



(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 I604726 B

(45)公告日：中華民國 106 (2017) 年 11 月 01 日

(21)申請案號：102128080

(22)申請日：中華民國 102 (2013) 年 08 月 06 日

(51)Int. Cl. : H04N21/2315(2011.01)

H03M13/27 (2006.01)

(30)優先權：2012/08/30 英國

1215425.8

(71)申請人：想像力科技有限公司 (英國) IMAGINATION TECHNOLOGIES LIMITED (GB)
英國(72)發明人：穆林 保羅 MURRIN, PAUL (GB)；安德森 艾德瑞恩 J ANDERSON, ADRIAN
JOHN (GB)；艾爾 哈賈爾 默罕梅德 EL-HAJJAR, MOHAMMED (LB)

(74)代理人：憐軼群；陳文郎

(56)參考文獻：

US 2009/0235020A1

US 2009/0313399A1

審查人員：張長軾

申請專利範圍項數：20 項 圖式數：10 共 55 頁

(54)名稱

用於數位信號處理之以瓦狀塊為基礎的交插及解交插技術

TILE BASED INTERLEAVING AND DE-INTERLEAVING FOR DIGITAL SIGNAL PROCESSING

(57)摘要

本案描述列-行交插資料的以瓦狀塊為基礎的交插及解交插。在一個實例中，解交插被分為兩個記憶體轉移階段，第一轉移階段自一晶載記憶體至一 DRAM，且第二轉移階段自該 DRAM 至一晶載記憶體。每一階段對資料的列-行交插區塊的一部分進行操作且對資料項重新排序，使得第二階段之輸出包含解交插資料。在第一階段中，根據記憶體讀取位址的一非線性序列自該晶載記憶體讀取資料項且將其寫入至該 DRAM。在第二階段中，根據有效率地使用 DRAM 介面之線性位址序列之叢發自該 DRAM 讀取資料項且根據記憶體寫入位址的一非線性序列將其寫回至晶載記憶體。

Tile based interleaving and de-interleaving of row-column interleaved data is described. In one example, the de-interleaving is divided into two memory transfer stages, the first from an on-chip memory to a DRAM and the second from the DRAM to an on-chip memory. Each stage operates on part of a row-column interleaved block of data and re-orders the data items, such that the output of the second stage comprises de-interleaved data. In the first stage, data items are read from the on-chip memory according to a non-linear sequence of memory read addresses and written to the DRAM. In the second stage, data items are read from the DRAM according to bursts of linear address sequences which make efficient use of the DRAM interface and written back to on-chip memory according to a non-linear sequence of memory write addresses.

指定代表圖：

- 符號簡單說明：
- 102 . . . 晶載記憶體/記憶裝置
 - 112 . . . DRAM
 - 502 . . . 第一示意圖
 - 506 . . . 第二示意圖
 - 521~524 . . . 箭頭
 - 561~564 . . . 箭頭

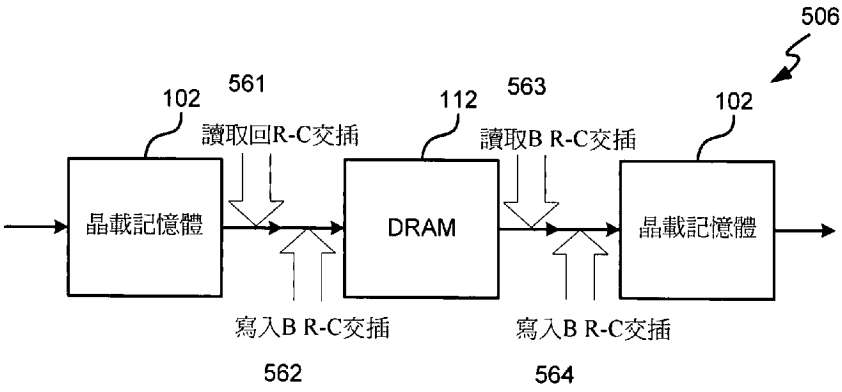
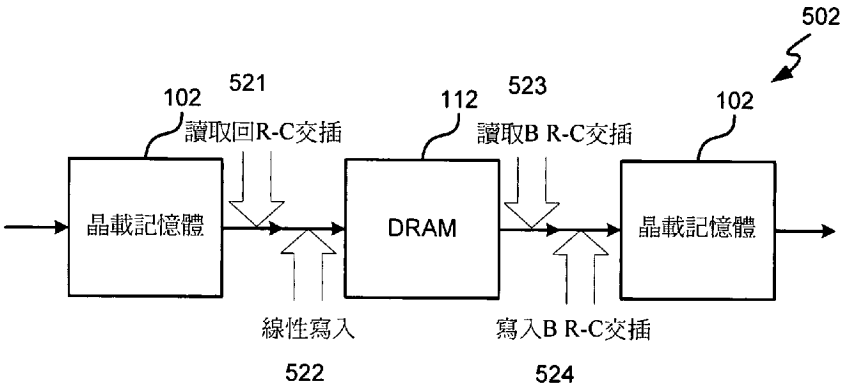


圖5

發明專利說明書

(本說明書格式、順序，請勿任意更動)

【發明名稱】(中文/英文)

用於數位信號處理之以瓦狀塊為基礎的交插及解交插技術

TILE BASED INTERLEAVING AND DE-INTERLEAVING
FOR DIGITAL SIGNAL PROCESSING

【技術領域】

[0001]本發明係有關於用於數位信號處理之以瓦狀塊為基礎的交插及解交插技術。

【先前技術】

發明背景

[0002]數位信號處理用於廣泛多種應用中。許多此等應用在以下意義上係即時的：資料的處理存在時間限制，以使資料對終端使用者有意義或有用。此應用的實例係數位廣播串流，諸如數位電視及數位無線電。數位信號處理系統需要能夠足夠快地處理且解碼即時串流，以允許資料輸出與接收一樣快(不包括緩衝)。

[0003]數位信號處理系統除使用更多通用數位信號處理器之外還通常使用一或多個專用硬體周邊。硬體周邊係經設計來以快速及有效率的方式執行特定信號處理任務之處理區塊。例如，交插及解交插係通常使用硬體周邊對即時資料執行的操作。交插及解交插係需要大量記憶體的操作，且執行此操作的硬體周邊使用相關聯的專用記憶裝置來對資料重新排序。

[0004]然而，不同類型之即時資料的要求可顯著地不同。例如，全世界所使用之各種不同數位電視及無線電標準通常具有以不同方式結構化之即時資料，例如，針對編碼、交插、等化等使用不同類型或參數。若數位信號處理系統足夠靈活以用於不同標準，則用於交插/解交插之專用記憶裝置必須足夠大以應對具有最大記憶體需求的標準。結果，用於交插/解交插硬體周邊之記憶體經常未被充分利用。

[0005]記憶裝置的實例係 DRAM(動態隨機存取記憶體)裝置。DRAM 裝置按頁來組織其儲存內容，每頁的大小通常為幾千位元組。每個 DRAM 一次僅能使有限數目個頁(通常四個)打開，且需要許多附加項循環來打開頁以便取用資料。

[0006]以下描述的實施例不限於解決已知數位信號處理系統的劣勢中之任一個或全部之實行方案。

【發明內容】

發明概要

[0007]提供此發明概述來以簡化形式引入在以下詳細描述中進一步描述的概念精選。此發明概述不意欲識別所主張之標的物的關鍵特徵或必需特徵，亦不意欲用來幫助確定所主張之標的物之範疇。

[0008]描述列-行交插資料的以瓦狀塊為基礎的交插及解交插。在一個實例中，解交插被分為兩個記憶體轉移階段，第一階段自晶載記憶體至 DRAM，且第二階段自 DRAM

至晶載記憶體。每一階段對資料的列-行交插區塊的一部分進行操作且對資料項重新排序，使得第二階段之輸出包含解交插資料。在第一階段中，根據記憶體讀取位址的非線性序列自晶載記憶體讀取資料項且將其寫入至 DRAM。在第二階段中，根據有效率地使用 DRAM 介面之線性位址序列之叢發自 DRAM 讀取資料項且根據記憶體寫入位址的非線性序列將其寫回至晶載記憶體。

[0009]第一態樣提供一種數位信號處理系統單晶片 (system-on-chip)，該數位信號處理系統單晶片包含：第一記憶體，其儲存以第一序列來配置的多個資料項，每一資料項在第一記憶體上具有相關聯的記憶體位址，且多個資料項包含資料項區塊之子集；第二記憶體；以及轉移引擎，其耦接至第一記憶體及第二記憶體且包含埠，該埠接至動態隨機存取記憶體 DRAM，其中該轉移引擎經組配來在第二記憶體轉移階段中將多個資料項自 DRAM 直接轉移至第二記憶體，且其中在第一記憶體轉移階段中，該轉移引擎經配置來根據記憶體讀取位址的預定非線性序列自第一記憶體讀取多個資料項且將多個資料項寫入至 DRAM，且其中在第二記憶體轉移階段中，該轉移引擎經配置來根據線性位址序列之叢發自 DRAM 讀取多個資料項，線性位址序列之每一叢發具有基於 DRAM 介面叢發大小來選擇的長度，且根據記憶體寫入位址的預定非線性序列將多個資料項寫入至第二記憶體，使得多個資料項在第二記憶體上以第二序列來配置，第二序列不同於第一序列，且其中第一

序列及第二序列中之一者包含列-行交插資料。

[0010]第二個態樣提供一種在數位信號處理系統中對資料項區塊執行交插或解交插操作的方法，該方法包含：根據記憶體讀取位址的預定非線性序列自第一晶載記憶體讀取以第一序列儲存的第一多個資料項，其中第一多個資料項包含資料項區塊之子集；將第一多個資料項寫入至動態隨機存取記憶體 DRAM；根據線性位址序列之叢發自 DRAM 讀取第一多個資料項，線性位址序列之每一叢發具有基於 DRAM 介面叢發大小來選擇的長度；以及根據記憶體寫入位址的預定非線性序列將第一多個資料項寫入至第二晶載記憶體，使得資料項在第二晶載記憶體上以第二序列來配置，第二序列不同於第一序列，且其中第一序列及第二序列中之一者包含列-行交插資料。

[0011]第三態樣提供一種電腦程式，該電腦程式包含電腦程式碼構件，當該程式在電腦上運行時，該電腦程式碼構件適於執行上述方法中之任一者之所有步驟。該電腦程式可實施在電腦可讀媒體上。

[0012]第四態樣提供一種執行交插或解交插操作的方法，大致如參照圖式之圖 5 至圖 10 中之任一者所描述。

[0013]本文描述之方法可由電腦執行，該電腦係由以機器可讀形式儲存於有形儲存媒體上之軟體來組配，例如以電腦程式之形式，其中該電腦程式包含用於組配電腦來執行所描述方法之組成部分之電腦程式碼。有形(或非暫時性)儲存媒體之實例包括碟片、拇指隨身碟(thumb drive)、記憶

卡等，且不包括傳播信號。軟體可適合於在並行處理器上或串行處理器上執行，使得方法步驟可以任何適合的次序執行或同時執行。

[0014]此承認韌體及軟體可為有價的、可分開交易之商品。意欲包含在「簡易」或標準硬體上運行或控制「簡易」或標準硬體來執行所需功能之軟體。亦意欲包含「描述」或定義硬體組配之軟體，諸如 HDL(硬體描述語言)軟體，其用於設計矽晶片或用於組配通用可規劃晶片來執行所需功能。

[0015]以上特徵可適當地組合，如熟習此項技術者將易於瞭解，且以上特徵可與實例之態樣中之任一者組合。

【圖式簡單說明】

[0016]現將參照以下圖式藉由實例來描述實施例，其中：

圖 1 例示出數位信號處理系統；

圖 2 例示出轉移引擎之示意圖；

圖 3 展示例示出各種示範性解交插方法之示意圖；

圖 4 例示出使用轉移引擎對兩個資料區塊執行之列-行操作之實例；

圖 5 展示例示出兩個另外的示範性解交插方法之示意圖；

圖 6 例示出圖 4 之列-行操作之實例，其具有增強來抵消 DRAM 裝置之限制；

圖 7 展示示範性時間交插資料區塊；

圖 8 係示範性解交插方法之流程圖；

圖 9 展示在圖 8 之用於圖 7 所展示之輸入交插區塊的方法之第一階段結束時儲存在 DRAM 中的資料項的柵格表示；以及

圖 10 展示在圖 8 之用於圖 7 所展示之輸入交插區塊的方法之第二階段結束時儲存在晶載記憶體中的資料項的柵格表示。

在所有圖中使用共同的參考數字來指示類似特徵。

【實施方式】

詳細說明

[0017] 以下僅藉由實例來描述實施例。此等實例代表申請人當前已知的將實施例投入實踐之最佳方式，但該等方式並非可達成此實踐之僅有方式。描述陳述了實例之功能及用於構成及操作實例之步驟序列。然而，相同或等效的功能及序列可藉由不同實例實現。

[0018] 以下描述一種利用通用數位信號處理器(DSP)以及專門硬體周邊兩者之數位信號處理系統。為允許對記憶體之有效率的使用，系統的不同元件能夠取用共用的晶載記憶體。可藉由轉移引擎，諸如直接記憶體存取(DMA)控制器，將資料項寫入至晶載記憶體或自晶載記憶體讀取資料項。晶載記憶體包含靜態隨機存取記憶體(SRAM)且轉移引擎亦具有接至動態 RAM(DRAM)之埠，該埠可在晶片外部或在晶片上。轉移引擎具有位址產生元件，該位址產生元件允許自記憶體讀取資料項的不同序列及/或將其寫入

至記憶體，且此等序列可包含資料項的線性及非線性序列。

[0019]『線性』一詞在本文中係關於資料項的讀取/寫入序列來使用，用來指讀取/寫入連續(或相連)的資料項。相反，『非線性』一詞在本文中係關於資料項的讀取/寫入序列來使用，用來指讀取/寫入非連續(或非相連)的資料項，且以下描述非線性序列之實例。

[0020]以下描述中對 DRAM 之任何使用意欲涵蓋任何形式之 DRAM，包括同步 DRAM、雙倍資料速率(DDR)DRAM(可稱為 DDR RAM)及叢發存取 DRAM。如上所述，DRAM 裝置按頁來組織其儲存內容，且一次僅能使有限數目個頁打開。當取用任何類型之 DRAM 時，頻繁取用不同頁的資料取用模式可為低效率的，因為該模式花費許多附加項循環來打開頁。在叢發取用 DRAM 中，DRAM 介面讀取/寫入 4 個、8 個、16 個、32 個或 64 個(或更多)連續位元組之叢發。使用不完全 DRAM 介面叢發之取用模式亦為低效率的。

[0021]當資料項在記憶體位置之間轉移或自一個記憶體轉移至另一記憶體(例如，SRAM 與 DRAM 之間)時，讀取/寫入資料項的不同序列之能力允許即時對資料項執行重新排序操作，諸如交插或解交插。此避免需要在數位信號處理系統上包括用於交插或解交插之專用(非共用)記憶體，進而又減少晶片面積及成本。所使用之不同序列可經配置以來抵消某些類型之記憶裝置，諸如 DRAM(就面積及因此成本而言，DRAM 比 SRAM 用起來更便宜，且因此可

使用更大的 DRAM)之效能限制，如以下更詳細描述。

[0022]在以下描述中，僅作為實例來使用時間交插/解交插；然而應瞭解，方法亦適用於其他形式之交插/解交插，諸如位元交插/解交插。

[0023]首先參照圖 1，其展示示範性數位信號處理系統單晶片 100 之結構。系統 100 包含晶載記憶體 102 及 DRAM 112，兩者皆連接至轉移引擎 106。記憶裝置 102、112 兩者皆用於資料項之儲存，且兩者皆可提供共用的記憶體空間(例如，儲存與數位信號處理系統有關之資料以及 MPEG 或其他視訊串流相關資料)。晶載記憶體 102 可為任何適合形式之 RAM，諸如(但不限於)SRAM，但不是 DRAM。DRAM 112 可在晶片上或在晶片外部(在其不可由 DSP 104 直接取用之意義上)，且在以下描述中，『晶載』記憶體一詞用於指作為非 DRAM 記憶元件之晶載記憶體 102，儘管事實上 DRAM 112 亦可為晶載記憶體(即，系統單晶片 100 之整體部分，因為其形成於同一片矽上)。

[0024]一或多個 DSP 104 連接至晶載記憶體 102。DSP 104 係可規劃來對資料執行信號處理計算，例如快速傅立葉變換及等化之處理器。當不考慮通用處理器時，DSP 104 比以下描述之硬體周邊更可組配。DSP 104 執行程式碼/指令來：自晶載記憶體 102 讀取資料，對資料執行信號處理操作，以及將資料寫回至晶載記憶體 102。

[0025]轉移引擎 106 亦連接至晶載記憶體 102，轉移引擎 106 為多個硬體(HW)周邊 108 提供對晶載記憶體 102 之

取用權。在一些實例中，轉移引擎 106 可呈直接記憶體存取(DMA)控制器之形式。轉移引擎 106 提供可由硬體周邊 108 使用的多個記憶體存取通道(例如，DMA 通道)，以允許自晶載記憶體 102 讀取資料或將資料寫入至晶載記憶體 102。

[0026]如以上所指出，硬體周邊 108 係經組配來執行特定信號處理任務之專門、專用的固定功能硬體區塊。例如，一個硬體周邊可為專門的 Viterbi 解碼區塊，且另一個硬體周邊可為專門的 Reed-Solomon 解碼區塊。硬體周邊亦可稱為加速器。硬體周邊中之每一者彼此獨立操作。硬體周邊可能可充分組配成具備特定針對其任務之操作參數，但硬體周邊未充分組配來改變其任務(例如，Viterbi 區塊不能被重新組配為 Reed-Solomon 區塊)。因此，硬體周邊比 DSP 104 更專門針對特定任務。然而，硬體周邊經配置來以非常快速且有效率的方式執行其專門任務。一般控制處理器 110 亦連接至晶載記憶體 102，一般控制處理器 110 可用於初始化，組配及控制數位信號處理系統之操作。

[0027]以上描述之數位信號處理系統提供信號處理操作中的靈活性。例如，系統可經配置來操作，使得不同 DSP 104 及硬體周邊 108 以任何所需的組配或序列來處理資料。每一硬體周邊或 DSP 可對由系統之其他部分提供並且儲存在晶載記憶體 102 中的一或多個資料區塊(本文中亦稱為資料緩衝器)進行操作，且產生並儲存將由系統之其他元件使用之一或多個資料緩衝器。此允許數位信號處理系統

用於多種不同類型之信號，例如，用於不同的廣播/電信標準。

[0028]對由晶載記憶體 102 提供之共同的記憶體空間之使用允許減少在系統單晶片 100 中提供之記憶體儲存總量。在不使用共同的記憶體空間之情況下，每一處理元件具備其自有的專用記憶體。例如，DSP 104 中之每一者可具有其自有的工作空間記憶體，一般控制處理器 110 具有用於儲存執行碼及資料之另一獨立的記憶體，硬體周邊 108 具有獨立的輸入及輸出緩衝器，以及一或多個額外記憶體可用於在處理元件之間交換資料。

[0029]因為數位信號處理系統可組配來允許實行不同的通訊標準，所以此等獨立的記憶體中之每一者需要針對對任何給定記憶體具有最大需求之特定標準來分別設定大小。換言之，DSP 記憶體需要為足夠大以適應對 DSP 記憶體具有最大需求之標準。類似地，硬體周邊緩衝器需要為足夠大以適應對硬體周邊緩衝器具有最高需求之標準(其可能不同於具有高 DSP 記憶體需求之標準)。因此，大量記憶體通常未被處理元件中的一些使用。

[0030]然而，若晶載記憶體 102 提供共同的記憶體空間，則可將不同標準的記憶體要求(而非不同標準對系統的個別元件之要求)作為整體來考慮。換言之，晶載記憶體 102 需要為足夠大以適應標準之最大的總的、總記憶體需求。其作用在於，使該等標準之間的不同記憶體需求平均化(例如，一個標準可能需要更多的 DSP 記憶體，但更小的緩衝

器，而另一標準可能係相反情況)。其作用在於，所需的總記憶體量顯著較低，且因此節省矽面積。

[0031]由晶載記憶體 102 提供之共同的記憶體空間可因此容納由系統使用之所有不同類型之資料，諸如數位信號處理器工作空間、用於一般控制處理器之執行碼及資料、用於硬體周邊中的一或多個之輸入及輸出緩衝器、用於在處理器之間交換資料之一或多個緩衝器，以及用於數位信號處理系統之其他組配資料。

[0032]現參照圖 2，其例示出轉移引擎 106 之示意圖。轉移引擎 106 包含：第一記憶體埠 202，其經配置來連接至晶載記憶體 102；以及第二記憶體埠 204，其經配置來連接至 DRAM 112。轉移引擎 106 亦包含多個周邊埠 206，其中每一者經配置來連接至相關聯的硬體周邊 108。記憶體埠 202、204 及周邊埠 206 全部連接至橫桿(crossbar)208，其允許此等埠中之任一者連接至此等埠中之任何其他埠。

[0033]轉移引擎 106 進一步包含位址產生元件 210，其耦接至記憶體埠 202、204 兩者且經配置來為連接至記憶體埠 202、204 之記憶體中的任一者或兩者產生讀取及/或寫入位址之序列。在一些實例中，位址產生元件 210 可包含可組配的位址產生器，其可規劃來以許多不同模式(例如，線性及非線性模式)操作，且其可經組配來自一組可能的模式中選擇一或操作模式。在其他實例中，位址產生元件 210 可包含一或多個專用硬體區塊，該等硬體區塊經配置來產生特定位址序列(例如，針對資料項之特定配置使用列-行模

式之序列，及針對資料項之特定配置使用叢發列-行模式之序列)。在一些實例中位址產生元件 210 可產生線性及非線性序列兩者，且在其他實例中，直接連接可用於線性序列且位址產生元件可用於僅產生非線性序列。

[0034]藉由產生讀取及/或寫入位址之非線性序列，位址產生元件 210 可執行對儲存於記憶體(例如，晶載記憶體 102 或 DRAM 112)上之資料項之非線性重新排序，該記憶體連接至轉移引擎 106 之埠中之一者。例如，圖 2 例示出儲存於晶載記憶體 102 上之資料項的第一序列 212 在轉移至 DRAM 112 期間可如何重新排序。在圖 2 之實例中，在晶載記憶體 102 上存在八個資料項，該等八個資料項儲存在用 0 至 7 表示之記憶體位址。在其他實例中，記憶體位址可自除零之外的基底位址開始，及/或每一個別資料項可大於記憶裝置上的單個記憶體位置。在此實例中，此等資料項被轉移至 DRAM 112，但以不同於第一序列 212 之第二序列 214 來排序。為清楚起見，第二序列 214 中的資料項儲存於 DRAM 112 上用 0'至 7'表示之記憶體位址，但在其他實例中此等位址可自除零之外的基底位址開始。

[0035]在第一個實例中，位址產生元件 210 可產生非線性讀取序列[3、6、4、1、2、7、0、5]且將此讀取序列提供至第一記憶體埠 202。位址產生元件 210 亦可產生線性寫入序列[0'、1'、2'、3'、4'、5'、6'、7']且將此序列提供至第二記憶體埠 204(其中僅為達成解釋之目的，DRAM 112 上的位址係用 0'、1'等表示，以區分該等位址與晶載記憶體 102

上之位址)。此導致第一記憶體埠 202 首先自讀取序列中的第一位址(位址 3)讀取資料項，其在此實例中係資料項「A」。經由橫桿 208 將此資料項傳遞至第二記憶體埠 204，第二記憶體埠 204 將此資料項寫入至寫入序列中的第一記憶體位址(位址 0')。此導致資料項「A」被重新排序，自第一序列 212 中的第四資料項變為第二序列 214 中的第一資料項。此操作藉由讀取讀取序列中的下一位址(位址 6、位址 4 等)及將對應的資料項(B、C、...)寫入至寫入序列中的下一位址(位址 1'、位址 2'、...)來重複。因此，來自第一序列的資料項(用 G、D、E、A、C、H、B、F 表示)現在以第二序列(A、B、C、D、E、F、G、H)儲存在 DRAM 上。

[0036]在第二個實例中，亦可藉由位址產生元件 210 產生線性讀取序列[0、1、2、3、4、5、6、7]及非線性寫入序列[6'、3'、4'、0'、2'、7'、1'、5']來達成相同的資料項重新排序。在此實例中，首先自晶載記憶體上的位址 0 讀取資料項「G」且將其寫入至 DRAM 上的位址 6'，後續接著自晶載記憶體上的位址 1 讀取資料項「D」且將其寫入至 DRAM 上的位址 3'，等等。類似地，在第三個實例中，亦可藉由位址產生元件 210 產生非線性讀取序列及非線性寫入序列來達成相同的資料項重新排序。此情形之一個實例將係讀取序列[0、2、4、6、1、3、5、7]及寫入序列[6'、4'、2'、1'、3'、0'、7'、5']。

[0037]在以上實例中之每一者中，在由轉移引擎 106 將資料項自晶載記憶體 102 直接轉移至 DRAM 112 期間執

行自第一序列至第二序列之重新排序。對於自 DRAM 112 至晶載記憶體 102 或自晶載記憶體至晶載記憶體中的另一位置之轉移，及類似地對於自 DRAM 至 DRAM 中的另一位址之轉移，亦可執行類似操作。

[0038] 以上實例亦展示讀取及寫入位址序列係在執行轉移之前完全產生。然而，此位址序列產生亦可與轉移同時執行，例如，藉由當正在讀取/寫入一或多個先前資料項時產生一或多個讀取及寫入位址。

[0039] 以上描述之處理程序允許作為至 DRAM 112 之記憶體轉移操作之整體部分將晶載記憶體 102 上之資料項重新排序成不同序列，且類似地，作為至晶載記憶體 102 之記憶體轉移操作之部分可將 DRAM 112 上的資料項重新排序成不同序列。此可用來實行交插或解交插，例如，藉由使用經配置來根據交插方案來產生讀取/寫入位址序列之位址產生元件 210。

[0040] 圖 3 展示例示出各種示範性解交插方法之示意圖。在第一示意圖 300 中，在自晶載記憶體 102 至晶載記憶體 102 之單個轉移中執行解交插。在後續兩個示意圖 302、304 中存在兩個轉移：一個轉移係自晶載記憶體 102 至 DRAM 112，且第二轉移係自 DRAM 回到晶載記憶體 102。在第二示意圖 302 中，可藉由以下方式來執行儲存於晶載記憶體 102 上之資料項之解交插：根據線性寫入序列將資料項寫入至 DRAM 112，及接著使用特定非線性序列自 DRAM 112 讀取回該等資料項，可將其稱為『列-行模式』

或『列-行交插』。以下參照圖 4 詳細描述此非線性序列。或者，可藉由以下方式來執行資料項之解交插：使用列-行模式將資料項寫入至 DRAM 112，及接著線性讀取回該等資料項，如圖 3 中的第三示意圖 304 所展示。

[0041]在圖 3 所展示之所有實行方案中，直至所有交插資料在輸入記憶體(即，在圖 3 中的各圖中之每一者之左手邊上展示之晶載記憶體 102)中，解交插處理程序才開始。

[0042]列-行模式考慮資料項將配置於具有多個列及行之一或多個柵格或表中。此係如圖 4 所例示，圖 4 展示：輸入資料項的第一區塊 402，其(僅為達成例示性目的)具有自 0 至 23 的相連記憶體位址；以及輸入資料項的第二區塊 404，其(同樣僅為達成例示性目的)具有自 24 至 47 的相連記憶體位址。若參照圖 3 中的第二個實例 302 來描述列-行模式，則此等記憶體位址在 DRAM 112 中。在圖 4 所展示之實例中，考慮資料項係每六個資料項具有分欄符號，如圖 4 中由虛線所指示。此意味考慮連續記憶體位址係沿著具有六列之柵格之各行來配置(且可將此描述為沿著行來寫入/讀取資料)。

[0043]以柵格形式呈現之資料項展示於圖 4 中，圖 4 展示針對輸入資料項的第一區塊 402 之第一柵格 406 及針對輸入資料項的第二區塊 404 之第二柵格 408。第一及第二柵格皆具有六列及四行。可注意到，連續定址之資料項係沿著行來配置。然而，在其他實例中，資料項亦可呈現為使得連續項改為沿著列來配置，在此情況下以下描述仍適

用，但應顛倒對列及行之參考。

[0044]列-行模式之目的係調換每一柵格，使得當輸入資料項(例如，來自 DRAM 112)以遍歷柵格之行之序列來配置時，輸出資料項(例如，輸出至晶載記憶體 102)以遍歷柵格之列之序列來配置。例如，參照柵格 406，若輸入資料序列的前四個資料項係 A、B、C、D(沿第一行讀取四個項)，則輸出資料序列的前四個資料項係 A、G、M、S(沿第一列讀取四個項)。諸如此操作之列-行操作因此取決於多少列被定義為存在於柵格中來改變資料項之次序。

[0045]為實行列-行模式，位址產生元件 210 產生導致列-行調換之讀取及寫入序列。此可藉由產生非線性讀取序列(例如，自 DRAM 112)及線性寫入序列(如圖 4 所例示且以下更詳細描述)或可藉由產生線性讀取序列(例如，自晶載記憶體 102)及非線性寫入序列(例如，如圖 3 中的第三個實例 304 所展示)來達成。在另外的實例中，亦可使用非線性讀取序列及非線性寫入序列來允許有效率的記憶體取用，如以下參照圖 6 描述。

[0046]圖 4 展示非線性讀取序列 410 之實例，可以看出其包含非連續記憶體位址。在一個實例中，可使用由以下偽碼所例示之演算法來產生位址序列：

```

N0 = rows * columns;
N1 = rows;
N2 = numBlocks * rows * columns;
For ind = 1 to numItems
    nextItemAddr = a + o;
    a = a + N1;
    if a >= N0
        a = a - N0 + 1;

```

```

        b = b + 1;
        if b >= N1
            a = 0;
            b = 0;
            o = rem(o + N0, N2);
        end
    end
end
end

```

[0047] 其中「rows」(N1)係柵格中的列數(在圖 4 的實例中係六個)，「columns」係柵格中的行數(在圖 4 的實例中係四個)，「numBlocks」係資料項的區塊數(在圖 4 的實例中係兩個)，且「numItems」係所有區塊之資料項的總數(在圖 4 的實例係 48)。變數「a」、「b」及「o」係演算法內使用之內部變數，其可全部初始化為零，或其中一或多個可初始化為非零值來應用位移。

[0048] 在計算 N0 之初始值(柵格中的列數)、N1(列數乘以行數)及 N2(列數、行數及資料項的區塊數之乘積)之後，演算法歷經存在之資料項數而反覆，在每次反覆中計算序列中的下一位址(「nextItemAddr」)。實際上，演算法跳過來自輸入序列的固定數目個資料項(例如，在圖 4 中係六個)直至達到列之末尾(由第一個「if」語句確定)，且接著使該列之起點遞增一並且重複。區塊之末尾可由第二個「if」語句偵測，該第二個「if」語句重設計算但加上根據餘數運算 rem(.) 計算出之位移(在圖 4 中係 24)。接著重複該處理程序，直至達到「numItems」。注意，「numItems」可被設定為大於存在的資料項的總數的值，且若如此，則一旦已取用所有區塊，演算法就折回至第一區塊。

[0049] 圖 4 展示由以上演算法產生之讀取序列 410，其

中頂部列展示第一區塊(柵格 406)之序列且底部列展示第二區塊(柵格 408)之序列。以讀取序列 410 的前四項作為實例，此等項自位址 0、6、12、18 讀取，該等位址對應於來自輸入資料項 402 的資料項 A、G、M、S。可以看出此對應於柵格 406 的第一列。

[0050]位址產生元件 210 產生具有連續記憶體位址之線性寫入序列 412，使得當讀取序列 410 及寫入序列 412 由轉移引擎 106 使用時，以非線性序列來讀取資料項且以寫入線性序列來寫入資料項。注意，為簡單起見，圖 4 中之寫入序列具有自 0 至 47 之位址，但在其他實例中位址可自任何基底位址開始。讀取序列 410 及寫入序列 412 之組合之結果可見於輸出資料項的第一區塊 414 及輸出資料項的第二區塊 416 中。藉由比較此等輸出資料項與柵格 406 及 408，可以看出已成功執行列-行操作。

[0051]亦可藉由產生線性讀取序列及非線性寫入序列而獲取相同結果(例如，圖 3 中之第二個實例 304)，如下(為簡潔起見僅展示第一區塊)：

[0052]讀取序列：

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
---	---	---	---	---	---	---	---	---	---	----	----	----	----	----	----	----	----	----	----	----	----	----	----

[0053]寫入序列：

0	4	8	12	16	20	1	5	9	13	17	21	2	6	10	14	18	22	3	7	11	15	19	23
---	---	---	----	----	----	---	---	---	----	----	----	---	---	----	----	----	----	---	---	----	----	----	----

[0054]可使用與以上詳細描述之非線性讀取序列類似的技術來產生非線性寫入序列。以上實例例示出位址產生元件 210 如何可用來對資料項之集合實行交插/解交插操作，諸如列-行交換。

[0055]圖 5 展示示意圖 502、506，其例示出兩個另外的示範性解交插方法，該等方法在與 DRAM 112 互動之方式方面更有效率。此等方法皆將時間解交插處理程序分為兩個記憶體至記憶體轉移操作(晶載記憶體至 DRAM 及接著 DRAM 至晶載記憶體)，其中每一者使用至少一個非線性位址序列。此等方法亦皆使用列-行組合(R-C)模式(如以上參照圖 4 所描述，箭頭 521 及 561)及叢發列-行模式(B R-C)模式(箭頭 523、524 及 562 至 564)。

[0056]雖然圖 5 展示資料項自晶載記憶體 102 轉移至 DRAM 112 及接著回到晶載記憶體 102，但應瞭解，資料項可改為被寫回至與最初自其讀取該等資料項之晶載記憶體分開的晶載記憶體，或資料項可被寫回至晶載記憶體 102 之不同部分。在一實例中，可自晶載記憶體 102 之一部分讀取(在由箭頭 521 及 561 指示的操作中)資料項，該部分可稱為瓦狀塊化緩衝器，且隨後可將資料項(以解交插形式)寫回至晶載記憶體 102 之一獨立部分(在由箭頭 524 及 564 指示的操作中)，該獨立部分可稱為解交插器輸出緩衝器。此等兩個緩衝器可具有不同大小。在以下描述中，對自晶載記憶體 102 被轉移且接著回到晶載記憶體 102 之資料項之任何參考僅作為實例，並且所描述之方法亦可用來將資料項自一個晶載記憶元件轉移至另一晶載記憶元件(經由 DRAM)或自晶載記憶體 102 之一部分轉移至晶載記憶體 102 之另一部分(經由 DRAM)。

[0057]可將叢發列-行模式視為以上描述之列-行模式

之變體，或者，可將列-行模式視為叢發長度為一之叢發列-行模式之特定實例。叢發列-行模式考慮資料被配置於具有列及行之柵格中(如以上所描述)；然而，代替在沿著列遍歷時自每一行僅讀取一個資料項(如在列-行情況下)，叢發列-行模式在沿著列跳至下一行之前(即，藉由跳過 $r-L$ 個資料項，其中 r =柵格中的列數)讀取預定數目個連續位址(其中此預定數目被稱為『叢發長度』 L)。例如，參照圖 4 之柵格 406，若叢發長度係三，則叢發列-行模式首先在一個叢發中讀取三個連續項(項 A、B、C)，接著移動至下一行且讀取接下來的三個連續項(G、H、I)，後續接著讀取 M、N、O 且接著讀取 S、T、U。叢發列-行模式接著折回到第一行且讀取 D、E、F，後續接著讀取 J、K、L 等。叢發列-行模式可因此被視為與列-行模式相同，除了讀取連續資料項之群組而非僅一個資料項之外，或者，列-行模式可被視為叢發長度等於一之叢發列-行模式。

[0058]在一個實例中，可使用由以下偽碼所例示之演算法來產生叢發列-行模式之讀取序列：

```

N1 = rows;
N3 = burstLength;
N4 = rows * columns - burstLength;
For ind = 1 to numItems
    nextItemAddr = a + o;
    a = a + 1;
    if a >= N3
        a = 0;
        o = o + N1;
        if o >= N4 + N1
            o = 0;
        elseif o >= N4 + N3
            o = o - N4;

```

```

        end
    end
end

```

[0059]此偽碼中的變數被定義為在以上列-行模式之描述中所陳述的。此外，「burstLength」(N3)係在每一叢發中讀取之連續或相連項之數目，且 N4 係列數(N1)與行數之乘積減 N3。注意，叢發列-行操作之寫入序列亦可以類似方式來產生。

[0060]叢發列-行模式可用來允許由某些類型之記憶裝置，諸如 DRAM 112，來有效率地執行解交插操作，尤其在 B R-C 模式中的叢發長度(L)與 DRAM 介面叢發大小相同或接近的情況下。藉由以此方式(或根據以下描述之其他實例)基於 DRAM 介面叢發大小來選擇叢發長度(或叢發大小)，此有效率地使用 DRAM 介面。相反，許多習知解交插器取用模式試圖連續讀取/寫入有很大間隔之資料項，從而由於不完全(DRAM 介面)叢發及跨越許多 DRAM 頁而導致對 DRAM 裝置之記憶體取用。

[0061]例如，圖 4 之列-行操作讀取由柵格中的列數間隔開之連續資料項。在存在大量列之實例中，此可導致跨越記憶裝置有很大間隔之取用，從而導致自不同 DRAM 頁進行低效率的取用。返回參看圖 3 所展示之實例，可以看出，在第二個實例 302 中，以列-行模式自 DRAM 讀取係低效率的，且在第三個實例 304 中，以列-行模式寫入至 DRAM 亦為低效率的。

[0062]圖 6 例示出解交插操作之實例，其不會導致與頻繁取用不同頁或部分填充叢發相關聯的 DRAM 取用之無效

率。此實例亦展示於圖 5 中之第一示意圖 502 中。圖 6 之實例產生與圖 4 之列-行結果相同的列-行結果(即，有六列、四行及兩個區塊的交換)，但使用許多遍線性循序記憶體取用來進行此操作，從而導致如 DRAM 之分頁裝置的有效率的操作。在圖 6 之實例中，轉移引擎自晶載記憶體 102 讀取輸入資料項之序列，將資料項儲存在 DRAM 112 上，且接著自 DRAM 112 讀回資料項且將其寫入至晶載記憶體 102(可能會覆寫其原始位置)，其中交換了列與行。

[0063] 為達成解釋之目的，輸入資料項 602 與圖 4 之實例中所使用的輸入資料項相同。存在總共 48 個資料項，其具有自零開始之記憶體位址之連續序列。首先，以每區塊或瓦狀塊具有六列及兩行之列-行模式自晶載記憶體 102 讀取資料項。如圖 6 所展示，可考慮該等資料項被配置於瓦狀塊 604 中，每一瓦狀塊具有六列及兩行。本文中使用的瓦狀塊大小僅作為實例，且在許多實行方案中，可設定瓦狀塊大小等於 DRAM 介面叢發大小。又(即，使用列-行模式)藉由如以上所描述之位址產生元件 210 產生沿著此等瓦狀塊中之每一者之列進行讀取之非線性讀取序列 606。亦藉由位址產生元件 210 產生線性寫入序列 608。轉移引擎 106 使用非線性讀取序列 606(圖 5 中的箭頭 521)自晶載記憶體 102 讀取且使用線性寫入序列 608(箭頭 522)寫入至 DRAM 112。以此方式寫入至 DRAM 並非低效率的，因為其線性地寫入至相連位址，且因此當資料項數足夠時僅將偶爾跨越 DRAM 頁邊界。

[0064]由於此操作，可以看出 DRAM 112 上的資料項 610 對應於自瓦狀塊 604 之列-行交換。接著由位址產生元件 210 產生非線性讀取序列 612，其自 DRAM 112 讀取回此等資料項。此讀取序列係使用叢發列-行模式產生的且經組配來避免低效率取用。叢發列-行模式在此實例中使用每叢發六項、十二列及兩行。因為 DRAM 讀取序列 612 讀取資料項之叢發，所以此等資料項在 DRAM 上位於連續位址且因此不太可能跨越頁邊界且亦將有效率地使用在 DRAM 介面上可利用的叢發(尤其當位址產生器叢發長度 L 接近 DRAM 介面叢發大小時)。因此，相對於(非叢發)列-行取用，將跨越顯著更少的頁邊界。

[0065]亦產生非線性寫入序列 614 來將資料項寫回至晶載記憶體 102。此寫入序列 614 亦係使用叢發列-行模式產生的，且在此實例中使用每叢發兩項、四列及三行。讀取序列 612(圖 5 中的箭頭 523)及寫入序列 614(箭頭 524)之組合使得寫回至晶載記憶體 102 之輸出資料項 616 係在與當執行具有六列、四行及兩個區塊之基本列-行操作時相同的序列中(此可與圖 4 相比較)，除了資料係儲存在 DRAM 112 上而不會導致由於頁邊界及不完全叢發所導致之無效率之外。此外，因為自晶載記憶體 102 之初始讀取(圖 5 中的箭頭 521)使用具有僅兩行之瓦狀塊之列-行操作，所以此允許整個瓦狀塊一到達晶載記憶體 102 時，至 DRAM 之資料轉移就開始，此比當使用四行區塊(如圖 4 中)時更快。此可在即時資料之情況下改良效能，在該情況下，資料在串

流中逐漸到達。

[0066] 圖 7 至圖 10 例示出另一解交插操作之實例，其不會導致與取用不同頁相關聯的 DRAM 取用之無效率。此方法亦展示於圖 5 中的第二示意圖 506 中。如自圖 5 可以看出，此方法僅涉及在一個叢發內與 DRAM 112 之線性互動，即，使用叢發列-行模式來寫入至 DRAM(箭頭 562)並且自 DRAM(箭頭 563)讀取。如上所述，此意味將跨越非常少的頁邊界且有效率地使用 DRAM 介面叢發，且此改良了方法之總體效率。

[0067] 僅為達成例示性目的，此方法考慮資料項被配置於具有多個列及行之一或多個柵格或表中(如在前述實例中)且進一步使用瓦狀塊之概念，該瓦狀塊係由呈列-行結構之資料集形成。如以下所描述，瓦狀塊可根據 DRAM 介面叢發或頁大小來設定大小。應瞭解，記憶體中的資料係儲存在相連的記憶體位置。

[0068] 圖 7 展示示範性時間交插資料區塊 700，其包含 200 個資料項(標記了位址 0 至 199)，該等資料項配置於 10 個瓦狀塊 702(T_0 至 T_9)中，每一瓦狀塊包含 20 項。在 DRAM 112 係叢發模式取用之 DRAM 的情況下，可選擇瓦狀塊大小來匹配 DRAM 介面叢發大小，且此進一步改良方法之效率，因為記憶體轉移(如以下所描述)有效率地使用 DRAM 介面。若瓦狀塊大小不匹配 DRAM 介面叢發大小，則瓦狀塊大小可替代地小於 DRAM 介面叢發大小，或每瓦狀塊可有多個叢發。在瓦狀塊大小不完全匹配 DRAM 介面叢發大

小之許多實例中，使瓦狀塊與 DRAM 中的頁邊界對準，且此可提供 DRAM 介面效率之顯著改良。如以下更詳細描述，瓦狀塊大小之選擇限制晶載 RAM 瓦狀塊化緩衝器(即，自其讀取資料之晶載記憶體 102)之大小，因為直至至少一個整個瓦狀塊被儲存在瓦狀塊化緩衝器中，方法才開始。

[0069]應瞭解，雖然圖 7 中之示範性時間交插區塊 700 包含 200 個資料項，但此等區塊可顯著大於此，且可包含數千個資料項。此外，時間交插區塊中的列及行之配置可由使用該方法之系統來設定。

[0070]解交插處理程序在此實例中被分為若干記憶體至記憶體轉移階段，其中每一轉移(或『瓦狀塊化工作』)轉移許多瓦狀塊，如可參考圖 8 所展示之流程圖來解釋。圖 8 所展示之方法在每一瓦狀塊化工作中轉移 N 個瓦狀塊且可選擇 N 的值等於一行瓦狀塊(例如，在圖 7 所展示之實例中， $N=2$)。然而，在其他實例中，瓦狀塊化工作可包含許多行瓦狀塊(例如，多於一行)以減少所需要的瓦狀塊化工作的數目。僅為達成解釋之目的，將參考圖 7 所展示之時間交插資料區塊 700 來描述圖 8 所展示之方法，且其中 $N=2$ 。在瓦狀塊化工作包含多於一行瓦狀塊之實例中，方法如以下所描述來操作，且僅改變位址產生器之組配(即，此會使位址產生器處理更多資料)。

[0071]一旦來自時間交插區塊的最小值為 N 的瓦狀塊(即，至少 N 個瓦狀塊)被儲存在晶載記憶體 102(區塊 802)

中，例如，一旦瓦狀塊 T_0 及 T_1 被儲存在晶載記憶體 102 中，方法即可開始。如上所述，晶載記憶體 102 中儲存此等交插瓦狀塊 T_0 及 T_1 之部分可稱為瓦狀塊化緩衝器，且因為記憶體至記憶體轉移之第一階段 81 對 N 個瓦狀塊進行操作，所以僅可將此瓦狀塊化緩衝器設定大小成能夠儲存 N 個瓦狀塊之資料。在一實例中，瓦狀塊化緩衝器可為彈性緩衝器，該彈性緩衝器可以允許一或多個瓦狀塊工作之方式來設定大小，其取決於系統輸送量、可利用的記憶體頻寬及 DRAM 介面。

[0072]使用列-行模式自晶載記憶體 102 讀取第一瓦狀塊 T_0 (區塊 804 及圖 5 中的箭頭 561)。所使用的第一瓦狀塊之非線性讀取序列因此係：

0	10	20	30	1	11	21	31	2	12	22	32	3	13	23	33	4	14	24	34
---	----	----	----	---	----	----	----	---	----	----	----	---	----	----	----	---	----	----	----

其中以上數字對應於資料項在晶載記憶體中之位址，如圖 7 所展示。返回參看先前對列-行模式（且尤其是所提供之偽碼實例）之描述，可以看出，讀取一資料項（即，一個資料項）且接著在讀取另一資料項之前跳過接下來的 9 個資料項。重複此操作，直至已讀取總共 4 個資料項（瓦狀塊中之行數），且接著以一個資料項之位移（即，讀取位址 1，後續接著讀取 11）重複整個處理程序，以此類推，直至已讀取整個瓦狀塊。

[0073]接著使用叢發列-行模式將資料項之此序列寫入至 DRAM 112(區塊 806 及箭頭 562)，其中叢發長度 L 等於瓦狀塊中的資料元素數（例如， $L=20$ ）：

0'	1'	2'	3'	4'	5'	6'	7'	8'	9'	10'	11'	12'	13'	14'	15'	16'	17	18'	19'
0	10	20	30	1	11	21	31	2	12	22	32	3	13	23	33	4	14	24	34

其中第一列對應於資料項在 DRAM 中的位址，其標記為 0'至 19'，以區分該等位址與晶載記憶體 102 中的原始位址，自該晶載記憶體 102 讀取該等資料項，該等原始位址展示於第二列中。

[0074]接著重複此等兩個操作(區塊 804 中的讀取操作及區塊 806 中的寫入操作)，直至所有 N 個瓦狀塊已寫入至 DRAM(在區塊 808 中為『是』)。在此階段，在已將 N 個瓦狀塊寫入至 DRAM 後，可能已自晶載記憶體 102 讀取所有儲存的資料項，且在該情況下可用來自時間交插區塊的資料項之另外 N 個瓦狀塊再填充該晶載記憶體(區塊 810)。或者，在晶載記憶體中已儲存有另外的瓦狀塊(例如，至少 N 個另外的瓦狀塊)的情況下，方法可繼續讀取額外瓦狀塊(區塊 804)且將其寫入至 DRAM(區塊 806)而不需要再填充該晶載記憶體(即，省略掉區塊 810)。

[0075]重複此第一階段 81，直至已自晶載記憶體 102 讀取整個時間交插區塊 700 且將其寫入至 DRAM(在區塊 812 中為『是』)，其中視情況再填充晶載記憶體 102(在區塊 810)。在此實例中，將有五個轉移，其中每一個轉移兩個瓦狀塊(因為 N=2 且區塊 700 包含 10 個瓦狀塊)。

[0076]圖 9 展示在如圖 7 所展示之輸入時間交插區塊 700 之第一階段 81 結束時儲存在 DRAM 中之資料項的柵格表示 902(由區塊 700 中的原始位址位置參考)。在此柵格 902 旁邊係第二柵格 904，該第二柵格 904 識別 DRAM 122 中

的每一資料項之位址(標記為 0'至 199'以區分該等位址與晶載記憶體 102 中的原始位址 0 至 199)。在此柵格表示中，原始瓦狀塊被重新排序(與區塊 700 相比較)，但未被解交插，且來自瓦狀塊之經重新排序的資料項佔據連續記憶體位址(例如，T₀被儲存在位址 0'至 19'中)。如自圖 9 可以看出，柵格包含 40 列及 5 行，使得每一行資料項(其中連續資料項係按行來配置)包含兩個瓦狀塊。一行中的瓦狀塊之間的邊界由虛線 906 標出。

[0077] 在方法之第二階段 82，將資料項轉移回至晶載記憶體 102(或至另一晶片記憶元件，如以上描述)，且使用另一重新排序操作來完成資料之解交插。使用具有叢發長度 L 之叢發列-行模式自 DRAM 112 讀取第一瓦狀塊 T₀(區塊 814 及圖 5 中的箭頭 563)，L 同樣等於瓦狀塊中之資料元素數(在此實例中 L=20)，即，讀取序列係：

0'	1'	2'	3'	4'	5'	6'	7'	8'	9'	10'	11'	12'	13'	14'	15'	16'	17'	18'	19'
0	10	20	30	1	11	21	31	2	12	22	32	3	13	23	33	4	14	24	34

其中第一列對應於資料項在 DRAM 112 中的位址，且第二列展示晶載記憶體 102 中的原始位址，自晶載記憶體 102 讀取該等資料項。

[0078] 接著使用叢發列-行模式將瓦狀塊 T₀寫入至晶載記憶體 102(區塊 816 及箭頭 564)。叢發列-行模式使用叢發長度 L，L 等於原始時間交插區塊 700 中的瓦狀塊中之行數，例如，在圖 7 所展示之實例中為四。因此將資料寫入至晶載記憶體中的四個連續位址，跳過接下來的 16 個位址(原始時間交插區塊中的行數=調換後的區塊中的列數

=20，20-4=16)，且接著將資料寫入至接下來的四個連續位址，以此類推。非線性寫入序列因此係：

0	1"	2"	3"	20"	21"	22"	23"	40"	41"	42"	43"	60"	...	80"	81"	82"	83"
0	10	20	30	1	11	21	31	2	12	22	32	3	...	4	14	24	34

其中第一列對應於晶載記憶體中之位址，寫入係針對該等位址，其標記為 0"、1"等，以區分該等位址與晶載記憶體 102 中的原始位址，其中在第一階段 81 自晶載記憶體 102 讀取該等資料項，且此等原始位址展示於第二列中。

[0079]應注意，寫入至 DRAM 且自 DRAM 讀取之前兩個叢發列-行操作(箭頭 562 及 563)中所使用的叢發長度使用相同的叢發長度(例如，L=20)，且寫入至晶載記憶體之此第三叢發列-行操作(箭頭 564)使用不同的叢發長度(例如，L=4)。

[0080]接著逐個瓦狀塊地(且使用與第一階段 81 相同的瓦狀塊大小)重複此第二階段 82，直至所有瓦狀塊已寫入至晶載記憶體 102(在區塊 818 中為『是』)。

[0081]圖 10 展示在如圖 7 所展示之輸入時間交插區塊 700 之第二階段 82 結束時儲存在晶載記憶體中之資料項的柵格表示 1002(由區塊 700 中的原始位址位置參考)。在此柵格 1002 旁邊係第二柵格 1004，該第二柵格 1004 識別晶載記憶體 102 中的每一資料項之位址(標記為 0"至 199"以區分該等位址與晶載記憶體 102 中的原始位址 0 至 199 及 DRAM 112 中的位址 0'至 199')。在此柵格表示中，原始資料項被解交插，如自圖 10 可以看出，使得第一瓦狀塊 T₀

現在包含四列及五行(而非如區塊 700 中的五列及四行), 如由點線輪廓所展示。如自圖 10 可以看出, 一個解交插區塊之柵格包含 20 列及 10 行。

[0082]應瞭解, 雖然圖 7、圖 9 及圖 10 展示位址自基底位址 0 開始, 但在其他實例中, 位址可自任何基底位址開始。

[0083]自以上解釋及圖 8 可以看出, 方法中之讀取/寫入工作可對許多瓦狀塊(例如, 一或多個瓦狀塊)而非整個時間交插區塊進行操作。此允許針對特定 DRAM 介面叢發大小來優化方法, 例如, 可將瓦狀塊設定成最多為與一個 DRAM 介面叢發相同的大小, 且瓦狀塊化工作則將係整數個 DRAM 介面叢發(例如, 在以上參照圖 7、圖 9 及圖 10 所描述之實例中係兩個)。由 DRAM 介面定義之 DRAM 介面叢發大小可在 DRAM 中的頁或子頁層級上設定且將取決於匯流排頻寬, 且可設定成使得叢發之開始與頁之開始對準, 且在可能的情況下完全在一頁內完成(以防止由於記憶體分頁導致之無效率)。如上所述, 在瓦狀塊大小不完全匹配 DRAM 介面叢發大小或係 DRAM 介面叢發大小之倍數的情況下, 瓦狀塊可改為與頁邊界對準, 以便改良 DRAM 效率, 此係以未使用的 DRAM 容量為代價。

[0084]雖然以上描述及圖 8 展示串行地執行方法(即, 在第一階段 81 已完成之後執行第二階段 82), 但可並行地執行方法之態樣, 使得自 SRAM 讀取來自一個時間交插區塊之瓦狀塊且將其寫入至 DRAM(在第一階段 81), 且同時

自 DRAM 讀取來自另一時間交插區塊之瓦狀塊且將其寫入至 SRAM(在第二階段 82)。此允許記憶體重複使用，因為寫入至 DRAM 之操作(區塊 806)可使用的位址集合可與在第二階段 82 進行讀取(區塊 814)的位址集合相同，只要定時係使得在用來自另一時間交插區塊之資料項覆寫特定位址(區塊 806)之前讀取該特定位址(區塊 814)即可。

[0085]圖 8 所展示且以上描述之方法將調換資料項柵格之操作分為(為了執行解交插)兩個獨立的部分。當自 SRAM 讀取(區塊 804 及圖 5 中的箭頭 561)且寫入至 DRAM(區塊 806 及箭頭 562)時執行調換之第一部分，且當自 DRAM 讀取(區塊 814 及箭頭 563)且寫回至 SRAM(區塊 816 及箭頭 564)時執行調換之第二部分。所有此等調換使用非線性位址序列；然而，使用不同的非線性序列。在第一部分中，列-行模式用於自 SRAM 讀取(叢發長度=1)，且在第二部分中，叢發列-行模式用於寫入至 SRAM(叢發長度=瓦狀塊中的行數)。與 DRAM 之互動(區塊 806 中的寫入及區塊 814 中的讀取)使用叢發列-行模式，該叢發列-行模式之叢發長度等於瓦狀塊中之資料元素數(例如，在圖 7 至圖 10 所展示之實例中 $L=20$)。

[0086]以上參照圖 5(實例 506)及圖 7 至圖 10 所描述之方法有效率地使用可利用的 DRAM(且尤其是叢發取用的 DRAM)頻寬，因為多階段處理程序之使用涉及資料瓦狀塊(而非整個時間交插區塊)之轉移，其中瓦狀塊大小係根據 DRAM 介面叢發大小來選擇。瓦狀塊之配置特定針對特定

實行方案，且以上描述之方法可適用於瓦狀塊之任何配置及每瓦狀塊任何資料項數。

[0087]例如，在 DVB-T2 中使用該方法時，一行中的瓦狀塊數(N)可設定成等於前向誤差校正(FEC)區塊之數目，使得圖 7 至圖 10 所展示之實例可對應於存在兩個 FEC 區塊之情形。在其他實例中，可存在三個 FEC 區塊，因此 $N=3$ ，且在瓦狀塊化工作中將自 SRAM 轉移三個瓦狀塊至 DRAM 且將其寫入至 DRAM 中的連續位址。

[0088]以上描述之方法，解交插處理程序被分為若干階段。使用所描述之方法，不必在解交插處理程序可開始之前將整個交插資料區塊儲存在瓦狀塊化緩衝器中。如參照圖 8 所描述，僅有必要在方法開始之前將 N 個瓦狀塊儲存在瓦狀塊化緩衝器中。

[0089]以上參照圖 5(實例 506)及圖 7 至圖 10 所描述之方法可使用圖 2 所展示之位址產生元件 210 來實行。此位址產生元件 210 可為可組配的或可包含特定硬體邏輯，該硬體邏輯經配置來產生在該方法之特定實行方案(例如，針對瓦狀塊之特定配置)中使用之所需要的(預定)非線性位址序列。

[0090]以上描述之方法可用於解交插任何交插資料區塊。實例應用包括 OFDM 信號，且尤其是數位地面電視(DTT)信號，諸如 DVB-T2；然而，方法不限於 OFDM、DTT 或 DVB-T2。以上描述之方法亦可用於交插資料來形成交插資料區塊。為將以上描述之方法用於交插，而非解交插，

方法步驟保持相同，且不同之處在於輸入資料(例如，如在區塊 802 所儲存)包含解交插資料(而非交插資料)且輸出資料(例如，如在圖 8 結束時寫回至 SRAM)包含交插資料(而非解交插資料)。

[0091]「處理器」及「電腦」一詞在本文中用於指具有處理能力以便可執行指令之任何裝置。熟習此項技術者將認識到，此等處理能力併入於許多不同裝置中，且因此「電腦」一詞包括機上盒、媒體播放器、數位無線電、PC、伺服器、行動電話、個人數位助理及許多其他裝置。

[0092]熟習此項技術者將認識到，用來儲存程式指令或資料之儲存裝置可分佈於網路上。例如，遠端電腦可將所描述之處理程序之實例儲存為軟體。區域或終端電腦可取用遠端電腦且下載運行程式之軟體之一部分或全部。或者，區域電腦在需要時可下載軟體之片段，或在區域終端機處執行一些軟體指令且在遠端電腦(或電腦網路)處執行一些軟體指令。熟習此項技術者亦將認識到，藉由使用熟習此項技術者已知的習知技術，可由專用電路、可規劃邏輯陣列或類似物執行軟體指令之全部或一部分。

[0093]對「邏輯」之特定參考指執行功能之結構。邏輯之實例包括經配置來執行該(等)功能之電路。例如，此電路可包括電晶體及/或在製造過程中可利用的其他硬體元件。舉例而言，此等電晶體及/或其他元件可用來形成實行及/或含有記憶體之電路或結構，諸如暫存器、正反器或鎖存器、邏輯運算器，諸如布爾運算、數學運算器，諸如加

法器、乘法器或移位器，以及互連。此等元件可提供為定製電路或標準單元庫、巨集或在其他抽象層級。此等元件可在特定配置中互連。邏輯可包括具有固定功能之電路且電路可經規劃來執行功能；此規劃可由韌體或軟體更新或控制機制來提供。經識別為執行一個功能之邏輯亦可包括實行子功能或子處理程序之邏輯。在一實例中，硬體邏輯具有實行固定功能操作、狀態機或處理程序之電路。

[0094]如熟習此項技術者將易於瞭解，可擴展或改變本文中給出之任何範圍或裝置的值而不會喪失所尋求之效果。

[0095]應瞭解，以上描述之益處及優勢可關於一個實施例或可關於若干實施例。實施例不限於解決所陳述問題中之任一者或全部的實施例或具有所陳述益處及優勢中之任一者或全部的實施例。

[0096]對「一」項之任何參考指該等項中的一或多者。「包含」一詞在本文中用來意味包括所識別的方法區塊或元件，但此等區塊或元件不包含詳盡的清單，且設備可含有額外區塊或元件，且方法可含有額外操作或元件。

[0097]本文所描述之方法之步驟可以任何適合的次序執行或視情況同時執行。此外，在不脫離本文所描述之主題之精神及範疇的情況下可自該等方法中之任一者中刪除個別區塊。以上描述之實例中之任一者之態樣可與所描述之其他實例中之任一者之態樣相結合以形成另外的實例而不會喪失所尋求之效果。

[0098]應瞭解，以上對較佳實施例之描述僅作為實例提供且可由熟習此項技術者進行各種修改。雖然以上已用某種程度之具體性或參考一或多個個別實施例來描述各種實施例，但熟習此項技術者可在不脫離實例之精神或範疇的情況下對所揭示之實施例進行眾多變更。

【符號說明】

100...晶片/系統	402、404、602...輸入資料項
102...晶載記憶體/記憶裝置	406...第一柵格
104...DSP	408、904、1004...第二柵格
106...轉移引擎	410...非線性讀取序列/讀取
108...硬體(HW)周邊/硬體周 邊	序列
110...一般控制處理器	412...線性寫入序列/寫入序 列
112...DRAM	414、416、616...輸出資料項
202...第一記憶體埠/記憶體 埠	521～524...箭頭
204...第二記憶體埠	561～564...箭頭
206...周邊埠	604、702...瓦狀塊
208...橫桿	606...非線性讀取序列
210...位址產生元件	608...線性寫入序列
212...第一序列	610...資料項
300、502...第一示意圖	612...DRAM 讀取序列/讀取 序列
302、506...第二示意圖	614...寫入序列
304...第三示意圖	700...時間交插區塊

802～818...區塊

906...虛線

902、1002...柵格

-
.
:
:

發明摘要

※ 申請案號： 102128080

※ 申請日： 102/08/06

※IPC 分類： H04N 21/2315 (2011.01)
H03M 13/27 (2006.01)

【發明名稱】(中文/英文)

用於數位信號處理之以瓦狀塊為基礎的交插及解交插技術
TILE BASED INTERLEAVING AND DE-INTERLEAVING FOR
DIGITAL SIGNAL PROCESSING

【中文】

本案描述列-行交插資料的以瓦狀塊為基礎的交插及解交插。在一個實例中，解交插被分為兩個記憶體轉移階段，第一轉移階段自一晶載記憶體至一DRAM，且第二轉移階段自該DRAM至一晶載記憶體。每一階段對資料的列-行交插區塊的一部分進行操作且對資料項重新排序，使得第二階段之輸出包含解交插資料。在第一階段中，根據記憶體讀取位址的一非線性序列自該晶載記憶體讀取資料項且將其寫入至該DRAM。在第二階段中，根據有效率地使用DRAM介面之線性位址序列之叢發自該DRAM讀取資料項且根據記憶體寫入位址的一非線性序列將其寫回至晶載記憶體。

【英文】

Tile based interleaving and de-interleaving of row-column interleaved data is described. In one example, the de-interleaving is divided into two memory transfer stages, the first from an on-chip memory to a DRAM and the second from the DRAM to an on-chip memory. Each stage operates on part of a row-column interleaved block of data and re-orders the data items, such that the output of the second stage comprises de-interleaved data. In the first stage, data items are read from the on-chip memory according to a non-linear sequence of memory read addresses and written to the DRAM. In the second stage, data items are read from the DRAM according to bursts of linear address sequences which make efficient use of the DRAM interface and written back to on-chip memory according to a non-linear sequence of memory write addresses.

圖式

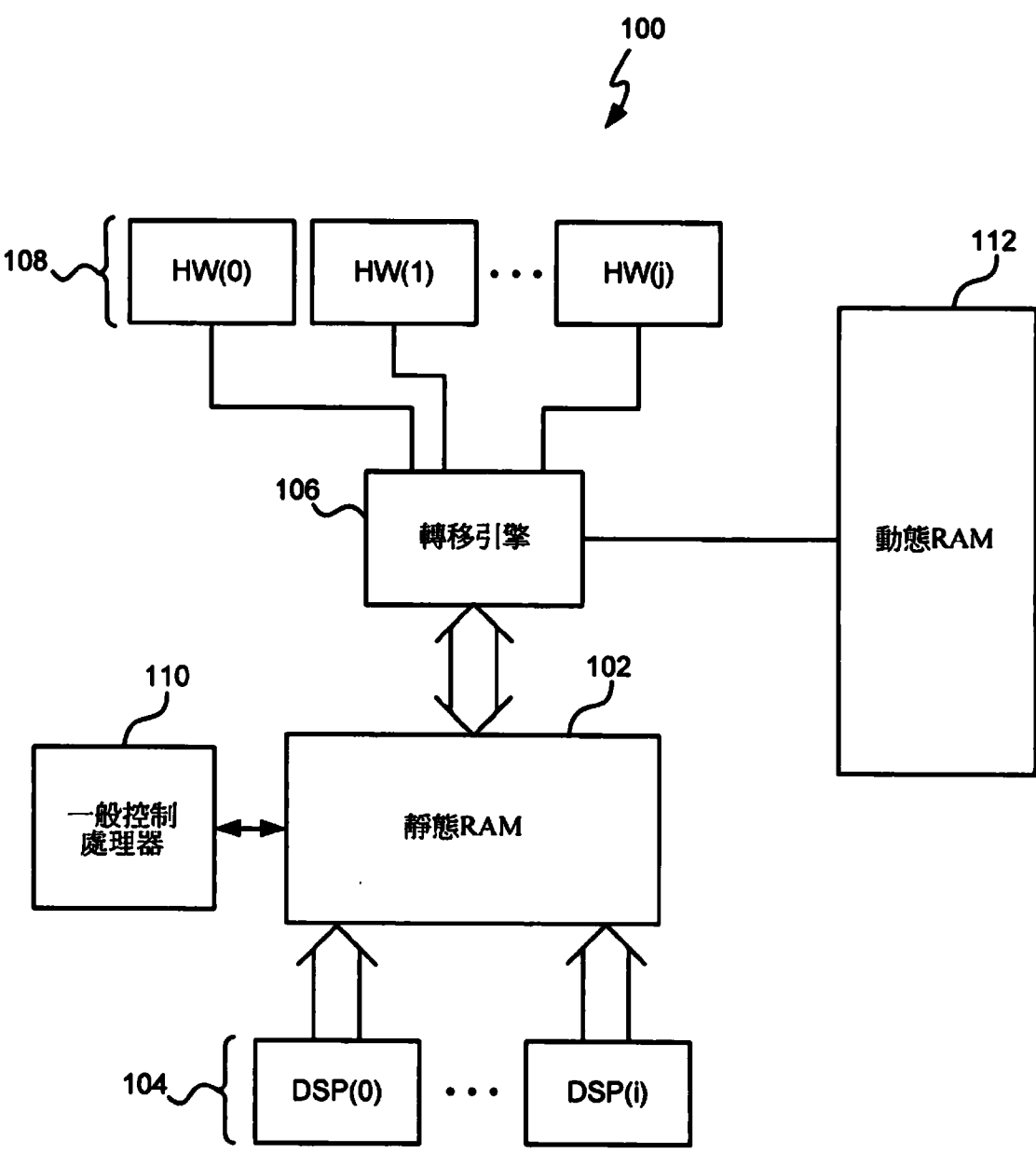


圖1

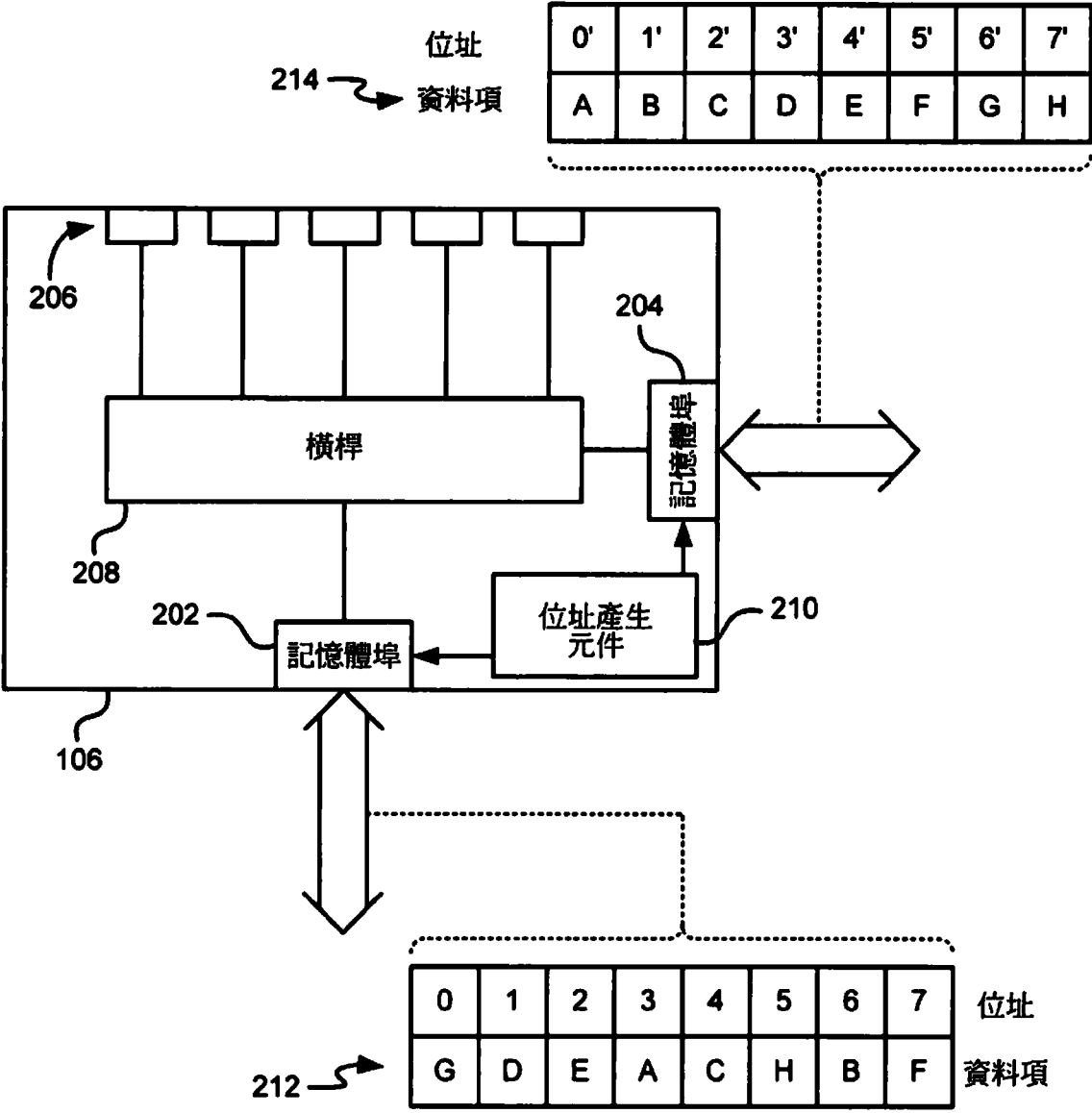


圖2

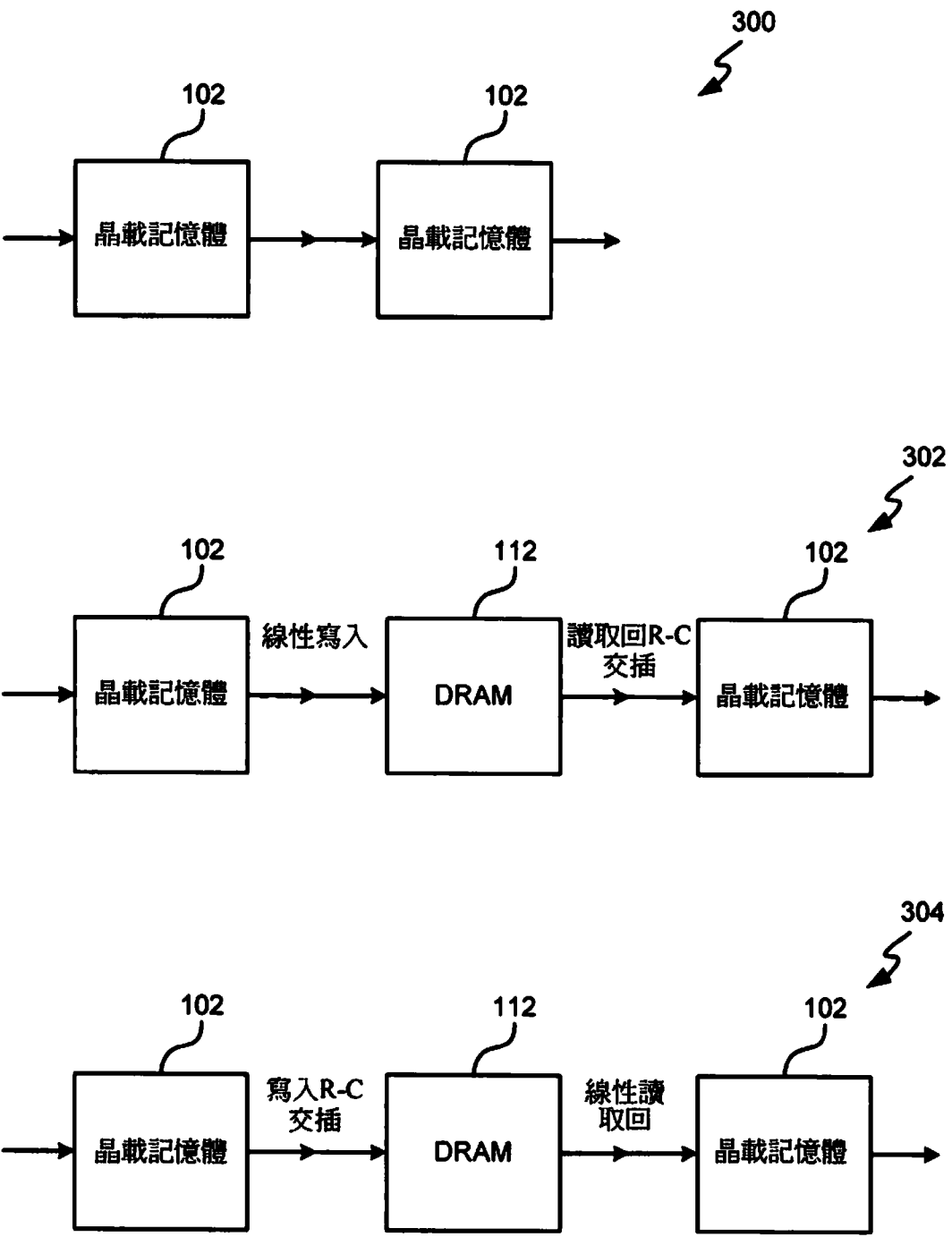


圖3

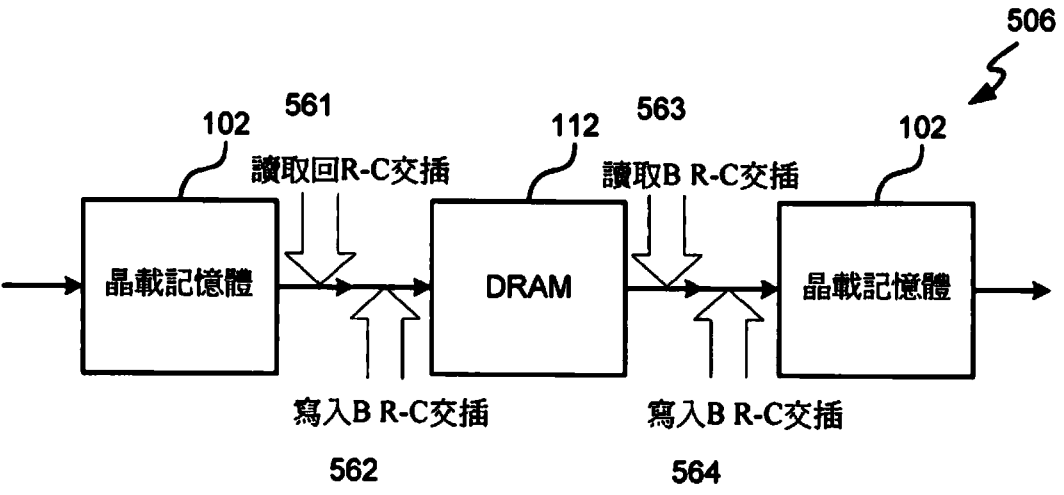
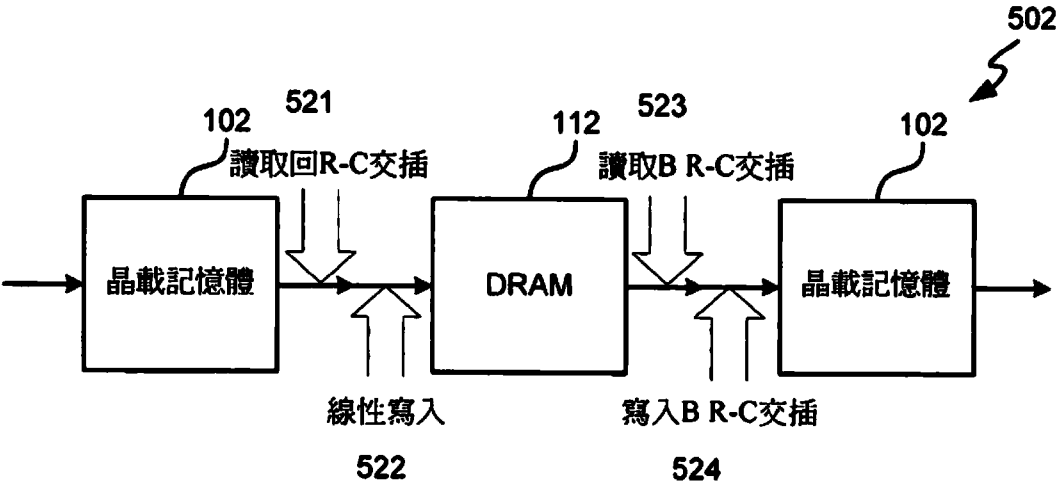


圖5

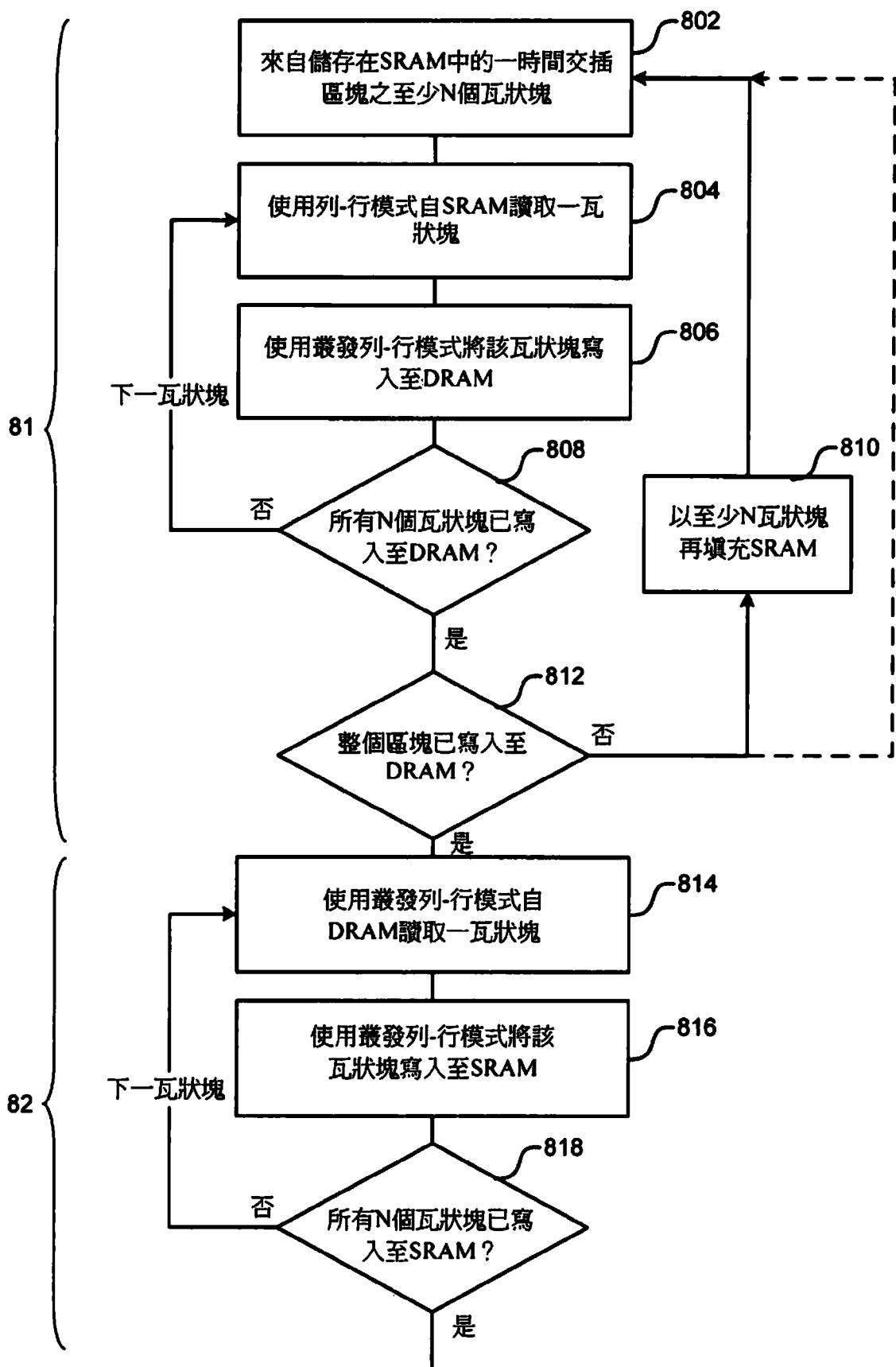


圖8

【代表圖】

【本案指定代表圖】：第（5）圖。

【本代表圖之符號簡單說明】：

102...晶載記憶體/記憶裝置

112...DRAM

502...第一示意圖

506...第二示意圖

521～524...箭頭

561～564...箭頭

【本案若有化學式時，請揭示最能顯示發明特徵的化學式】：

(無)

申請專利範圍

1. 一種數位信號處理系統單晶片，其包含：

一第一記憶體，其儲存以一第一序列來配置之多個資料項，其中每一資料項在該第一記憶體上具有一相關聯的記憶位址；

一第二記憶體；以及

一轉移引擎，其耦接至該第一記憶體及該第二記憶體且包含一埠，該埠接至一動態隨機存取記憶體 (DRAM)，其中該轉移引擎組配來在一第一記憶體轉移階段中將該等多個資料項自該第一記憶體直接轉移至該 DRAM，且在一第二記憶體轉移階段中將該等多個資料項自該 DRAM 直接轉移至該第二記憶體，

其中在該第一記憶體轉移階段中，該轉移引擎配置來根據記憶體讀取位址的一預定非線性序列自該第一記憶體讀取該等多個資料項且將該等多個資料項寫入至該 DRAM，以及

其中在該第二記憶體轉移階段中，該轉移引擎配置來根據多個線性位址序列之叢組自該 DRAM 讀取該等多個資料項，其中線性位址序列之每一叢組具有基於一 DRAM 介面叢組大小選擇的一長度，且根據記憶體寫入位址的一預定非線性序列將該等多個資料項寫入至該第二記憶體，使得該等多個資料項在該第二記憶體上以一第二序列來配置，該第二序列不同於該第一序列，且

其中該第一序列及該第二序列中之一者包含列-行交插資料，且其中該第二序列相對於該第一序列接受列-行交插或是解交插。

- 5 2. 如請求項 1 之數位信號處理系統單晶片，其中該第一記憶體及該第二記憶體皆為靜態隨機存取記憶體。
3. 如請求項 1 之數位信號處理系統單晶片，其中該第一記憶體及該第二記憶體為同一晶載記憶體。
4. 如請求項 1 之數位信號處理系統單晶片，其進一步包括該 DRAM。
- 10 5. 如請求項 1 之數位信號處理系統單晶片，其中該等多個資料項包含一資料項區塊之一子集，且該轉移引擎被進一步配置來重複該第一記憶體轉移階段及該第二記憶體轉移階段，直至該資料項區塊已全部寫入至該第二記憶體。
- 15 6. 如請求項 1 之數位信號處理系統單晶片，其進一步包含至少一個位址產生元件，其配置來產生記憶體讀取位址的該預定非線性序列及記憶體寫入位址的該預定非線性序列。
7. 如請求項 1 之數位信號處理系統單晶片，其中該等多個資料項包含一資料項區塊之一子集，且該資料項區塊被定義為配置為一柵格，該柵格包含許多列資料項及許多行資料項。
- 20 8. 如請求項 7 之數位信號處理系統單晶片，其中該柵格進一步包含多個瓦狀塊，每一瓦狀塊包含該柵格的一矩形

部分且進一步包含 R 列資料項及 C 行資料項，且其中該等多個資料項包含一或多個瓦狀塊。

5 9. 如請求項 8 之數位信號處理系統單晶片，其中記憶體讀取位址的該預定非線性序列針對第一多個資料項中的每一瓦狀塊包含：非連續記憶位址之一序列，該等非連續記憶位址由固定數目個記憶體位址分開且以一初始起始位址開始，該固定數目對應於比該柵格中的列數少一個，直至到達該瓦狀塊之一邊界；後續接著非連續記憶位址之一或多個額外序列，其中每一額外序列以一位移後的初始起始位址開始。

10 10. 如請求項 8 之數位信號處理系統單晶片，其中記憶體寫入位址的該預定非線性序列包含：C 個連續記憶位址之群組之一序列，該等位址在該第二記憶體中由固定數目個記憶體位址分開且在該第二記憶體中以一初始起始位址開始，該固定數目對應於比該柵格中的行數少 C 個。

15 11. 如請求項 8 之數位信號處理系統單晶片，其中該等多個資料項包含該柵格之一瓦狀塊。

20 12. 如請求項 8 之數位信號處理系統單晶片，其中在該第二記憶體轉移階段中，線性位址序列之該叢組包含 X 個連續記憶位址之叢組之一序列，該等位址在該第二記憶體中由固定數目個記憶體位址分開且在該第二記憶體中以一初始起始位址開始，其中 X 等於該柵格之一瓦狀塊中的資料項數。

13. 如請求項 8 之數位信號處理系統單晶片，其中在該第一

記憶體轉移階段中，該轉移引擎配置來根據線性位址序列之叢組將該等多個資料項寫入至該 DRAM，其中線性位址序列之每一叢組具有基於一 DRAM 介面叢組大小來選擇的一長度。

5 14. 如請求項 13 之數位信號處理系統單晶片，其中在該第一記憶體轉移階段中，線性位址序列之該叢組包含 X 個連續記憶位址之叢組之一序列，該等位址在該第二記憶體中由固定數目個記憶位址分開且在該第二記憶體中以一初始起始位址開始，其中 X 等於該柵格之一瓦狀塊中的資料項數。

10 15. 如請求項 8 之數位信號處理系統單晶片，其中一瓦狀塊係基於該 DRAM 介面叢組之一大小來設定大小。

16. 一種在數位信號處理系統中對資料項執行交插或解交插操作的方法，該方法包含：

15 根據記憶體讀取位址的一預定非線性序列自一第一晶載記憶體讀取第一多個資料項，該等第一多個資料項係按一第一序列來儲存；

將該等第一多個資料項寫入至一動態隨機存取記憶體(DRAM)；

20 根據線性位址序列之叢組自該 DRAM 讀取該等第一多個資料項，其中線性位址序列之每一叢組具有基於一 DRAM 介面叢組大小來選擇的一長度；以及

根據記憶體寫入位址的一預定非線性序列將該等第一多個資料項寫入至一第二晶載記憶體，使得該等資

料項在該第二晶載記憶體上以一第二序列來配置，該第二序列不同於該第一序列，且其中該第一序列及該第二序列中之一者包含列-行交插資料，且其中該第二序列相對於該第一序列接受列-行交插或是解交插。

- 5 17. 如請求項 16 之方法，其中該等第一多個資料項包含一資料項區塊之一子集，其中該資料項區塊係定義為配置為一柵格，該柵格包含許多列資料項及許多行資料項，該柵格進一步包含多個瓦狀塊，每一瓦狀塊包含該柵格的一矩形部分且進一步包含 R 列資料項及 C 行資料項，且其中該等第一多個資料項包含一或多個瓦狀塊，且其中根據記憶體讀取位址的一預定非線性序列自一第一晶載記憶體讀取以一第一序列來儲存的第一多個資料項針對該等第一多個資料項中的每一瓦狀塊包含：
- 10

(i)讀取在該第一晶載記憶體中的一初始起始位址之一資料項；

15

(ii)跳過固定數目個資料項，該固定數目對應於比該柵格中的列數少一個；

(iii)讀取一資料項；

(iv)重複步驟(ii)及(iii)，直至到達該瓦狀塊之一邊界；

20

(v)對該初始起始位址加上一位移；以及

(vi)重複步驟(i)至(v)，直至已讀取該瓦狀塊中的每一資料項。

18. 如請求項 16 之方法，其中該等第一多個資料項包含一

資料項區塊之一子集，其中該資料項區塊係定義為配置為一柵格，該柵格包含許多列資料項及許多行資料項，該柵格進一步包含多個瓦狀塊，每一瓦狀塊包含該柵格的一矩形部分且進一步包含 R 列資料項及 C 行資料項，且其中該等第一多個資料項包含一或多個瓦狀塊，且其中根據記憶體寫入位址的一預定非線性序列將該等第一多個資料項寫入至一第二晶載記憶體包含：

(i)將來自該等第一多個資料項之 C 個資料項寫入至該第二晶載記憶體中的多個連續位址，其中針對該瓦狀塊，以該第二晶載記憶體中的一初始起始位址開始；

(ii)跳過該第二晶載記憶體中的固定數目個位址，該固定數目對應於比該柵格中的行數少 C 個；

(iii)將來自該等第一多個資料項之 C 個資料項寫入至該第二晶載記憶體中的多個連續位址；以及

(iv)重複步驟(ii)及(iii)。

19. 如請求項 16 之方法，其中該等第一多個資料項包含一資料項區塊之一子集，其中該資料項區塊係定義為配置為一柵格，該柵格包含許多列資料項及許多行資料項，該柵格進一步包含多個瓦狀塊，每一瓦狀塊包含該柵格的一矩形部分且進一步包含 R 列資料項及 C 行資料項，且其中該等第一多個資料項包含一或多個瓦狀塊，且其中將該等第一多個資料項寫入至該 DRAM 包含：

(i)將來自該等第一多個資料項之 X 個資料項寫入至該 DRAM 中的多個連續位址，其中針對該瓦狀塊，

以該 DRAM 中的一初始起始位址開始；

(ii)跳過該 DRAM 中的固定數目個位址；

(iii)將來自該等第一多個資料項之 X 個資料項寫入至該 DRAM 中的多個連續位址；以及

5 (iv)重複步驟(ii)及(iii)，其中 X 等於該柵格之一瓦狀塊中的資料項數。

20. 如請求項 16 之方法，其中該等第一多個資料項包含一資料項區塊之一子集，其中該資料項區塊係定義為配置為一柵格，該柵格包含許多列資料項及許多行資料項，
10 該柵格進一步包含多個瓦狀塊，每一瓦狀塊包含該柵格的一矩形部分且進一步包含 R 列資料項及 C 行資料項，且其中該等第一多個資料項包含一或多個瓦狀塊，且其中根據線性位址序列之叢組自該 DRAM 讀取該等第一多個資料項包含：

15 (i)自該 DRAM 中的多個連續位址讀取來自該等第一多個資料項之 X 個資料項，其中以該 DRAM 中的一初始起始位址開始；

(ii)跳過該 DRAM 中的固定數目個位址；

(iii)自該 DRAM 中的多個連續位址讀取來自該等第一多個資料項之 X 個資料項；以及

20 (iv)重複步驟(ii)及(iii)，其中 X 等於該柵格之一瓦狀塊中的資料項數。