超過N個位元。在所圖解闡釋之實施例中，假定處理器14係一32位元處理器(其原始(未經擴充)整數資料類型係32個位元且其指標值係32個位元)。因此，GPR之主要部分係32個位元(亦即， $N=32$)。在所圖解闡釋之實施例中，GPR之主要部分對應於暫存器檔案34之下部部分(例如，下半部分)(亦即，每一GPR之位元32:63)。在所圖解闡釋之實施例中，經擴充部分亦係32個位元(亦即， $M=32$)，且對應於暫存器檔案34之上部部分(例如，上半部分)(亦即，每一GPR之位元0:31)。另一選擇係，可不同地組織純量GPR檔案34，只要主要部分包含N個位元且擴充部分包含M個位元(超過N個位元)，以使得每一GPR(或用於SIMD作業之GPR之一子組)包含 $N+M$ 個位元。舉例而言，主要部分可對應於暫存器檔案34之一上部部分且擴充部分對應於暫存器檔案34之一下部部分。替代實施例可具有不同M值及N值，例如， $N=32$ 及 $M=96$ ，其中一SIMD向量之寬度對應地大於此等實施例之原始整數資料類型，且因此支援更大SIMD向量長度，同時維持處理器之基底N位元架構。

在圖2之所圖解闡釋之實施例中，暫存器檔案34之擴充部分係由陰影表示且由於該擴充部分僅用於SIMD作業可將其稱作一SIMD擴充部分。在一替代實施例中，其他專

用功能(例如，雙精度浮動點值)可使用該擴充部分。然而，在一個實施例中，擴充部分不用於處理器14內之非SIMD整數作業。同樣，應注意每一暫存器之擴充部分不可離開每一暫存器之主要部分獨立地存取或尋址。亦即，舉例而言，r0之擴充部分僅可藉由存取r0來存取且不可獨立於r0之主要部分被尋址，此乃因擴充部分僅係r0之N位元主要部分之一M位元擴充。此亦係圖2中所圖解闡釋之GPR r1-r31之擴充部分及主要部分中之每一者之情況。

圖13中圖解闡釋根據先前技術之一指數形式非SIMD(亦即，純量)整數載入指令(lwz[u]x)之一個實施例。如圖13中所圖解闡釋，lwz[u]x係一指示一目的地暫存器rD及兩個源暫存器rA及rB之32位元指令。該lwz[u]x指令允許一欲規定之更新值(U)指示待執行一具有指數而不具有更新之載入還是一具有指數及更新之載入。在U=1(lwzux)之情況下，欲作為指令之作業之部分來執行rA之一更新。在U=0(lwzx)之情況下，不執行rA之更新。該lwz[u]x指令包含一識別指令之功能或類型之作業碼欄位(位元0:5)。該指令內之一目的地欄位(位元6:10)係一識別純量暫存器檔案34內之一暫存器D之被標記為「rD」之目的地暫存器識別符。該指令內之一第一源欄位(位元11:15)係一識別純量暫存器檔案34內之一暫存器A之被標記為「rA」之源暫存器識別符，且該指令內之一第二源欄位(位元16:20)係一識別純量暫存器檔案34內之一暫存器B之被標記為「rB」之源暫存器識別符。該指令亦包含一可用以進一步規定該指令

之功能性之子作業碼欄位(位元21:31)。應注意，在所圖解闡釋之實施例中，U作為該指令之子作業碼內之一位元被包含在內(在所圖解闡釋之實施例中位於位元位置25處)。

該lwz[u]x指令將一來自記憶體之32位元整數字載入至目的地暫存器之主要部分中。因此，在該所圖解闡釋之其中處理器14係一32位元處理器之實施例中，lwz[u]x指令可用以將一32位元整數字載入至目的地暫存器之主要部分中。由於此係一正常(非SIMD)整數載入指令，因此資料暫存器中僅SIMD部分未受影響。該指令之作業與該指令在一無SIMD能力之ISA相容處理器上之作業相同。在執行一lwz[u]x指令後，rA之主要部分(其儲存一基底位址值)之內容及rB之主要部分(其儲存一索引值)之內容皆係用以計算一有效位址(例如，EA[0:31])。然後，將位於記憶體12中由EA[0:31]指向之記憶體位置處之32位元字載入至rD之主要(整數)部分中。亦即將記憶體12中由EA指向之字載入至rD[32:63]中。若U=0(亦即，一lwzx指令)，則不執行rA之更新。然而，若U=1(亦即，一lwzux指令)，則除了以記憶體資料更新rD之載入作業以外，待執行rA之一更新。亦即，可使用正常載入指令之更新形式來以一等於所計算之EA[0:31]值之新基底值更新rA之基底值，該新基底值用於針對一使用該已更新基底值之隨後載入指令之載入作業。因此，更新形式指令藉由將一載入作業與一位址運算元更新作業組合成一單一指令而提供額外並行性。應注意，對於正常整數載入指令而言，既不使用亦不更新暫存器rA、

rB及rD之擴充部分(僅SIMD部分)。

類似地，應注意，一對應正常(非SIMD、純量)整數儲存指令(stw[u]x rS、rA、rB)類似於載入指令運作。stw[u]x指令將來自rS指示之GPR之整數字儲存至記憶體中由EA指向之位置(其中儲存基底位址值之rA之內容及儲存索引值之rB之內容皆用以計算EA[0:31])。正如載入指令之情況，若U=0(亦即，一stwx指令)，則不執行rA之更新。然而，若U=1(亦即，一stwux指令)，則rA之一更新係藉由以所計算之EA[0:31]更新rA之主要部分而發生。應注意，對於整數儲存指令而言，既不使用亦不更新暫存器rA、rB、及rS之擴充部分(僅SIMD部分)。

圖3中圖解闡釋根據本發明之一個實施例之一指數形式向量(SIMD)載入指令(evldh[u]x)之一個實施例。在所圖解闡釋之實施例中，evldh[u]x係一指示一目的地暫存器rD及兩個源暫存器rA及rB之32位元指令。該evldh[u]x指令允許一欲規定之更新值(U)指示待執行一具有指數而不具有更新之載入還是一具有指數及更新之載入。在U=1(evldhux)之情況下，欲作為該指令之作業之部分執行rA之一更新，以下將更加詳細地闡述。在U=0(evldhx)之情況下，不執行rA之更新。該evldh[u]x指令包含一識別指令之功能或類型之作業碼欄位(位元0:5)。該指令內之一目的地欄位(位元6:10)係一識別純量暫存器檔案34內之一暫存器D之被標記為「rD」之目的地暫存器識別符。該指令內之一第一源欄位(位元11:15)係一識別純量暫存器檔案34內之一暫存器A

之被標記為「rA」之源暫存器識別符，且該指令內之一第二源欄位(位元16:20)係一識別純量暫存器檔案34內之一暫存器B之被標記為「rB」之源暫存器識別符。該指令亦包含一可用以進一步規定該指令之功能性之子作業碼欄位(位元21:31)。應注意，在所圖解闡釋之實施例中，U作為該指令之子作業碼內之一位元被包含在內(在所圖解闡釋之實施例中位於位元位置25處)。然而，在替代實施例中，可提供是否欲更新之指示作為該指令內之不同作業碼或作為該指令內之一不同位元。另一選擇係，可實施其他位元長度指令且每一指令欄位之位元大小皆係實施方案特定。然而，如以上所論述，應注意在諸多指令集架構中，作業碼及子作業碼空間係有限的，且因此額外指令位元長度可係不可用。同樣，在其他實施例中，可實施額外欄位或其他欄位。另外，可將欄位rD、rA及rB之排序改變為除圖3中所圖解闡釋之次序以外之其他序列。

該evldh[u]x指令將一來自記憶體之雙字載入至目的地暫存器之四個半字中。因此，在所圖解闡釋之其中處理器14係一32位元處理器之實施例中，該evldh[u]x指令可用以將4個16位元向量元素載入至SIMD目的地暫存器中。在執行一evldh[u]x指令後，rA(其儲存一基底位址值)及rB(其儲存一索引值)之內容皆用以計算一有效位址(例如，EA[0:31])。(應注意，若rB中之索引值係0，則rA中之基底位址值可直接用作EA。)然後，將位於記憶體12中由EA[0:31]指向之記憶體位置處之雙字載入至經組合之rD之擴充與主要部

分中。亦即，將記憶體 12 中由 EA 指向之半字載入至 rD[0:15] 中，將記憶體 12 中由 EA+2 指向之半字載入至 rD[16:31] 中，將記憶體 12 中由 EA+4 指向之半字載入至 rd[32:47] 中，且將記憶體 12 中由 EA+6 指向之半字載入至 rD[48:63] 中。(應注意，如以下將參照一循環緩衝尋址實例論述，此等 EA 之用以將 4 個向量元素載入至 rD 中之任一者可在一個循環緩衝器之一長度邊界處繞回。)若 U=0(亦即，一 evldhx 指令)，則不執行 rA 之更新。然而，若 U=1(亦即，一 evldhux 指令)，則除了以記憶體資料更新 rD 之整個內容(主要部分及擴充部分兩者)之載入作業以外，rA 或 rB 之擴充部分係用以確定欲如何執行 rA 之一更新。亦即，rA 或 rB 之一擴充部分可用以進一步提供控制資訊，該控制資訊界定 rA 之更新如何發生以實施不同尋址方案，例如線性尋址、具有環繞之循環緩衝尋址或位元反轉尋址。在一個實施例中，rA 之已更新值係基於 rA 之先前值，其中此可藉由修改基底位址或 EA 達成。可選擇 SIMD 載入指令之更新形式來以一新基底值更新 rA 之基底值，該新基底值用於一使用該已更新基底值之隨後載入指令。因此，更新形式指令可藉由將一載入作業與一位址運算元更新作業組合成一單一 SIMD 指令而提供額外並行性。

類似地，應注意對應之 SIMD 儲存指令(evstdh[u]x rS、rA、rB)類似於 SIMD 載入指令運作。該 evstdh[u]x 指令將來自 rS 指示之 GPR 之四個半字儲存至記憶體中由 EA 指向之位置(其中儲存基底位址值之 rA 之內容及儲存索引值之 rB

之內容皆用以計算EA[0:31])。(應注意，若rB中之索引值係0，則rA中之基底位址值可直接用作EA。)正如SIMD載入指令之情況，若U=0(亦即，一evstdhx指令)，則不執行rA之更新。然而，若U=1(亦即，一evstdhux指令)，則rA或rB之擴充部分係用以確定欲如何執行rA之更新。亦即，當執行SIMD載入或儲存指令時，rA或rB之一擴充部分可用以進一步提供控制資訊(亦即，修改資訊)，該控制資訊界定rA之更新如何發生以實施不同尋址技術，例如具有環繞之循環緩衝尋址或位元反轉尋址。因此，接下來關於計算EA及使用一擴充部分以進一步界定當U=1時如何執行一更新之說明適用於指數形式SIMD載入及指數形式SIMD儲存指令兩者。然而，為便於解釋，本文中之諸多實例對應於SIMD載入指令。同樣，應注意除一半字以外可在SIMD載入或儲存指令中界定任一元素大小。舉例而言，本文中所提供之說明亦可類似地適用於一分別自記憶體載入8個位元組或將該8個位元組儲存至記憶體之evldb[u]x或evstdb[u]x。該等說明亦可類似地適用於一分別自記憶體載入2個字或將該2個字儲存至記憶體之evldw[u]x或evstdw[u]x。

圖4圖解闡釋根據本發明之一個實例之一GPR之一擴充部分及一主要部分。GPR之主要部分(位元32:63)儲存一位址運算元(例如，基底位址或索引值)。因此，此可對應於由rA指派之GPR之主要部分(其儲存基底位址作為一位址運算元)或由rB指派之GPR之主要部分(其儲存索引值作為

一位址運算元)。擴充部分(位元0:31)包含一模式欄位及一位址更新控制欄位。在所圖解闡釋之實施例中，模式欄位係一位於位元0:2中之3位元欄位且位址更新控制欄位係一位於GPR之位元3:31中之29位元欄位。模式欄位可用以界定針對一SIMD載入或儲存指令待執行何種類型之位址更新，且若需要，位址更新控制欄位可提供用於該更新之額外控制資訊。在一個實施例中，若模式欄位具有一%000之值，則當執行具有更新之正常整數載入及儲存指令時執行之一正常更新作業(以上參照圖13所論述)經指示其中所計算之EA[0:31]被儲存至rA之主要部分中(rA[32:63])。若該模式欄位具有一%001之值，則執行一循環緩衝位址更新，其中以欲存取之下一循環緩衝器元素之位址值更新rA之主要部分(rA[32:63])。儲存於位址更新控制欄位中之額外資訊係用以執行此更新，以下將參照圖5及6更加詳細地闡述。若模式欄位具有一%010之值，則執行一位元反轉位址更新，其中以欲根據位元反轉尋址(例如，以存取一儲存FFT資料之緩衝器)而存取之下一緩衝器元素之位址值更新rA之主要部分(rA[32:63])。儲存於位址更新控制欄位中之額外資訊係用以執行此更新，以下將參照圖7-11更加詳細地闡述。因此，若針對SIMD載入或儲存指令U=1，則rA或rB之擴充部分之模式欄位之值係用以指示將如何執行rA之主要部分之一更新，其中按需使用位址更新控制欄位以執行此等更新。應注意，若U=0，則不存取擴充部分。亦應注意，無論用於彼等指令之對應之U位元之值如何，

不針對具有更新之正常整數載入或儲存指令存取擴充部分。

在接下來之實例中，將假定針對SIMD載入或儲存指令，若 $U=1$ ，則將存取 rA 之擴充部分以確定如何執行 rA 之主要部分之位址更新。(另一選擇係，應注意，若 $U=1$ ，則可設計處理器14以使得存取 rB 之擴充部分而非 rA 之擴充部分以確定如何執行 rA 之主要部分之位址更新。)應注意，藉由使用該擴充部分，不需要獨立地尋址或存取一單獨GPR或位址位置以提供例如模式或位址更新控制資訊之額外資訊。此外，不需要指令內之額外欄位，此乃因該指令已指示 rA 來用於提供基底位址值之目的。然而，由於僅需要 rA 之主要部分來存取基底位址值，因此擴充部分可用以提供額外資訊。應注意，無論 U 或模式欄位之值如何，用於向量載入或儲存指令之有效位址計算保持不變。亦即，藉由將 rA 之內容($rA[32:63]$)添加至 rB 之內容($rB[32:63]$)來計算用於 $evldh[u]x$ 或 $evstdh[u]x$ 之EA。亦即， $EA[0:31]=$ 「 rA 之內容+ rB 之內容」。同樣，應注意，在所圖解闡釋之其中假定處理器14係一32位元處理器之實例中，EA係一32位元值。亦可將EA稱作一記憶體存取位址，此乃因其提供將向其儲存資料或將自其中載入資料之記憶體位址。

圖5中圖解闡釋可用於執行一 $evldh[u]x$ 指令(或一 $evstdh[u]x$ 指令)來實施循環緩衝尋址之 rA 之一主要部分及一擴充部分之一實例。如以上所論述， rA 之主要部分包含

基底位址作為記憶體運算元。rA之擴充部分內之模式欄位係%001，其指示一循環緩衝尋址技術，且位址更新控制欄位包含用於實施該循環緩衝尋址之額外資訊。在所圖解闡釋之實施例中，位址更新控制欄位包含位元3:7中之一5位元模數(Mod)欄位，位元8:16中之一8位元偏移(O)欄位及位元16:31中之一16位元長度(L)欄位。應注意，替代實施例可不同地將資訊組織在rA之擴充部分內，或可包含除位址更新控制欄位中修改資訊中之任何資訊以外或代替該任何資訊之其他類型之資訊。L欄位提供循環緩衝器中元素之數目。O欄位提供用於更新計算之距下一元素之距離。Mod欄位指示記憶體中循環緩衝器之開始位址被對準至記憶體中之一 2^{Mod} 位元組邊界。應注意，Mod欄位亦界定可允許循環緩衝器之最大長度。如將自以下看出，此允許一更快環繞計算。

圖6圖解闡釋記憶體12中儲存一循環緩衝器之一部分。(應注意，可將圖6之循環緩衝器稱作一資料結構且其可在被存取之前形成於記憶體12內。)將圖6中之循環緩衝器之開始位址(亦即，基底位址)提供為%10000。循環緩衝器之長度係11(指示11個位元組)，且偏移(亦即，所存取之元素之間的距離)係4個位元組。因此，應注意，緩衝器元素被編號1-11，其指示欲根據基底位址%10000及偏移值4存取該等緩衝器元素之次序。亦即，首先存取位置%1000，繼之以位置%10100，繼之以位置%11000，繼之以位置%10001(此包含一環繞)，繼之以位置%10101，繼之以位

置%11001，繼之以位置%10010(此同樣包含一環繞)，繼之以位置%10110，繼之以位置%11010，繼之以位置%10011(此亦包含一環繞)，繼之以位置%10111等等。

因此，在執行一其中模式欄位係%001之evldhux指令(U=1)後，基底位址係%10000(在此實例中，假定索引值係0)，且偏移=4，rD載入有由%10000尋址之記憶體位置之內容。同樣，欲作為evldhux之執行之部分將rA之主要部分更新為%10100以使得不需要一單獨指令來將循環緩衝器之存取點提前至位於4個位元組遠之下一元素。(此時，可執行使用rD中之結果之一個或多個額外指令。)亦應注意，僅更新rA之主要部分，而rA之擴充部分(其中包含模式欄位及位址更新欄位)保持不變。因此，在執行一隨後evldhux指令後，使用rA、rD之已更新版本被更新有由%10100尋址之記憶體位置之內容，且同樣將rA之主要部分自%10100更新為%11000。由於未更新rA之擴充部分，因此仍將長度提供為11且將偏移提供為4。在執行一隨後evldhux指令後，將4添加至%11000(其係%11110)將導致位址超過緩衝器之最後元素(%11010)。此可使用位址之最低4個有效位元確定，此乃因已知對於此實例而言循環緩衝器之開始位址被對準至%10000($2^{\text{Mod}}=2^4=\%10000$)且長度係11。在此情況下，可自該位址減去緩衝器之長度以產生%10001之下一元素位址。然而，由於已知緩衝器位址之開始元素係%10000(由於mod值，例如 $2^{\text{Mod}}=2^4=\%10000$ ，其在一16位元組邊界上被對準)，該mod值可用以確定在一

環繞後將更新開始緩衝器位址之哪一部分。亦即，在一環繞之情況下， $rA[(64-Mod) : 63] = (rA[48:63] + \text{偏移})[(64-Mod) : 63]$ 其中在一更新後僅修改位元(64-Mod) : 63。因此，rD被更新有由%11000尋址之記憶體位置之內容且將rA之主要部分自%11000更新為%10001(且由於環繞而不係%11110)。應注意，mod值亦提供緩衝器之最大指數為 $2^{(Mod)} - 1$ 。亦即，為使用mod值來執行環繞計算，不可將最後循環緩衝器元素置為高於%11111。

圖5中圖解闡釋可用於執行一evldh[u]x指令(或一evstdh[u]x指令)以實施位元反轉尋址之rA之一主要部分及一擴充部分之一實例。如以上所論述，rA之主要部分包含基底位址作為位址運算元。rA之擴充部分內之模式欄位係%010，其指示一位元反轉尋址技術，且位址更新控制欄位包含用於實施位元反轉尋址之額外資訊。在所圖解闡釋之實施例中，該位址更新控制欄位包含位元16:31中之一16位元遮罩欄位。應注意，替代實施例可不同地將資訊組織在rA之擴充部分內，或可包含除位址更新控制欄位中修改資訊中之任何資訊以外或代替該修任何資訊之其他類型之資訊。遮罩欄位提供一係用以執行位元反轉遞增之遮罩值。

諸多類型之過濾演算法利用緩衝器來保持各組輸入樣本及自一組過濾作業(例如，FFT過濾器)所計算之輸出樣本。通常以一位元反轉方式存取此等過濾器以獲得資料且以一對應於自然計算次序之預定次序儲存輸出。舉例而

言，對於一具有以一線性次序被儲存之元素0、1、2、3、4、5、6及7之8元素FFT緩衝器而言，該等元素需要被存取之位元反轉次序係元素0、4、2、6、1、5、3及7。圖8圖解闡釋當以一位元反轉方式存取資料元素時(例如，當自一儲存於記憶體12中之緩衝器存取FFT資料元素時)可使用之位元反轉尋址之一實例。在圖8之實例中，假定8個資料元素(元素0-7)皆被以線性次序儲存。可藉由一表示值0至7之3位元二進制位址來尋址此等元素中之每一者。此等二進制值%000、%001、%010、%011、%100、%101、%110及%111分別對應於元素0-7中之每一者。然後，將此等二進制值「位元反轉」以導致每一3位元二進制值之鏡像。在反轉對此等二進制值中之每一者進行位元反轉後，經位元反轉之元素導致元素0-7分別被以元素0、元素4、元素2、元素6、元素1、元素5、元素3及元素7之位元反轉次序尋址。應注意，為便於解釋，本文中將闡述之實例假定欲以一位元反轉方式存取8個資料元素之一緩衝器；然而，在替代實施例中，可以一位元反轉次序尋址任一數目資料元素之一緩衝器。

基於一緩衝器中若干資料元素或樣本(例如，一FFT中點之數目)以及一樣本之資料大小之遮罩值係用以確定一欲被以一位元反轉方式存取之下一元素。可使用已知之使用遮罩值來確定欲被以一位元反轉方式存取之下一元素之方法。圖9圖解闡釋一提供基於資料樣本數目及資料樣本大小兩者之各種遮罩值之表格之一實例。為存取一欲被藉助

位元反轉尋址存取之含有 n 個位元組大小之資料元素之緩衝器，該遮罩值在最低有效位元位置處具有 $\log_2 n$ 個一(1)且在剩餘之最高有效位元位置處具有 $\log_2 n$ 個零(0)。然而，若資料大小係一半字或一字之倍數，則構造遮罩以使得1向左移位 \log_2 (位元組中之資料大小)且0位於最低有效位元位置處。應注意，對於表格之每一列而言，隨著遮罩值自位元組行前進至雙字行，該等遮罩值每次向左移位一個位元。同樣，應注意，對於表格之每一行而言，隨著遮罩值自8個元素之大小向下前進至64個元素之大小，該等遮罩值將一額外下部次序位元設定為「1」。可將圖9之表格擴充為包含更大之緩衝器大小(其通常係2之冪)且亦可包含更大之資料大小。圖9之表格可儲存於圖1之控制單元28內，或可儲存於系統10內別處。可將圖9之表格儲存為一值表格或可以邏輯閘實施該表格。替代實施例可以各種不同方式儲存或接收圖9之表格之資訊。

圖10圖解闡釋一位元反轉尋址序列，其可使用用以存取圖11之實例性緩衝器之`evldh[u]x`指令達成。在圖11之實例中，緩衝器儲存8個資料元素，其中每一資料元素係一半字大小。(應注意，圖11之緩衝器可用於一FFT應用中，且亦可將其稱作一FFT樣本緩衝器。亦應注意，可將圖11之緩衝器稱作一資料結構且可在被存取之前形成於記憶體12內。)舉例而言，圖11圖解闡釋儲存在位址位置`0xDCAABCF0`處開始之8個半字大小資料元素之記憶體12之一部分。由於每一元素係一半字大小(例如，16個位

元)，因此分別由位址值 0xDCAABCF2、0xDCAABCF4、0xDCAABCF6、0xDCAABCF8、0xDCAABCFA、0xDCAABCFC 及 0xDCAABC FE 以線性次序依序尋址接下來 7 個元素，元素 1-元素 7。(應注意，亦可將此等位址值中之每一者稱作指標值。)若欲以一位元反轉次序存取此等位址值(如圖 8 中所闡述)，則將以以下次序存取此等位址：0xDCAABCF0(對應於元素 0)、0xDCAABCF8(對應於元素 4)、0xDCAABCF4(對應於元素 2)、0xDCAABCFC(對應於元素 6)、0xDCAABCF2(對應於元素 1)、0xDCAABCFA(對應於元素 5)、0xDCAABCF6(對應於元素 3)及 0xDCAABC FE(對應於元素 7)。

因此，在圖 10 中之位元反轉尋址之實例中，在執行 `evldh[u]x` 指令之前(其中此指向記憶體 12 中待被存取之緩衝器之第一元素，元素 0)，將 0xDCAABCF0 之一基底位址值(亦即，初始指標值)置於暫存器 rA 之主要部分中。將 0x00000000 之值置於 rB 中作為索引值。同樣，針對此實例，遮罩值(取自圖 4 之圖表)係「000...00001110」，此乃因該緩衝器含有 8 個半字大小之資料元素。對應於 0x000E 之此遮罩值儲存於 rA 之擴充部分中。因此，在執行一其中 $U=1$ 且模式欄位係 %010 之 `evldhux` 指令時，EA 被計算為 0xDCAABCF0(基底位址+索引值)，且 rD 接收係緩衝器之元素 0 之 `mem[EA]` 之內容。(應注意，在此實例中，由於 rB 中之索引值係 0，因此基底位址值可被直接提供為 EA。)同樣，由於 $U=1$ 且模式欄位係 %010，因此在執行 `evldhux` 指令後，rA 之擴充部分中之位址更新控制欄位(亦即，遮罩

值)係用以確定如何更新rA之主要部分。使用任一已知方法，0x000E之遮罩值係用以計算位元反轉尋址序列中之下一位址(亦即，對應於元素4之0xDCAABCF8)。因此，將rA之主要部分更新為0xDCAABCF8之新基底位址值。應注意，僅更新rA之主要部分以使得遮罩值及模式欄位值保持不變。同樣，此時，可執行一個或多個額外指令，該等額外指令在一隨後資料處理作業中使用rD中之結果(元素0)。在執行一其中U=1且使用相同rA及rB之隨後evldhux指令後，模式值仍係%010且遮罩值仍係0x000E。在此情況下，基底值係以先前evldhux指令更新之0xDCAABCF8。此外，使用已知方法，遮罩值0x000E係用以計算在位元反轉尋址序列中緊跟0xDCAABCF8之下一位址(亦即，對應於元素2之0xDCAABCF4)。因此，將0xDCAABCF4之內容元素2載入至rD中，且將rA之主要部分更新為0xDCAABCF4。以此方式，一系列evldhux指令可用以以一位元反轉次序存取圖11之緩衝器之緩衝器元素，可以該位元反轉次序更新rA之主要部分以反映位元反轉尋址序列中之下一位址(由於U=1)。

應注意，已參照載入來自記憶體之元素闡述了以上之循環緩衝器尋址實例及位元反轉尋址實例；然而，以上方法亦適用於將元素儲存至記憶體中，其中當一evstdh[u]x指令中U=1且模式=%001或%010時，可針對每一尋址方案如以上所闡述更新rA之主要部分以指向欲將一資料元素儲存至其之下一位址位置。

圖 12 中圖解闡釋一位移形式向量載入指令 (evldh[u]) 之一個實施例。在所圖解闡釋之實施例中，evldh[u] 係一 32 位元指令，該指令指示一目的地暫存器 rD、一個源暫存器 rA 及一提供一位移值之立即值欄位。該 evldh[u] 指令允許一欲規定之更新值 (U) 以指示待執行一具有位移之載入還是一具有位移及更新之載入。在 $U=1(\text{evldhu})$ 之情況下，欲作為該指令之作業之部分執行 rA 之更新。在 $U=0(\text{evldh})$ 之情況下，不執行 rA 之更新。該 evldh[u] 指令包含一識別指令之功能或類型之作業碼欄位 (位元 0:5)。該指令內之一目的地欄位 (位元 6:10) 係一識別純量暫存器檔案 34 內之一暫存器 D 之被標記為「rD」之目的地暫存器識別符。該指令內之一源欄位 (位元 11:15) 係一識別純量暫存器檔案 34 內之一暫存器 A 之被標記為「rA」之源暫存器識別符。該指令內之一無符號立即欄位 (UIMM, 位元 16:20) 係一提供一位移值之立即值。該指令亦包含一可用以進一步規定該指令之功能性之子作業碼欄位 (位元 21:31)。應注意，在所圖解闡釋之實施例中，U 作為該指令之子作業碼內之一位元被包含在內 (在所圖解闡釋之實施例中位於位元位置 25 處)。然而，在替代實施例中，可提供是否欲更新之指令作為該指令內之不同子作業碼或作為該指令內之一不同位元。另一選擇係，可實施其他位元長度指令且每一指令欄位之位元大小係實施方案特定。然而，如以上所論述，應注意在諸多指令集架構中，作業碼及子作業碼空間係有限的，且因此額外指令位元長度可係不可用。同樣，在其他實施例

中，可實施額外欄位或其他欄位。另外，可將rD欄位、rA欄位及立即欄位之次序改變為除圖12中所圖解闡釋之序列以外之其他序列。

該evldh[u]指令(evldh[u] rD、rA，UIMM)類似於evldh[u]x指令；然而，該evldh[u]指令不是提供一儲存一用以計算EA之索引值之第二源暫存器rB，而是在指令本身內提供一直接用以計算EA之立即值。類似於evldh[u]x指令，evldh[u]指令將一來自記憶體之雙字載入至目的地暫存器之四個半字中。因此，在所圖解闡釋之其中處理器14係一32位元處理器之實施例中，evldh[u]指令可用以將4個16位元向量元素載入至目的地暫存器中。然而，在執行一evldh[u]指令後，將提供於該指令之立即欄位內之位移值(UIMM)添加至rA(其儲存一基底位址值)之內容以計算一有效位址(例如，EA[0:31])。然後，將位於記憶體12中由EA[0:31]指向之記憶體位置處之雙字載入至rD中。亦即，將記憶體12中由EA指向之半字載入至rD[0:15]中，將記憶體12中由EA+2指向之半字載入至rD[16:31]中，將記憶體12中由EA+4指向之半字載入至rD[32:47]中，且將記憶體12中由EA+6指向之半字載入至rD[48:63]中。

若U=0(亦即，一evldh指令)，則不執行rA之更新。然而，若U=1(亦即，一evldhu指令)，則rA之擴充部分可用以確定如何執行rA之更新。在一個實施例中，若U=1(亦即，一evldhu指令)，則rA之擴充部分及UIMM兩者皆用以確定如何執行rA之更新。舉例而言，在一個實施例中，若

U=1 且 UIMM 不係 0，則無論 rA 之擴充部分中模式欄位之值如何，以與在一具有位移及更新之正常整數載入中相同之方式(類似於以上參照圖 13 所闡述之用於具有指數及更新之正常整數載入所執行之更新)執行一其中將 EA 之值儲存至 rA 之主要部分 (rA[32:63]) 中之正常更新。然而，若 UIMM 值=0 且 U=1，則指示一更新形式，但通常 rA 之值中將不發生改變，此乃因 rA 中 0 至當前值之 UIMM 值之額外值將不提供一經改動之 rA 值。因此，在一個實施例中，指令之變量 U=0、UIMM=0 及 U=1、UIMM=0 產生相同結果。此提供一使用此一指令編碼以指示額外更新控制值欲用於 SIMD 載入或儲存指令(例如，在 rA 之擴充部分中)之機會。因此，若 U=1、UIMM=0 且 rA 之擴充部分中之模式欄位值係 %000，則執行一其中將 EA 之值儲存至 rA 之主要部分 (rA[32:63]) 中之正常更新。然而，若 U=1、UIMM=0 且 rA 之擴充部分中之模式欄位值係 %001 或 %010，則針對 rA 之主要部分分別執行如以上所闡述之一循環緩衝器尋址更新或一位元反轉尋址更新。亦即，在此等情況下(其中模式欄位係 %001 或 %010)，rA 之擴充部分係用以進一步提供控制資訊(例如，位址更新控制欄位)，該控制資訊界定 rA 之更新如何發生以實施不同尋址方案，例如具有環繞之循環緩衝尋址或位元反轉尋址。因此，以上結合圖 4-11 所提供之其中 rA 之擴充部分係用以結合如何更新 rA 之主要部分提供額外修改資訊之說明亦適用於 evldh[u] 指令。

類似地，應注意對應儲存指令 (evstdh[u]x rS、rA, UIMM)

類似於載入指令運作。該 `evstdh[u]` 指令將來自由 `rS` 指示之 GPR 之四個半字儲存至記憶體中由 `EA` 指向之位置(其中儲存基底位址值及 `UIMM`(位移值)之 `rA` 之內容皆用以計算 `EA[0:31]`)。正如載入指令之情況，若 `U=0`(亦即，一 `evstdh` 指令)，則不執行 `rA` 之更新。然而，若 `U=1`(亦即，一 `evstdhu` 指令)，則 `rA` 之擴充部分及 `UIMM` 之值可用以確定如何執行 `rA` 之更新。亦即，當在指令中指示 `UIMM=0` 之一值時，`rA` 之一擴充部分可用以進一步提供控制資訊(亦即，修改資訊)，該控制資訊界定 `rA` 之更新如何發生以實施不同尋址技術，例如具有環繞之循環緩衝尋址或位元反轉尋址。因此，以上提供之關於 `EA` 之計算及 `UIMM` 之使用及一用以確定當 `U=1` 時如何執行一更新之擴充部分之說明適用於位移形式載入指令及位移形式儲存指令兩者。同樣，應注意，除一半字以外可在載入或儲存指令中界定任一元素大小。舉例而言，本文中所提供之說明亦可類似地適用於一分別自記憶體載入 8 個位元組或將 8 個位元組儲存至記憶體之 `evldb[u]` 或 `evstdb[u]x`。該等說明亦可類似地適用於一分別自記憶體載入 2 個字或將 2 個字儲存至記憶體之 `evldw[u]` 或 `evstdw[u]`。

至此應瞭解，已提供用於一資料處理系統中之資料處理指令，該等資料處理指令能夠使用其源暫存器中之一者之以擴充部分以結合如何修改其源暫存器中之一者之一主要部分提供額外資訊。以此方式，不需要額外作業碼空間。此外，在一個實施例中，此等指令內之一個或多個位元

(例如，U位元)可用以確定是否執行源暫存器中之一者之主要部分之更新且確定在計算該新已更新值時是否使用擴充部分中之位址更新控制資訊。在另一實施例中，該指令內一位移欄位之值可用以確定在計算該新已更新值時是否使用擴充部分中之位址更新控制資訊。

應注意，可如以上所論述用以實施循環緩衝尋址或被位元反轉尋址之處理器14內之電路可位於控制單元28內或執行單元32內。已知電路可用以實施以上結合根據一循環緩衝尋址技術或一位元反轉尋址技術確定下一位址所闡述之功能性。應注意，可藉助以一與所圖解闡釋之循環尋址及位元反轉尋址技術類似方式之交替模式編碼指示額外或交替尋址模式或類型。

因實施本發明之裝置大部分係由熟習此項技術者已知之電子組件及電路構成，因此為理解及瞭解本發明之基本概念且為不混淆或擾亂對本發明之教示起見，將不在任一比以上所圖解闡釋之被視為必要之範圍大的範圍內解釋電路細節。

將本文所用之術語「程式(program)」定義為經設計用於在一電腦系統上執行之一指令序列。一程式或電腦程式可包含一次常式、一函數、一程序、一目標方法、一目標實施方案、一可執行應用程式、一小型應用程式、一伺服小型應用程式、一源程式碼、一目標程式碼、一共用程式庫/動態載入程式庫及/或經設計用於在一電腦系統上執行之其它指令序列。

可使用各種不同資訊處理系統實施以上可適用之實施例中之某些實施例。舉例而言，雖然圖1及其論述闡述一例示性資訊處理架構，但呈現此例示性架構僅在論述本發明之各種態樣時提供一有用參考。當然，出於論述之目的已簡化對該架構之說明，且其僅係可根據本發明使用之諸多不同類型之適當架構中之一者。熟習此項技術者將認識到，邏輯區塊之間的邊界僅係闡釋性且替代實施例可合併邏輯區塊或電路元件或將一替代功能性之分解施加於各種邏輯區塊或電路元件上。

因此，應理解，本文中所繪示之架構僅係例示性，且實際上可實施達成相同功能性之諸多其他架構。在一抽象但仍明確意義上，達成相同功能性之任一組件配置皆有效地「關聯」以便達成所期望之功能性。因此，可將本文中經組合以達成一特定功能性之任何兩個組件視為彼此「相關聯」以便達成所期望之功能性，無論架構或中間組件如何。同樣地，亦可將如此關聯之任何兩個組件視為「可運作地連接」或「可運作地耦合」至彼此以達成所期望之功能性。

同樣，舉例而言，在一個實施例中，所圖解闡釋之系統10之元件係位於一單個積體電路上或同一器件內之電路。另一選擇係，資料處理系統10可包含任一數目之彼此互連之單獨積體電路或單獨器件。舉例而言，記憶體12可位於一與處理器14相同之積體電路上或位於一單獨積體電路上或位於與資料處理系統10之其他元件離散地間隔開之另一

周邊或從屬器件內。周邊器件18及I/O電路16亦可位於單獨積體電路或器件上。亦應理解，可以矽或其他半導體材料或另一選擇係藉由矽或其他半導體材料之軟體程式碼表示實施本文中所闡述之所有電路。同樣地，可以任何適當類型之一硬體描述語言體現系統10。軟體程式碼或硬體描述語言可體現在任一類型之電腦可讀媒體上。

此外，熟習此項技術者將認識到以上所闡述作業之功能性之間的邊界僅係闡釋性。在某些實例中，可將多個作業之功能性組合成一單個作業，且/或可將一單個作業之功能性分佈於額外作業中。另外，替代實施例可包括一特定作業之多個實例，且可在各種其他實施例中改動作業之次序。

本文中所闡述之軟體中之所有軟體或某些軟體可係系統10之(舉例而言)自電腦可讀媒體(例如記憶體12)或其他電腦系統上之其他媒體接收之元素。此電腦可讀媒體可永久地、可移除地或遠端地耦合至一例如系統10之資訊處理系統。電腦可讀媒體可包含(舉例而言但不侷限於)任一數目之以下媒體：磁性儲存媒體，其中包含磁碟及磁帶儲存媒體；光學儲存媒體，例如光碟媒體(例如，CD-ROM、CD-R等)及數位視訊磁碟儲存媒體；非揮發性記憶體儲存媒體，其中包含基於半導體之記憶體單元，例如快閃記憶體、EEPROM、EPROM、ROM；鐵磁數位記憶體；MRAM；揮發性儲存媒體，其中包含暫存器、緩衝器或快取記憶體、主記憶體、RAM等；及資料傳輸媒體，其中包含電腦

網路、點對點電信設備及載波傳輸媒體，僅舉幾個例子。

在一實施例中，資料處理系統10係一例如一個人電腦系統之電腦系統。其他實施例可包含不同類型之電腦系統。電腦系統係可經設計以供給一個或多個使用者獨立計算能力之資訊處置系統。電腦系統可以包含但不侷限於主機、小型電腦、伺服器、工作站、個人電腦、文字編譯器、個人數位助理、電子遊戲機、汽車及其他嵌入式系統、蜂巢式電話及各種其他無線器件等若干形式存在。一典型電腦系統包含至少一個處理單元、關聯記憶體及若干輸入/輸出(I/O)器件。

一電腦系統根據一程式來處理資訊且經由I/O器件產生所得輸出資訊。一程式係一指令列表，例如一特定應用程式及/或一作業系統。一電腦程式通常內部儲存於電腦可讀儲存媒體上或經由一電腦可讀傳輸媒體被傳輸至電腦系統。一電腦處理程序通常包含一執行(運行)程式或一程式之一部分、當前程式值及狀態資訊及由作業系統用來管理該處理程序之執行之資源。一父代處理程序可卵生其他子代處理程序以幫助執行該父代處理程序之總體功能性。具體而言，由於父代處理程序孵育出若干子代處理程序以執行該父代處理程序之總體功能性之一部分，因此有時可將由子代處理程序(及孫代處理程序等)執行之功能闡述為由父代處理程序執行。

雖然本文參照特定實施例闡述了本發明，但在不背離以下申請專利範圍中所陳述之本發明之範疇之情況下可進行

各種修改及改變。舉例而言，除十六進製以外可使用多種編號格式。可使用對資料處理架構之改變。各種已知資料儲存電路中之任一者可用以實施記憶體及暫存器。可端視應用程式改變資料匯流排、暫存器、向量欄位及資料運算元之位元大小。另外，可將用於預定常數之值改變為具有除本文中所圖解闡釋之值以外之其他值。任一類型之半導體處理技術可用以實施本文中所闡述之電路。因此，應從一闡釋性而非一限定性意義上來看待本說明書及圖式，且所有此等修改皆意欲包含於本發明之範疇內。本文中關於特定實施例所闡述之任何益處、優點或問題之解決方案皆非意欲被理解為任何或所有申請專利範圍之關鍵、所需或必須特徵或元素。

如本文中所使用之術語「經耦合(coupled)」並非意欲侷限於一直接耦合或一機械耦合。

此外，將本文中所使用之術語「一(a)」或「一(an)」界定為一個或一個以上。同樣，不應將申請專利範圍中例如「至少一個(at least one)」及「一個或多個(one or more)」之介紹性片語之使用理解為意味著由不定冠詞「一(a)」或「一(an)」介紹之另一申請專利範圍元素將任一含有該所介紹之申請專利範圍元素之特定申請專利範圍侷限於僅含有一個此元素之發明，即使當同一申請專利範圍包含介紹性片語「一或多個(one or more)」或「至少一個(at least one)」及例如「一(a)」或「一(an)」等不定冠詞時亦係如此。此同樣適用於定冠詞之使用。

除非另有說明，否則所使用之例如「第一(first)」及「第二(second)」之術語係用以在此等術語所闡述之多個元素之間任意區分。因此，此等術語未必意欲指示此等元素之時間或其他優先次序。

以下係本發明之各種實施例。

項1包含一種運作一資料處理系統之方法，該方法包含：由一N位元處理器執行一第一指令，其中該第一指令係一具有更新之記憶體存取指令，其中對該第一指令之該執行包含：存取複數個暫存器中之至少一個源暫存器，其中該存取該至少一個源暫存器包含存取該複數個暫存器中之一第一暫存器，其中該複數個暫存器中之每一暫存器包含N個位元之一主要暫存器部分及M個位元之一擴充暫存器部分，其中該第一暫存器之該主要暫存器部分包含一第一位址運算元；使用該第一位址運算元形成一記憶體存取位址；使用該記憶體存取位址作為一用於一記憶體存取之位址；產生一已更新位址運算元，其中該已更新位址運算元係基於該第一位址運算元；及將該已更新位址運算元寫入至該第一暫存器之該主要部分。該產生包含存取該至少一個源暫存器中之一源暫存器之一擴充部分以獲得修改資訊且在該產生一已更新位址運算元過程中使用該修改資訊。

項2包含項1之方法，其中該存取至少一個源暫存器中之一源暫存器之一擴充部分以獲得修改資訊包含存取該第一暫存器之該擴充暫存器部分以獲得該修改資訊。

項3包含項2之方法，其中該第一暫存器係一用於該第一指令之基底暫存器且該第一位址運算元係一基底位址。

項4包含項1之方法，其中該至少一個源暫存器包含該複數個暫存器中之一索引暫存器，其中該存取至少一個源暫存器中之一源暫存器之一擴充部分以獲得修改資訊包含存取該索引暫存器之該擴充暫存器部分以獲得該修改資訊。

項5包含項1之方法，其中該第一指令係用於存取一儲存於一循環緩衝器中之記憶體運算元，且該修改資訊包含該循環緩衝器之緩衝長度之一指令。

項6包含項5之方法，其中該修改資訊包含該循環緩衝器之一模數之一指示。

項7包含項5之方法，其中該修改資訊包含用於存取該循環緩衝器之一偏移值之一指示。

項8包含項1之方法，其中當執行該第一處理器指令時，該第一處理器指令執行一位元反轉遞增作業，且該修改資訊包含用於該位元反轉遞增作業之一遮罩值之一指示。

項9包含項1之方法，其中該執行一第一指令包含確定該第一指令之一偏移欄位指示一第一值，且該執行該第一指令包含回應於指示該第一值之該偏移欄位存取該至少一個源暫存器中之一源暫存器之一擴充部分以獲得修改資訊，其中若具有更新之一記憶體存取指令之一偏移欄位不指示該第一值，則針對具有更新之一記憶體存取指令產生一已更新位址運算元將不包含針對該具有更新之該記憶體存取指令使用來自一源暫存器之一擴充部分之修改資訊。

項10包含項1之方法，其中當執行純量整數指令時不使用該複數個暫存器之該等擴充部分。

項11包含項1之方法，其中提供該複數個暫存器之該等擴充部分以用於儲存SIMD向量元素。

項12包含項1之方法，其中該第一指令係一SIMD指令。

項13包含項1之方法，其中該至少一個源暫存器中之該源暫存器之一擴充部分包含一模式欄位及一控制欄位；該修改資訊位於該控制欄位中；且該產生一已更新位址運算元包含按照複數個更新技術產生一已更新位址運算元，其中該模式欄位指示該複數個技術中之哪一技術將用於該產生。

項14包含項1之方法且進一步包含由該處理器執行一第二指令，其中該第二指令係一具有更新之記憶體存取指令，且其中對該第二指令之該執行包含：存取該第一暫存器之該主要暫存器部分以獲得該已更新位址運算元，使用該已更新位址運算元形成一第二記憶體存取位址，且使用該第二記憶體存取位址作為一用於一第二記憶體存取之位址。

項15包含項1之方法，其中該形成該記憶體存取位址包含對該第一位址運算元添加有一第一值以推導出該記憶體存取位址，且該產生一已更新位址運算元包含使用該修改資訊修改該記憶體存取位址。

項16包含一種處理器，該處理器包含：複數個暫存器，該複數個暫存器中之每一者包含一主要部分及一擴充部

分，其中該主要部分係N個位元且該擴充部分係M個位元，其中該處理器係一N位元處理器；一執行單元，該執行單元用於執行指令，其中在由該執行單元對指令之該執行時，該複數個暫存器中之若干暫存器用作源暫存器及目的地暫存器；且其中該執行單元存取該複數個暫存器中之至少一個源暫存器以用於執行一具有更新之第一記憶體存取指令，其中該執行單元使用來自該至少一個源暫存器中之一源暫存器之一擴充部分之修改資料以更新來自該至少一個源暫存器中之一第一源暫存器之一主要部分之一第一位址運算元以產生一已更新位址運算元，其中將該已更新位址運算元寫入至該第一源暫存器之該主要部分。

項17包含項16之處理器，其中該執行單元使用來自該第一源暫存器之一擴充部分之修改資料以更新該第一位址運算元。

項18包含項16之處理器，其中該第一記憶體存取指令係用於存取儲存於一循環緩衝器中之一記憶體運算元，且該修改資訊包含該循環緩衝器之緩衝長度之一指示。

項19包含項16之處理器，其中當執行該第一記憶體存取指令時，該第一記憶體存取指令執行一位元反轉遞增作業，且該修改資訊包含用於該位元反轉遞增作業之一遮罩值之一指示。

項20包含項16之處理器，其中在由該執行單元執行純量整數指令時不使用該複數個暫存器之該等擴充部分，且提供該複數個暫存器之該等擴充部分以用於儲存SIMD向量

元素。

項20包含項16之處理器，其中M等於N。

【圖式簡單說明】

本發明以實例方式予以圖解闡釋且不侷限於隨附圖式，在該等隨附圖式中相同參考編號指示相同元件。為簡化及清晰地圖解闡釋起見，圖式中之元件未必按比例繪製。

圖1以方塊圖形式圖解闡釋一根據本發明之一種形式之一SIMD資料處理系統；

圖2以方塊圖形式圖解闡釋根據本發明之形式之圖1之SIMD資料處理系統之純量通用暫存器檔案；

圖3圖解闡釋根據本發明之一個實施例用於圖1之SIMD資料處理系統之執行之一種形式之一指數形式載入向量指令；

圖4以方塊圖形式圖解闡釋根據本發明之一個實施例之圖2之純量通用暫存器檔案之一通用暫存器；

圖5以方塊圖形式圖解闡釋根據本發明之一個實施例之圖2之純量通用暫存器檔案之一通用暫存器；

圖6以方塊圖形式圖解闡釋圖1之記憶體之一部分以圖解闡釋循環緩衝器尋址之一實例；

圖7以方塊圖形式圖解闡釋根據本發明之一個實施例之圖2之純量通用暫存器檔案之一通用暫存器；

圖8圖解闡釋位元反轉尋址之一實例；

圖9圖解闡釋用於不同資料大小及資料數目之遮罩之實例值之一表格，該等值可用於確定位元反轉尋址增量；

圖 10 以圖形式圖解闡釋根據本發明之一個實施例之一實例性基底位址指標值、一實例性索引值、一實例性遮罩值及一實例性位元反轉尋址序列；

圖 11 以圖形式圖解闡釋圖 1 之 SIMD 資料處理系統之儲存一快速傅立葉轉換 (FFT) 緩衝器之資料元素之記憶體之一部分；

圖 12 圖解闡釋根據本發明之一個實施例用於圖 1 之 SIMD 資料處理系統之執行之一種形式之一位移形式載入向量指令；及

圖 13 圖解闡釋用於圖 1 之處理器之執行之一種形式之一先前技術非 SIMD 指數形式載入指令。

【主要元件符號說明】

| | |
|----|--------|
| 10 | 資料處理系統 |
| 12 | 記憶體 |
| 14 | 處理器 |
| 16 | 輸入/輸出 |
| 18 | 其他周邊器件 |
| 20 | 系統匯流排 |
| 22 | 導體 |
| 24 | 導體 |
| 26 | 導體 |
| 28 | 控制單元 |
| 30 | 指令單元 |
| 32 | 執行單元 |

| | |
|----|-----------|
| 34 | 純量通用暫存器檔案 |
| 36 | 匯流排介面單元 |
| 38 | 載入/儲存單元 |
| 40 | 導體 |
| 42 | 導體 |
| 46 | 導體 |
| 48 | 導體 |
| 50 | 導體 |
| 52 | 導體 |
| 54 | 導體 |
| 56 | 導體 |
| 58 | 導體 |
| 60 | 導體 |
| 62 | 導體 |

發明專利說明書

(本說明書格式、順序及粗體字，請勿任意更動，※記號部分請勿填寫)

※申請案號： 98119101

※申請日： 98.6.8

※IPC 分類： ~~G06F~~

G06F 15/80

(2006.01)

一、發明名稱：(中文/英文)

在單一指令多資料之資料處理器中提供擴充尋址模式

PROVISION OF EXTENDED ADDRESSING MODES IN A SINGLE INSTRUCTION MULTIPLE DATA (SIMD) DATA PROCESSOR

二、中文發明摘要：

由一N位元處理器(14)執行一具有更新之第一記憶體存取指令包含存取複數個暫存器(34)中之至少一個源暫存器，其中該存取包含存取一第一暫存器，其中該複數個暫存器中之每一暫存器包含一N個位元之主要部分及一M個位元之擴充部分，其中該第一暫存器之該主要部分包含一第一位址運算元。該第一指令之該執行進一步包含：使用該第一位址運算元形成一記憶體存取位址；使用該記憶體存取位址作為一用於一記憶體存取之位址；產生一已更新之位址運算元；及將該已更新之位址運算元寫入至該第一暫存器之該主要部分。該產生包含存取該至少一個源暫存器中之一源暫存器之一擴充部分以獲得修改資訊且在該產生一已更新之位址運算元中使用該修改資訊。

三、英文發明摘要：

Executing a first memory access instruction with update by an N-bit processor (14) includes accessing at least one source register of a plurality of registers (34), wherein the accessing includes accessing a first register, wherein each register of the plurality of registers includes a main portion of N bits and an extension portion of M bits, wherein the main portion of the first register includes a first address operand. The execution of the first instruction further includes forming a memory access address using the first address operand; using the memory access address as an address for a memory access; producing an updated address operand; and writing the updated address operand to the main portion of the first register. The producing includes accessing an extension portion of a source register of the at least one source register to obtain modifying information and using the modifying information in the producing an updated address operand.

七、申請專利範圍：

1. 一種運作一資料處理系統之方法，其包括：

藉由一N位元處理器執行一第一指令，其中該第一指令係一具有更新之記憶體存取指令，其中該第一指令之該執行包含：

存取複數個暫存器中之至少一個源暫存器，其中該存取該至少一個源暫存器包含存取該複數個暫存器中之一第一暫存器，其中該複數個暫存器中之每一暫存器包含N個位元之一主要暫存器部分及M個位元之一擴充暫存器部分，其中該第一暫存器之該主要暫存器部分包含一第一位址運算元；

使用該第一位址運算元形成一記憶體存取位址；

使用該記憶體存取位址作為一用於一記憶體存取之位址；

產生一已更新位址運算元，其中該已更新位址運算元係基於該第一位址運算元；

將該已更新位址運算元寫入至該第一暫存器之該主要部分；

其中該產生包含存取該至少一個源暫存器中之一源暫存器之一擴充部分以獲得修改資訊且在該產生一已更新位址運算元中使用該修改資訊。

2. 如請求項1之方法，其中該存取至少一個源暫存器中之一源暫存器之一擴充部分以獲得修改資訊包含存取該第一暫存器之該擴充暫存器部分以獲得該修改資訊。

3. 如請求項2之方法，其中該第一暫存器係一用於該第一指令之基底暫存器且該第一位址運算元係一基底位址。
4. 如請求項1之方法，其中：

該至少一個源暫存器包含該複數個暫存器中之一索引暫存器，其中該存取至少一個源暫存器中之一源暫存器之一擴充部分以獲得修改資訊包含存取該索引暫存器之該擴充暫存器部分以獲得該修改資訊。
5. 如請求項1之方法，其中：

該第一指令係用於存取一儲存於一循環緩衝器中之記憶體運算元；且

該修改資訊包含該循環緩衝器之緩衝長度之一指示。
6. 如請求項5之方法，其中該修改資訊包含該循環緩衝器之一模數之一指示。
7. 如請求項5之方法，其中該修改資訊包含用於存取該循環緩衝器之一偏移值之一指示。
8. 如請求項1之方法，其中：

當執行該第一處理器指令時，該第一處理器指令執行一位元反轉遞增作業；

該修改資訊包含用於該位元反轉遞增作業之一遮罩值之一指示。
9. 如請求項1之方法，其中：

該執行一第一指令包含確定該第一指令之一偏移欄位指示一第一值；

該執行該第一指令包含回應於指示該第一值之該偏移

欄位存取該至少一個源暫存器中之一源暫存器之一擴充部分以獲得修改資訊；

其中若具有更新之一記憶體存取指令之一偏移欄位不指示該第一值，則針對一具有更新之記憶體存取指令產生一已更新位址運算元將不包含針對該具有更新之記憶體存取指令使用來自一源暫存器之一擴充部分之修改資訊。

10. 如請求項1之方法，其中在執行純量整數指令時不使用該複數個暫存器之該等擴充部分。

11. 如請求項1之方法，其中提供該複數個暫存器之該等擴充部分用於儲存SIMD向量元素。

12. 如請求項1之方法，其中該第一指令係一SIMD指令。

13. 如請求項1之方法，其中：

該至少一個源暫存器中之該源暫存器之一擴充部分包含一模式欄位及一控制欄位；

其中該修改資訊位於該控制欄位中；

其中該產生一已更新位址運算元包含按照複數個更新技術產生一已更新位址運算元，其中該模式欄位指示該複數個技術中之哪一技術將用於該產生。

14. 如請求項1之方法，其進一步包括：

由該處理器執行一第二指令，其中該第二指令係一具有更新之記憶體存取指令，其中對該第二指令之該執行包含：

存取該第一暫存器之該主要暫存器部分以獲得該已

更新位址運算元；

使用該已更新位址運算元形成一第二記憶體存取位址；

使用該第二記憶體存取位址作為一用於一第二記憶體存取之位址。

15. 如請求項1之方法，其中：

該形成該記憶體存取位址包含對該第一位址運算元添加有一第一值以推導出該記憶體存取位址；

該產生一已更新位址運算元包含使用該修改資訊修改該記憶體存取位址。

16. 一種處理器，其包括：

複數個暫存器，該複數個暫存器中之每一者包含一主要部分及一擴充部分，其中該主要部分係N個位元且該擴充部分係M個位元，其中該處理器係一N位元處理器；

一執行單元，該執行單元用於執行指令，其中在由該執行單元對指令之該執行中，該複數個暫存器中之若干暫存器用作源暫存器及目的地暫存器；

其中該執行單元存取該複數個暫存器中之至少一個源暫存器以用於執行一具有更新之第一記憶體存取指令，其中該執行單元使用來自該至少一個源暫存器中之一源暫存器之一擴充部分之修改資料以更新來自該至少一個源暫存器中之一第一源暫存器之一主要部分之一第一位址運算元以產生一已更新位址運算元，其中將該已更新

位址運算元寫入至該第一源暫存器之該主要部分。

17. 如請求項16之處理器，其中該執行單元使用來自該第一源暫存器之一擴充部分之修改之資料以更新該第一位址運算元。

18. 如請求項16之處理器，其中：

該第一記憶體存取指令係用於存取儲存於一循環緩衝器中之一記憶體運算元；

該修改資訊包含該循環緩衝器之緩衝長度之一指示。

19. 如請求項16之處理器，其中：

當執行該第一記憶體存取指令時，該第一記憶體存取指令執行一位元反轉遞增作業；

該修改資訊包含用於該位元反轉遞增作業之一遮罩值之一指示。

20. 如請求項16之處理器，其中：

在由該執行單元執行純量整數指令時不使用該複數個暫存器之該等擴充部分；

提供該複數個暫存器之該等擴充部分以用於儲存SIMD向量元素。

21. 如請求項16之處理器，其中M等於N。

八、圖式：

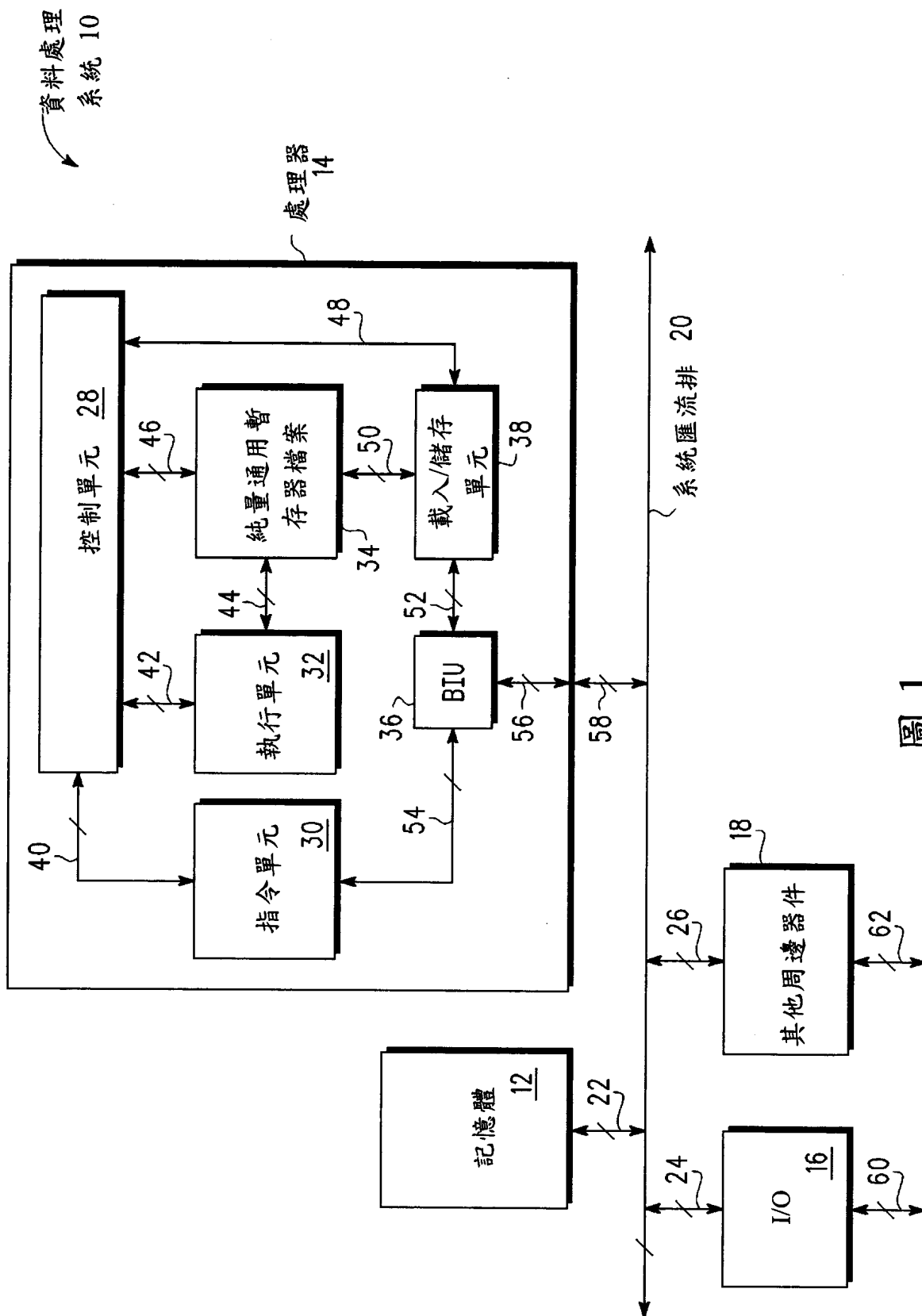


圖 1

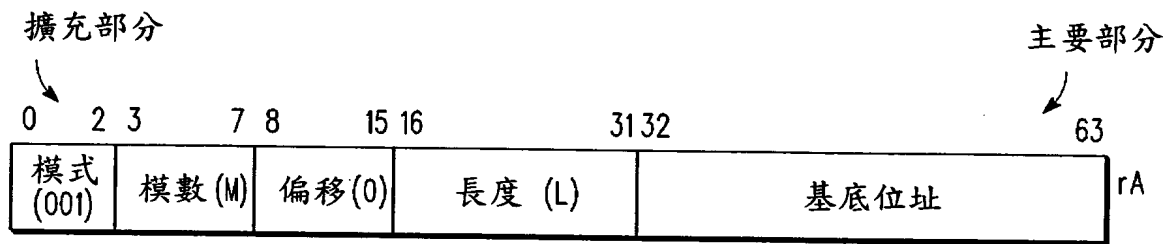


圖 5

記憶體之
部分12

長度 = 11
基底位址 = %10000
偏移 = 4
模數 = 4 (最大長度 = 2⁴)

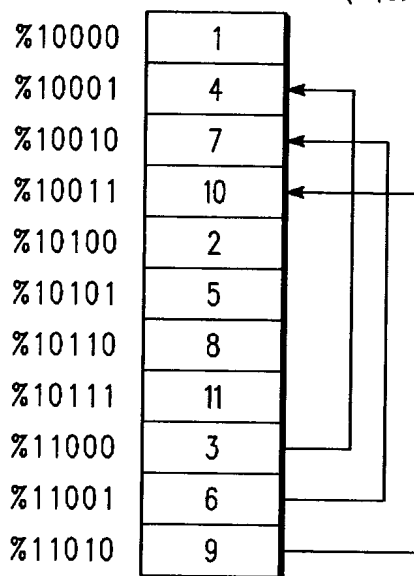


圖 6

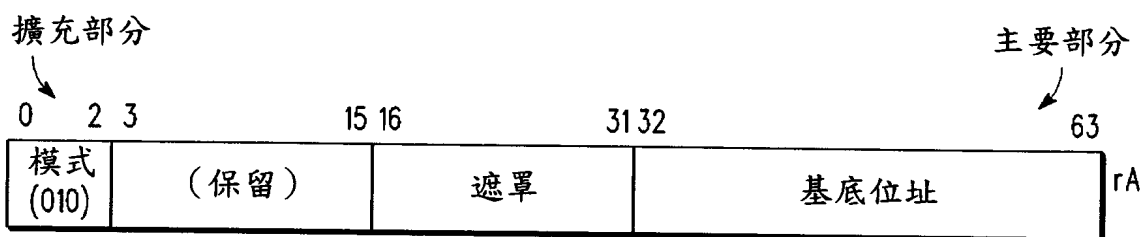


圖 7

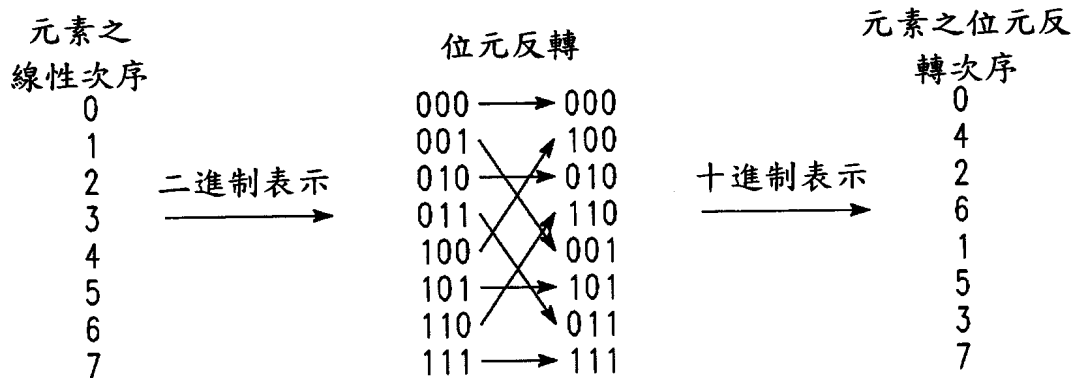
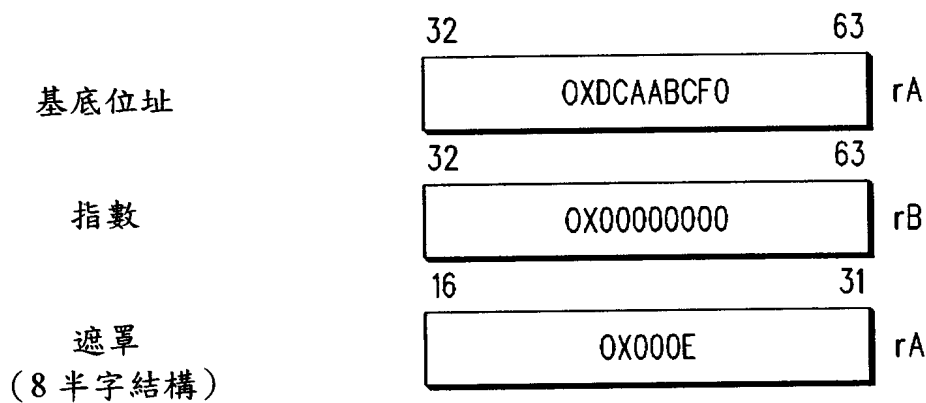


圖 8

| 資料 樣本數目 | 資料大小 | | | |
|------------|----------------|----------------|-----------------|------------------|
| | 位元組 | 半字 | 字 | 雙字 |
| 8 | 000...00000111 | 000...00001110 | 000...000011100 | 000...0000111000 |
| 16 | 000...00001111 | 000...00011110 | 000...000111100 | 000...0001111000 |
| 32 | 000...00011111 | 000...00111110 | 000...001111100 | 000...0011111000 |
| 64 | 000...00111111 | 000...01111110 | 000...011111100 | 000...0111111000 |

圖 9



尋址序列:

- OXDCAABCF0
- 已更新 rA[32:63] → OXDCAABCF8
- 已更新 rA[32:63] → OXDCAABCF4
- 已更新 rA[32:63] → OXDCAABCFC
- 已更新 rA[32:63] → OXDCAABCF2
- 已更新 rA[32:63] → OXDCAABCF A
- 已更新 rA[32:63] → OXDCAABCF6
- 已更新 rA[32:63] → OXDCAABCFE

圖 10

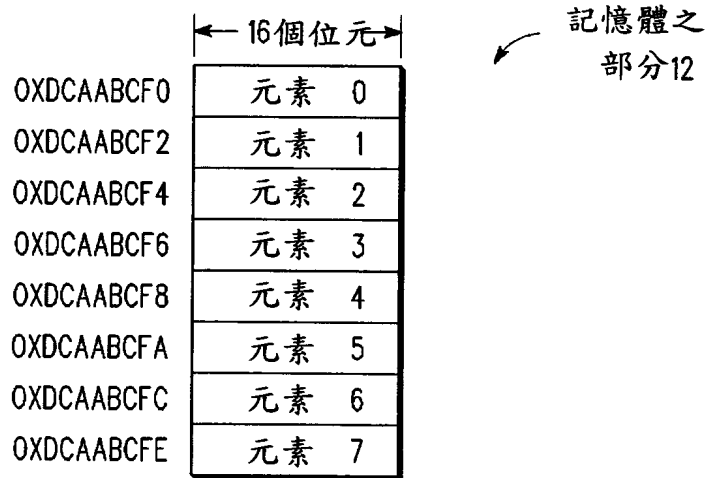


圖 11

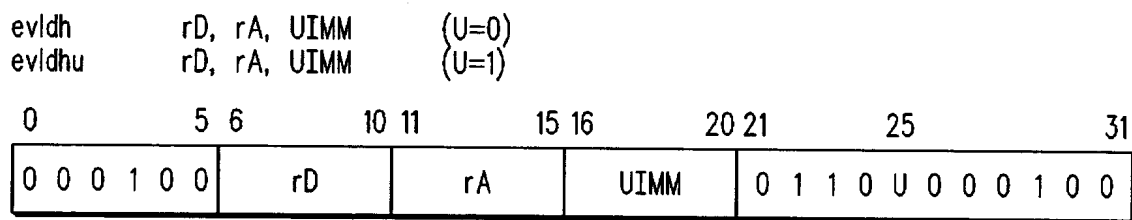


圖 12

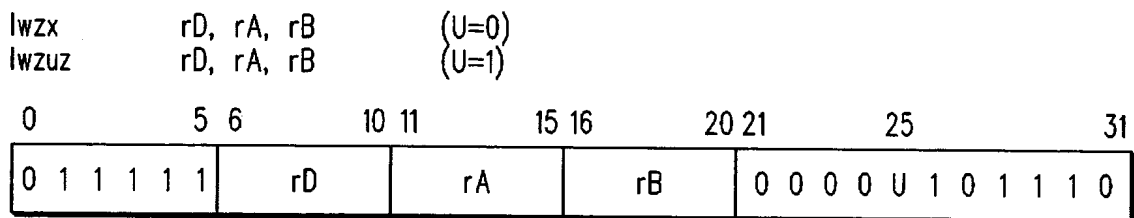


圖 13

先前技術

四、指定代表圖：

(一)本案指定代表圖為：第(2)圖。

(二)本代表圖之元件符號簡單說明：

34 暫存器/純量通用暫存器檔案

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)