

(此處由本局於收
文時黏貼條碼)**發明專利說明書**

99年12月8日修正替換頁

公告本

(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：096123333

※申請日期：96年06月27日

※IPC分類：G06F 12/00 (2006.01)

一、發明名稱：

(中) 用於虛擬化交易式記憶體之總體溢位之裝置、方法與系統

(英) Apparatus, method and system for global overflow in a virtualized transactional memory

二、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 2 2 0 0 號

(英) 2200 Mission College Blvd., Santa Clara, CA 95052, USA

國籍：(中英) 美國 U.S.A.

三、發明人：(共 2 人)1. 姓名：(中) 潔西 巴尼斯
(英) BARNES, JESSE

國籍：(中) 美國

(英) U.S.A.

2. 姓名：(中) 瑞維 洛傑沃
(英) RAJWAR, RAVI

國籍：(中) 印度

(英) INDIA

四、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 ; 2006/06/30 ; 11/479,902 有主張優先權

五、中文發明摘要

發明之名稱：用於虛擬化交易式記憶體之總體溢位之裝置、方法與系統

本文描述用於虛擬化及/或延伸交易式記憶體的方法及裝置。交易係使用局部共用交易式記憶體來執行，使用如快取記憶體。在該共用交易式記憶體溢位時，該交易式記憶體被虛擬化及/或延伸進入較高階的記憶體，諸如系統記憶體。在溢位事件時，諸如在目前交易未決期間，諸如先前被存取之快取記憶體線的逐出(eviction)，溢位旗標被設定以通知處理器/核心，該交易式記憶體將被虛擬化在總體溢位表中。該總體溢位表的基礎位址也潛在地被儲存，以參考該較高階記憶體中之該總體溢位表的基礎。

六、英文發明摘要

發明之名稱： **APPARATUS, METHOD AND SYSTEM FOR GLOBAL OVERFLOW
IN A VIRTUALIZED TRANSACTIONAL MEMORY**

A method and apparatus for virtualizing and/or extending transactional memory is described herein. Transactions are executed using local shared transactional memory, such as a cache memory. Upon overflowing the shared transactional memory, the transactional memory is virtualized and/or extended into a higher-level memory, such as a system memory. Upon an overflow event, such as an eviction of a cache line previously accessed during a currently pending transaction, an overflow flag is set to notify processors/cores that the transactional memory is to be virtualized in a global overflow table. A base address of the global overflow table is also potentially stored to reference the base of the global overflow table in the higher-level memory.

七、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：

100：多核心處理器，110,115：執行單元，
120,121：排程器，
160,165,170,175：執行緒，
140,141：提取解碼方塊，
150：匯流排介面單元，
145：較高階的快取記憶體，
135：微碼 ROM，136：微碼唯讀記憶體，
130,131：配置器更名器方塊，
125,126：重排序/止用單元，
103,108：較低階的快取記憶體，
104,105,106：快取記憶體線，
104a,105a,106a：記憶體線，
104b,105b,106b：存取追蹤欄，
107,109：溢位模組。

八、本案若有化學式時，請揭示最能顯示發明特徵的化學式：無

九、發明說明

【發明所屬之技術領域】

本發明與處理器執行的領域且特別是執行操作群組有關。

【先前技術】

在半導體處理及邏輯設計方面的進展，已允許存在於積體電路裝置上的邏輯數量增加。結果是，電腦系統組構已從系統中的單或多個積體電路，演進到存在於各個積體電路上的多核心及多邏輯處理器。處理器或積體電路典型上包含單個處理器晶模，而該處理器晶模可包括任意數量的核心或邏輯處理器。

例如，單一個積體電路可具有一或多個核心。核心一詞，通常意指在積體電路上有能力保持一獨立架構狀態的邏輯，其每一獨立的架構狀態與至少某些專用的執行資源相關。至於另一例，單積體電路或單核心可具有用以執行多軟體執行緒(thread)的多硬體執行緒，其也被稱為多執行緒積體電路或多執行緒核心。多硬體執行緒通常分享公用資料快取記憶體、指令快取記憶體、執行單元、分支預測器、控制邏輯、匯流排介面、及其它處理器資源，同時為每一個邏輯處理器保持獨有的架構狀態。

只要增加積體電路上之核心及邏輯處理器的數量，就能夠執行更多的軟體執行緒。不過，可能同時執行之軟體執行緒的數量增加，會產生該等軟體執行緒間同時共用資

料的問題。解決多核心或多邏輯處理器系統中存取共用資料的常見方法之一包含鎖的使用，以保證對共用資料之多存取間的互斥。不過，無止境地增加執行多軟體執行緒的能力，有可能導致錯誤的爭用及執行的序列化。

另一項資料同步技術包括交易式記憶體(TM)的使用。通常，交易的執行包括推測地執行一群複數個微操作、操作、或指令。不過，在先前的硬體式 TM 系統中，如果交易對於記憶體而言變得太大，即溢位，則該交易通常會被重新開始。在此，花在執行交易直至溢位的時間是潛在的浪費。

【發明內容及實施方式】

在以下的描述中，將提出很多特定的細節，諸如用以支援交易執行之特定硬體的例子，處理器中特定類型的局部/記憶體，以及特定類型的記憶體存取及所在位置等，以便提供對本發明的全盤瞭解。不過，很明顯，對於熟悉此方面技術的人士而言，實用本發明並不需要使用這些特定細節。在其它的例中，已為吾人所熟知的組件或方法，諸如在軟體中交易的編碼、交易的劃界(demarcation)、多核心及多執行緒處理器的特定架構、中斷產生/處理、快取記憶體組織、及微處理器的特定操作細節等，都不詳細描述，以避免對本發明造成不必要的混淆。

本文所描述的方法及裝置係用於延伸及/或虛擬化交易式記憶體(TM)，以支援交易之執行期間區域記憶體的溢

位。特別是，虛擬化及/或延伸交易式記憶體，主要是參考多核心處理器的電腦系統來討論。不過，用於延伸/虛擬化交易式記憶體的方法及裝置並無制，其可在任何積體電路裝置或系統上實施或與其結合，諸如細胞式電話、個人數位式助理、內嵌式控制器、線動平台、桌上型平台、及伺服器平台、以及與其它資源結合，諸如利用交易式記憶體的硬體/軟體執行緒。

現請參閱圖 1，圖中說明多核心處理器 100 的實施例，其具有延伸交易式記憶體的能力。交易式執行通常包括將複數個指令或操作分類成爲一交易、碼的基元區段、碼的關鍵區段。在某些情況中，文字指令的使用，意指係由複數個操作所組成的巨集指令。用以識別交易的方法通常有二。第一例包括在軟體中將交易劃界。在此，某些軟體劃界被包括在碼中，以識別一交易。在另一實施例中，可結合前述的軟體劃界，交易藉由硬體來分類，或由指示交易之開始與交易之結束的指令來組織。

在處理器中，交易可用推測或非推測地來執行。在第二情況中，指令群係以某種型式的鎖來執行，或保證對要被存取之記憶體位置的有效存取。在另一選擇中，交易的推測執行更爲常見，交易係被推測地執行，並在該交易結束時被確定。如在本文中所使用的交易的未決定，意指一交易已開始執行，且尚未被確定或中止，即懸而未決。

典型上，在交易的推測執行期間，直至該交易被確定前，對記憶體的更新無法做到總體地可見。當該交易仍在

未決期間，從記憶體載入及寫入記憶體的位置被追蹤。在這些記憶體位置的確認成功時，在該交易做到總體可見的期間，該交易被確定並做更新。不過，如果該交易在此未決期間被無效，該交易被重新開始，不做更新的總體可見。

在說明的實施例中，處理器 100 包括 2 個核心，即核心 101 及 102；雖然可存在有任何數量的核心。核心通常指的是位於積體電路上之任何有能力保持獨立架構狀態的邏輯，其中每一個獨立保持的架構狀態與至少一個專用的執行資源相關。例如，在圖 1 中，核心 101 包括執行單元 110，而核心 102 包括執行單元 115。即使執行單元 110 與 115 是邏輯地分開描述，但它們可實體地配置成同一個單元的一部分，或緊鄰在一起。不過，例如，在執行單元 115 上，排程器 120 無法為核心 101 執行排程。

相對於核心，硬體執行緒典型上指的是位在積體電路上能夠保持獨立架構狀態的任何邏輯，其中，該獨立保持的架構狀態對執行資源共用存取。如所見，關於某些處理資源被共用而其它則為一架構狀態所專用，硬體執行緒與核心之命名間的界線重疊。然而，核心與硬體執行緒通常被作業系統視為個別的邏輯處理器，每一個邏輯處理器具有執行一個執行緒的能力。因此，處理器（諸如處理器 100）具有執行多執行緒的能力，諸如執行緒 160、165、170、及 175。雖然所說明的每一個核心（諸如核心 101）具有執行多軟體執行緒的能力，諸如執行緒 160 及 165，但

一核心也可能只有執行單一個執行緒的能力。

在一實施例中，處理器 100 包括對稱的核心 101 及 102。在此，核心 101 及 102 係類似的核心，具有類似的組件及架構。或者，核心 101 及 102 可以是具有不同組件及架構的非對稱核心。然而，現在以對稱的核心來描述核心 101 及 102，將討論核心 101 中的功能方塊，關於核心 102 則避免重複的討論。須注意，所說明的功能方塊係邏輯功能方塊，其可包括可與其它功能方塊間共用或邊界重疊的邏輯。此外，每一個功能方塊並不需要但有可能以不同的結構互連。例如，提取及解碼方塊 140 可包括提取及/或預提取單元，解碼單元耦接至該提取單元，且指令快取記憶體耦接在提取單元之前、解碼單元之後、或與提取及解碼單元兩者耦接。

在一實施例中，處理器 100 包括匯流排介面單元 150，用以與外部裝置及較高階的快取記憶體 145 通信，諸如第二階的快取記憶體，其為核心 101 與 102 間所共用。在另一實施例中，核心 101 與 102 每一個都包括各自獨立的第二階快取記憶體。

提取、解碼、及分支預測單元 140 耦接至第二階快取記憶體 145。在一例中，核心 101 包括用以提取指令的提取單元，用以解碼被提取之指令的解碼單元，以及用以儲存被提取之指令、被解碼之指令、或被提取與被解碼之指令之組合的指令快取記憶體或追蹤快取記憶體。在另一實施例中，提取及解碼方塊 140 包括具有分支預測器及/或

分支目標緩衝器的預提取器。此外，唯讀記憶體(諸如微碼 ROM 135)也有可能用來儲存較長或較複雜之經解碼的指令。

在一例中，配置器及更名器方塊 130 包括用以保留資源的配置器，諸如用來儲存指令處理結果的暫存器檔。不過，核心 101 有可能具有亂序執行的能力，此時，配置器及更名器方塊 130 也保留其它資源，諸如用來追蹤指令的重排序緩衝器。方塊 130 也可包括暫存器更名器，用以將程式/指令參考暫存器更名為核心 101 內部的暫存器。重排序/止用單元 125 包括諸如上述重排序緩衝器等組件，用以支援亂序執行，及亂序執行過之指令稍後的止用。如例所示，被載入重排序緩衝器中的微操作被執行單元亂序地執行，並接著按照該等微操作進入重排序緩衝器之相同的順序被搬出重排序緩衝器，即止用。

在本實施例中，排程器及暫存器檔方塊 120 包括用以在執行單元 110 上排程指令的排程器單元。事實上，指令有可能按照其類型及執行單元 110 的可用度在執行單元 110 上被排程。例如，執行單元 110 具有一可用的浮點執行單元，則浮點指令在執行單元 110 的埠上被排程。執行單元 110 也包括相關的暫存器檔，用來儲存資訊指令處理的結果。核心 101 中可用的例示性執行單元包括浮點執行單元、整數執行單元、跳躍執行單元、載入執行單元、儲存執行單元、及其它習知的執行單元。在一實施例中，執行單元 110 也包括保留站(reservation station)及/或位址產

生單元。

在說明的實施例中，較低階的快取記憶體 103 被利用做為交易式記憶體，特別是，較低的階快取記憶體 103 係用來儲存關於元件之最近的使用/操作，諸如運算元。快取記憶體 103 包括快取記憶體線，諸如線 104、105、及 106，其也可指的是快取記憶體 103 內的記憶體位置或區塊。在一實施例中，快取記憶體 103 被組織成關聯的快取記憶體組；不過，快取記憶體 103 也可組織成完整的關聯、組關聯、直接映射、或其它已知的快取記憶體組織。

如圖示說明，線 104、105、及 106 包括部或欄，諸如部 104a 及欄 104b。在一實施例中，線、位置、區塊或字元，諸如線 104、105、及 106 的部分 104a、105a、及 106a 能夠儲存多個元件。元件意指通常儲存在記憶體中的任何指令、運算元、資料運算元、變數、或其它邏輯值的群組。例如，快取記憶體線 104 在部 104a 中儲存 4 個元件，包括 1 個指令及 3 個運算元。儲存在記憶體線 104a 中的元件可為包裹或壓縮狀態、以及未壓縮狀態。此外，儲存在快取記憶體 103 中的元件有可能不與快取記憶體 103 之線、組、或路徑 (ways) 的邊界對齊。以下將參考例示性實施例更詳細討論記憶體 103。

快取記憶體 103 以及處理器 100 中的其它特徵與裝置儲存及/或操作邏輯值。通常，使用邏輯位準、邏輯值、或邏輯上的值也意指 1 及 0，其單純地代表二進位的邏輯狀態。例如，1 意指高邏輯位準及 0 意指低邏輯位準。在

電腦系統中也使用其它的值表示法，諸如邏輯值或二進位值的 10 進位及 16 進位表示法。例如十進位的值 10，在二進位值中以 1010 表示，在 16 進位中以字母 A 表示。

在圖 1 所說明的實施例中，追蹤對於線 104、105、及 106 的存取以支援交易的執行。諸如欄 104b、105b、及 106b 等存取追蹤欄被用來追蹤對於與其所對應之記憶體線的存取。例如，記憶體線/部 104a 與對應的追蹤欄 104b 相關。在此，存取追蹤欄 104b 與快取記憶體線 104a 相關並對應，例如追蹤欄 104b 包括了快取記憶體線 104 的部分位元。相關可透過實體配置，如圖示說明，或其它相關，諸如以位址來參考記憶體線 104a 或硬體中的 104b 或軟體速查表來關連或映射存取追蹤欄 104b。事實上，交易存取欄係在硬體、軟體、韌體或以上這些的任意組合中實施。

因此，在交易的執行期間存取線 104a 時，存取追蹤欄 104b 追蹤該存取。存取包括操作，諸如讀、寫、儲存、載入、逐出、監聽 (snoop)、或其它對記憶體位置之習知的存取。

例如簡化的說明例，假設存取追蹤欄 104b、105b、及 105b 中包括兩個交易位元，即：第一讀追蹤位元及第二寫追蹤位元。在原設狀態中，即第一邏輯值，存取追蹤欄 104b、105b、及 105b 中的第一及第二位元分別代表快取記憶體線 104、105、及 106 在交易的執行期間未被存取，即，在交易的未決定期間。在從快取記憶體線 104a，或與

快取記憶體線 104a 相關之系統記憶體位置的載入操作導致從線 104a 載入時，存取欄 104b 中的第一讀追蹤位元被設定成第二狀態 / 值，諸如第二邏輯值，用以代表在交易的執行期間已發生從快取記憶體線 104 的讀取。同樣地，在寫到快取記憶體線 105a 時，存取欄 105b 中的第二寫追蹤位元被設定成第二狀態，以代表在交易的執行期間發生寫到快取記憶體線 105。

因此，如果檢查與線 104a 相關之欄 104a 中的交易位元，且該交易位元表現原設狀態，則在交易的未決定期間快取記憶體線 104 未被存取。反之，如果該第一讀取追蹤位元表現第二值，則快取記憶體線 104 在交易的未決定期間已被前一存取。更明確地，在交易的執行期間發生從線 104a 的載入，例如以存取欄 104b 中被設定的第一讀取追蹤位元表示。

在交易的執行期間，存取欄 104b、105b、及 105b 也具有其它用途。例如，交易的確認傳統上以兩種方式完成。第一，如果追蹤到會造成交易放棄的無效存取，則在該無效存取之時放棄該交易，且可能重新開始。另者，在確定前，在該交易結束時完成交易執行期間之線 / 位置之存取の確認。在此時刻，如果確認成功，或如果該確認不成功而被放棄，則該交易被確定。在這兩種情形中，以存取追蹤欄 104b、105b、及 105b 來識別在交易的執行期間那一條線已被存取很有用處。

例如另一簡化的說明例，假設第一交易正被執行中，

且在該第一交易的執行期間發生從線 105a 的載入。結果是，對應的存取追蹤欄 105b 指示，在交易的執行期間發生對於線 105 的存取。由於存取追蹤欄 105b 表示該線 105 被第一未決定的交易載入，如果第二交易造成關於線 105 的衝突，則根據第二交易對線 105 的存取，立刻放棄第一或第二交易。

在一實施例中，有對應的欄 105b 指示線 105 被第一未決定的交易前一存取，則在第二交易造成關於線 105 的衝突時產生一中斷。當兩個未決定的交易間發生衝突時，該中斷被原設處置器及 / 或用於初始化該第一或第二交易之放棄的放棄處置器處置。

交易一旦放棄或確定，在交易之執行期間所設定的交易位元被清除，以確保該交易位元的狀態被重置到原設狀態，以供後續交易期間之稍後的存取追蹤。在另一實施例中，存取追蹤欄也可儲存資源 ID，諸如核心 ID 或執行緒 ID，以及交易 ID。

關於以上及以下即將參考圖 1 所提及，利用較低階的快取記憶體 103 做為交易式記憶體。不過，交易式記憶體並無此限制。事實上，也有可能使用較高階的快取記憶體 145 做為交易式記憶體。在此，對於快取記憶體 145 之線的存取被追蹤。如所述，在較高階記憶體 (諸如快取記憶體 145) 中有可能使用諸如執行緒 ID 或交易 ID 等識別器，在快取記憶體 145 中追蹤那一個交易、執行緒或資源實施存取。

可能的交易式記憶體還有另外的例子，與處理元件相關的複數個暫存器，或做為執行空間的資源，或用於儲存變數、指令、或資料的暫存記憶體，都可用做為交易式記憶體。在此例中，記憶體位置 104、105、及 106 係一組暫存器，包括暫存器 104、105、及 106。交易式記憶體的其它例子包括快取記憶體、複數個暫存器、暫存器檔 (register file)、靜態隨機存取記憶體 (SRAM)、複數個鎖存器、或其它儲存元件。須注意，當讀取或寫入一記憶體位置時，處理器 100 或處理器 100 上的任何處理資源都可定址一系統記憶體位置、虛擬記憶體位址、實體位址、或其它位址。

只要交易不使交易式記憶體 (諸如較低階的快取記憶體 103) 溢位，則各交易間的衝突，由存取欄 104b、105b、及 105b 分別追蹤對於對應之行 104、105、及 105 之存取的操作來偵測。如前所述，使用存取追蹤欄 104b、105b、及 105b 可使交易有效、確定、無效、及/或放棄。不過，當一交易使記憶體 103 溢位時，回應一溢位事件，溢位模組 107 被用來支援交易式記憶體 103 的虛擬化及/或延伸，即，將該交易的狀態儲存到第二記憶體。記憶體 103 溢位時即放棄該交易，其致使與交易中先前執行之操作相關之執行時間的損失，因此，以虛擬化該交易狀態而繼續執行來取代。

溢位事件可包括記憶體 103 之任何實際的溢位或記憶體 103 之溢位的任何預測。在一實施例中，溢位事件在記

記憶體 103 中選擇用於逐出或實際逐出在目前未決定之交易的執行期間被前一存取的線。換言之，一操作正在使已被目前未決定之交易存取之記憶體線填滿的記憶體 103 溢位。結果是，記憶體 103 選擇與未決定之交易相關之要被逐出的線。基本上，記憶體 103 被填滿，且嘗試藉由逐出與仍未決定之交易相關的線以產生空間。快取記憶體的取代、線的逐出、確定、存取追蹤、交易衝突檢查及交易確認，可用已知或其它可用的技術。

不過，溢位事件並不限於記憶體 103 的實際溢位。例如，預測一交易對記憶體 103 而言太大也可構成溢位事件。在此，使用演算法或其它預測方法來決定交易的大小，並在記憶體 103 被實際溢位前先產生溢位事件。在另一實施例中，溢位事件是一巢套式交易的開始。關於巢套式交易係更複雜，且要取用較多的記憶體來支援，第一階巢套式交易或後續階巢套式交易的偵測可能導致溢位事件。

在一實施例中，溢位邏輯 107 包括用以儲存溢位位元的溢位儲存元件，諸如暫存器，以及基礎位址儲存元件。雖然是以與快取記憶體控制邏輯同一個功能方塊來說明溢位邏輯 107，但用以儲存溢位位元的暫存器及基礎位址暫存器有可能存在於處理器 100 中的任何位置。例如，處理器 100 上的每一個核心都包括有溢位暫存器，用以儲存總體溢位表之基礎位址的表示及溢位位元。不過，實施溢位位元與基礎位址並無此限制。事實上，為處理器 100 上之所有核心或執行緒可見的總體暫存器可包括溢位位元及基

礎位址。或者，每一核心或硬體執行緒包括一實體位址暫存器及一包括溢位位元的總體暫存器。如所見，可實施任何數量的組構來為溢位表儲存溢位位元及基礎位址。

溢位位元係根據溢位事件來設定。接續上述的實施例，在記憶體 103 中選擇在未決定之交易的執行期間已被前一存取而構成溢位事件的線用於逐出，該溢位位元係根據記憶體 103 中所選擇之用於逐出的線來設定，該用於逐出的線在未決定之交易的執行期間已被前一存取。

在一實施例中，溢位位元係使用硬體來設定，諸如當一線（諸如線 104）被選擇用於逐出且在未決定的交易期間已被前一存取時，以邏輯來設定溢位位元。例如，快取記憶體控制器 107 根據任何數量之已知或其它可用的快取記憶體替換演算法來選擇用於逐出的線 104。事實上，快取記憶體替換演算法可能傾向不取代在未決定之交易的執行期間已被前一存取的快取記憶體線（諸如線 104）。儘管如此，在選擇用於逐出的線 104 時，快取記憶體控制器或其它邏輯會檢查存取追蹤欄 104b。邏輯根據欄 104b 中的值來決定快取記憶體線 104 在未決定之交易的執行期間是否已被存取，如前文中的討論。如果快取記憶體線 104 在未決定之交易的執行期間已被前一存取，則處理器 100 中的邏輯設定總體溢位位元。

在另一實施例中，使用軟體或韌體來設定總體溢位位元。在類似的情況中，當決定線 104 在未決定之交易期間被前一存取時，即產生一中斷。該中斷被位在執行單元

110 中所執行的使用者處置器及/或其它放棄處置器處置，其設定總體溢位位元。須注意，如果該總體溢位位元目前被設定，即記憶體 103 已溢位，則該硬體及/或軟體不須再次設定該位元。

如用來說明溢位位元的例子，一旦溢位位元被設定，硬體及/或軟體即追蹤對於快取記憶體線 104、105、及 106 的存取、確認交易、檢查衝突，並執行其它與交易有關的操作，該等操作典型上與記憶體 103 及利用延伸交易記憶體的存取欄 104b、105b、及 106b 相關。

基礎位址被用來識別虛擬化交易式記憶體的基礎位址。在一實施例中，虛擬化交易式記憶體被儲存在第二記憶體裝置中，其為比記憶體 103 大的記憶體，諸如較高階的快取記憶體 145，或與處理器 100 相關的系統記憶體裝置。結果是，第二記憶體有能力處置使記憶體 103 溢位的交易。

在一實施例中，延伸的交易式記憶體意指用以儲存該交易之狀態的總體溢位表。因此，基礎位址代表該總體溢位表的基礎位址，其是用來儲存交易的狀態。總體溢位表類似於參考存取追蹤欄 104b、105b、及 106b 對記憶體 103 的操作。如說明例，假設線 106 被選擇用於逐出。不過，存取欄 106b 表示線 106 在未決定之交易的執行期間已被前一存取。如上所述，如果總體溢位位元尚未設定，則根據該溢位事件設定該總體溢位位元。

如果總體溢位表未被建立，則為該表配置第二記憶體

的量。例如，產生頁錯誤以指示該溢位表的初始頁尚未被配置。接著，作業系統配置第二記憶體的一範圍給該總體溢位表。第二記憶體的範圍可意指總體溢位表的頁。接著，該總體溢位表之基礎位址的表示被儲存在處理器 100 中。

在逐出線 106 之前，交易的狀態被儲存在總體溢位表中。在一實施例中，儲存交易的狀態包括將對應於與溢位事件相關之操作及/或線 106 的登錄列儲存於該總體溢位表中。該登錄列可包括與線 106 相關之任何位址的組合，諸如實體位址、存取追蹤欄 106b 的狀態、與線 106 相關的資料元件、線 106 的大小、作業系統控制欄、及/或其它欄位。以下將參考圖 3-5 更詳細討論總體溢位表及第二記憶體。

必然地，當為交易之一部分的指令或操作通過處理器 100 之管線時，對於交易式記憶體的存取(諸如快取記憶體 103)被追蹤。此外，當交易式記憶體被填滿時，即其溢位時，該交易式記憶體被延伸進入到位在處理器 100 上或與處理器 100 相關/耦接的其它記憶體。此外，整個處理器 100 的暫存器都有可能儲存用以表示該交易式記憶體已被溢位的溢位旗標，以及用以識別該延伸之交易式記憶體之基礎位址的基礎位址。

雖然已特別地參考圖 1 所示的例示性多核架構討論了交易式記憶體，但延伸及/或虛擬化交易式記憶體，可在用來對資料執行指令/操作的任何處理系統中實施。例如

，能夠平行執行多交易的內嵌式處理器，即有可能用來實施虛擬化的交易式記憶體。

現回到圖 2a，圖中說明多核心處理器 200 的實施例。在此，處理器 200 包括核心 205-208 等 4 個核心，但也可使用其它數量的核心。在一實施例中，記憶體 210 係快取記憶體。在此，圖示說明的記憶體 210 係在核心 205-208 之功能方塊的外部。在一實施例中，記憶體 210 是共用快取記憶體，諸如第二階或其它較高階的快取記憶體。不過，在一實施例中，功能方塊 205-208 代表核心 205-208 的架構狀態，且記憶體 210 是與該等核心其中之一(諸如核心 205)或核心 205-208 所指定/相關的第一階或較低階的快取記憶體。因此，如所說明，記憶體 210 可以是核心內的較低階快取記憶體，諸如圖 1 中所說明的記憶體 103，較高階的快取記憶體，諸如圖 1 中所說明的快取記憶體 145，或其它儲存元件，諸如以上所討論之暫存器之集合的例子。

每一個核心包括有暫存器，諸如暫存器 230、235、240、及 245。在一實施例中，暫存器 230、235、240、及 245 係特定機器暫存器(MSR)。然而，暫存器 230、235、240、及 245 可以是處理器 200 中的任何暫存器，諸如每一核心之架構狀態暫存器組中部分的暫存器。

每一個暫存器包括一交易溢位旗標：旗標 231、236、241、及 246。如上所述，在有溢位事件時，交易溢位旗標被設定。溢位旗標係經由硬體、軟體、韌體或其任意組合

來設定。在一實施例中，溢位旗標係一位元，其有可能具有兩個邏輯狀態。不過，溢位旗標可以是任何數量的位元，或當記憶體溢位時用以識別的其它狀態表示。

例如，如果在核心 205 上所執行做為交易之一部分的操作使快取記憶體 210 溢位，則硬體(諸如邏輯)或軟體(諸如使用者處置器)被引動以處置溢位中斷，設定旗標 231。在第一邏輯狀態(其為原設狀態)中，核心 205 使用記憶體 210 執行交易。一般使用快取記憶體 210 實施逐出、存取追蹤、衝突檢查、及確認，其包括方塊 215、220、及 225，以及對應的欄 216、221、及 226。不過，當旗標 231 被設定為第二狀態時，快取記憶體 210 被延伸。根據一旗標的被設定，諸如旗標 231，其餘的旗標 236、241、及 246 也跟著被設定。

例如，根據一溢位位元被設定，在核心 205-208 間傳送的協定訊息設定其它旗標。例如，假設溢位旗標 231 係根據發生於記憶體 210 中的溢位事件而被設定，在本例中，記憶體 210 為核心 205 中的第一階資料快取記憶體。在一實施例中，在設定旗標 231 之後，在互相連接核心 205-208 的匯流排上傳送廣播訊息用以設定旗標 236、241、及 246。在另一實施例中，核心 205-208 以點對點、環狀、或其它形式互相連接，來自核心 205 的訊息被送往每一個核心，或逐一核心向前傳送，以設定旗標 236、241、及 246。須注意，類似的訊息傳送等可在多處理器的形式中實施，以確保多個實體處理器間各旗標被設定，如下文中

的討論。當核心 205-208 中的旗標被設定時，後續的交易執行被告知，以便為存取追蹤、衝突檢查、及/或確認檢查虛擬/延伸記憶體。

先前的討論包括一包括有多核心的單實體處理器 200。不過，當核心 205-208 係分散於系統中之各分離的實體處理器時，也可使用類似的組構、協定、硬體、及軟體。在此例中，每一處理器具有一溢位暫存器，諸如具有各自之旗標的暫存器 230、235、240、及 245。一旦設定一個溢位旗標，其餘的溢位旗標也可在該等處理器間的互連上，經由協定通信之類似的方法來設定。在此，在廣播匯流排上或點對點互連的通信交換來傳遞被設定為代表溢位事件發生之值的溢位旗標值。

接下來請參閱圖 2b，圖中說明具有溢位旗標之多核心處理器的另一實施例。相對於圖 2a，在處理器 200 中只存在單個溢位暫存器 250 及溢位旗標 251，以取代每一核心 205-208 都包括有一溢位暫存器及溢位旗標。因此，在溢位事件時，旗標 251 被設定，且可被每一個核心 205-208 總體地可見。因此，如果旗標 251 被設定，則使用總體溢位表實施存取追蹤、確認、衝突檢查、及其它的交易執行操作。

如說明例，假設在交易的執行期間記憶體 210 已溢位，則結果是，暫存器 250 中的溢位位元 251 被設定。此外，後續的操作使用虛擬化交易式記憶體來追蹤。如果為了衝突或用於在確定一交易前之確認而僅檢查記憶體 210，

則追蹤溢位記憶體將不會發現衝突/存取。不過，如果是利用溢位記憶體來實施衝突檢查及確認，則該衝突可被偵測到，且該交易被放棄，取代對一衝突之交易的確定。

如前所述，在設定目前未被設定的溢位旗標時，如果尚未配置空間，則總體溢位表所需的空間被請求/配置。反之，當一交易被確定或放棄時，總體溢位表中對應於該交易的登錄列被釋放。在一實施例中，釋放一登錄列包括清除該登錄列中的存取追蹤狀態或其它欄位。在另一實施例中，釋放一登錄列包括從該總體溢位表中刪除該登錄列。當一溢位表中的最後登錄列被釋放時，總體溢位位元被清除而回到原設狀態。基本上，釋放總體溢位表中的最後登錄列，此代表任何未決定的交易都能裝入快取記憶體 210 中，且溢位記憶體目前未用於交易的執行。圖 3-5 更詳細討論溢位記憶體，且特別是總體溢位表。

現回到圖 3，圖中說明包括多核心之處理器耦接至較高階記憶體的實施例。記憶體 310 包括線 315、320、及 325。存取追蹤欄 316、321、及 326 分別對應於線 315、320、及 325。每一個存取欄用來追蹤對於記憶體 310 中其所對應之線的存取。處理器 300 也包括核心 305-308。須注意，記憶體 310 可以是核心 305-308 之任何核心中的低階快取記憶體，或為核心 305-308 所共用的較高階快取記憶體，或任何其它已知或用其它方式在處理器中可被利用做為交易式記憶體的可用記憶體。每一核心包括用以儲存總體溢位表之基礎位址的暫存器，諸如暫存器 330、335、

340、及 345。當使用記憶體 310 執行一交易時，當未配置總體溢位表時，基礎位址 331、336、341、及 346 可能未儲存總體溢位表的基礎位址。

不過，當記憶體 310 溢位時，溢位表 355 被配置。在一實施例中，當溢位表 355 尚未配置時，根據使記憶體 310 溢位的操作而產生中斷或頁錯誤。使用者處置器或核心級(kernel-level)的軟體根據該中斷或頁錯誤將較高階記憶體 350 的範圍配置給溢位表 355。如其它例，總體溢位表係根據被設定的溢位旗標而配置。在此，當溢位旗標被設定時，即嘗試對總體溢位表的寫入。如果寫操作失敗，則在該總體溢位表中配置新頁。

較高階記憶體 350 可以是較高階的快取記憶體、僅與處理器 300 相關的記憶體、為包括處理器 300 之系統所共用的系統記憶體、或位階高於記憶體 310 的任何其它記憶體。配置給溢位表 355 之記憶體 350 中的第一個範圍稱為溢位表 355 的第一頁。以下將參考圖 5 更詳細討論多頁溢位表。

在將空間配置給溢位表 355 之時，或在將記憶體配置給溢位表 355 之後，溢位表 355 的基礎位址被寫入暫存器 330、335、340、及 345。在一實施例中，以核心級的碼將總體溢位表的基礎位址寫入基礎位址暫存器 330、335、340、及 345 其中之一的每一個。或者，以硬體、軟體、或韌體將總體溢位表的基礎位址寫入基礎位址暫存器 330、335、340、及 345 其中之一，且該基礎位址經由核心

305-308 間的訊息傳送協定發佈給其餘的基礎位址暫存器。

如圖示說明，溢位表 355 包括登錄列 360、365、及 370。登錄列 360、365、及 370 包括位址欄 361、366、及 371，以及交易狀態資訊(T.S.I.)欄 362、367、及 372。如溢位表 355 之操作的例示性簡化例，假設來自第一交易的操作具有被存取的線 315、320、及 325，以對應之存取欄 316、321、及 326 的狀態來表示。在第一交易的未決定期間，線 315 被選擇用於逐出。由於存取追蹤欄 316 的狀態代表該線 315 在第一交易期間已被前一存取，且該交易仍未決定，於是發生溢位事件。如上所述，溢位旗標/位元可能被設定。此外，如果未配置有頁或需要另一頁，則將記憶體 350 中的頁被配置給溢位表 355。

如果不需要配置頁，則總體溢位表之目前的基礎位址係由暫存器 330、335、340、或 345 儲存。或者，在初始配置時，溢位表 355 的基礎位址被寫入/發佈給暫存器 330、335、340、或 345。根據溢位事件，登錄列 360 被寫入溢位表 355。登錄列 360 包括位址欄 316，用以儲存與線 315 相關的位址表示。

在一實施例中，與線 315 相關的位址係元件儲存在線 315 中之位置的實體位址。例如，該實體位址係元件在主儲存裝置(諸如系統記憶體)中之儲存位置的實體位址之表示。藉由在溢位表 355 中儲存實體位址，即有可能偵測核心 305-308 之所有存取間的衝突。

反之，當虛擬記憶體位址被儲存到位址欄 316、366、及 367 時，具有不同虛擬記憶體基礎位址及偏移的處理器或核心具有不同之記憶體的邏輯視野。結果是，對於同一實體記憶體位置的存取有可能不會被偵測為一衝突，因為各核心間觀看實體記憶體位置的虛擬記憶體位址有可能不同。不過，如果虛擬記憶體位址是被儲存在溢位表 355 中，結合 OS 控制欄中的上下文識別器，即有可能發現總體衝突。

與線 315 相關之位址表示的另一實施例包括部分或整個虛擬記憶體位址、快取記憶體線位址、或其它實體位址。位址的表示包括有 10 進位、16 進位、2 進位、雜湊值 (hash value)、或位址之所有或任何部分的其它表示/調處 (manipulation)。在一實施例中，標籤值 (其為位址的一部分) 是一位址的表示。

除了位址欄 361 之外，登錄列 360 還包括交易狀態資訊 362。在一實施例中，交易狀態資訊欄 362 用於儲存存取追蹤欄 316 的狀態。例如，如果存取追蹤欄 316 包括交易寫入位元及交易讀取位元等兩個位元分別追蹤對於線 315 的寫入及讀取，則交易寫入位元與交易讀取位元的邏輯狀態被儲存到交易狀態資訊欄 362 內。不過，與交易相關的任何資訊都可儲存在交易狀態資訊 362 內。以下將參考圖 4a-4b 討論溢位表 355 及有可能儲存在溢位表 355 中的其它欄位。

圖 4a 說明總體溢位表的實施例。總體溢位表 400 包

括登錄列 405、410、及 415，其對應於交易執行期間具有被溢位之記憶體的操作。例如，在執行中之交易中的一操作使記憶體溢位。登錄列 405 被寫入總體溢位表 400。登錄列 405 包括實體位址欄 406。在一實施例中，實體位址欄 406 用來儲存與記憶體中之線相關的實體位址，其供正在使該記憶體溢位的操作參考。

如說明例，假設正被執行的第一操作係為交易的一部分，參考具有實體位址 ABCD 的系統記憶體位置。根據該操作，一快取記憶體控制器選擇被該實體位址之一部分 ABC 映射的快取記憶體線，成為用於逐出的快取記憶體線，導致一溢位事件。須注意，ABC 的映射也可包括變換成與位址 ABC 相關的虛擬記憶體位址。由於發生溢位事件，與操作及/或該快取記憶體線相關的登錄列 405 被寫入溢位表 400。在此例中，登錄列 405 的實體位址欄 406 中包括實體位址 ABCD 的表示。由於快取記憶體的組織有很多，諸如直接映射及設定相關的組織，因此，將多個系統記憶體位置映射至單一快取記憶體線或一組快取記憶體線，該快取記憶體線位址有可能參考複數個系統記憶體位置，諸如 ABCA、ABCB、ABCC、ABCE 等，結果是，經由將該實體位址 ABCD 或這些位址的某些表示儲存到實體位址 406 中，即有可能較容易偵測到交易衝突。

除了實體位址欄 406 之外，其它的欄還包括資料欄 407、交易狀態欄 408、及作業系統控制欄 409。資料欄 407 用以儲存元件，諸如指令、運算元、資料、或與使記

記憶體溢位之操作相關的其它邏輯資訊。須注意，每一記憶體線具有儲存多個資料元件、指令、或其它邏輯資訊的能力。在一實施例中，資料欄 407 用以儲存資料元件或要被逐出之記憶體線中的元件。在此，資料欄 407 為選用。例如，在溢位事件時，元件不是儲存在登錄列 405 中，除非該被逐出的記憶體線是在修改狀態，或其它的快取記憶體同調 (coherency) 狀態。除了指令、運算元、資料元件、及其它邏輯資訊之外，資料欄 407 也可包括其它資訊，諸如記憶體線的大小。

交易狀態欄 408 用以儲存與使一交易式記憶體溢位之操作相關的交易狀態資訊。在一實施例中，快取記憶體線的附加位元係存取追蹤欄，用於儲存與該快取記憶體線之存取有關的交易狀態資訊。在此，附加位元的邏輯狀態被儲存在交易狀態欄 408 中。基本上，被逐出的記憶體線被虛擬化，並連同實體位址及交易狀態資訊儲存在較高階的記憶體中。

此外，登錄列 405 包括作業系統控制欄 409。在一實施例中，作業系統控制欄 409 係用於追蹤執行上下文。例如，作業系統控制欄 409 係一 64 位元欄，用以儲存用於追蹤與登錄列 405 相關之執行上下文的上下文 ID 表示。諸如登錄列 410 及 415 等多個登錄列包括類似的欄，諸如實體位址欄 411 及 416、資料欄 412 及 417、交易狀態欄 413 及 418、以及作業系統欄 414 及 419。

接下來請參閱圖 4b，圖中顯示儲存交易狀態資訊之溢

位表的特定說明例。溢位表 400 包括與參考圖 4a 所討論之類似的欄。反之，登錄列 405、410、及 415 包括交易讀取 (Tr) 欄 451、456、及 461，以及交易寫入 (Tw) 欄 452、457、及 462。在一實施例中，Tr 欄 451、456、461 與 Tw 欄 452、457、及 462 分別用於儲存讀取位元及寫入位元的狀態。在一例中，讀取位元與寫入位元分別追蹤對於相關快取記憶體線的讀取與寫入。在寫入登錄列 405 使表 400 溢位時，讀取位元的狀態被儲存在 Tr 欄 451 中，以及寫入位元的狀態被儲存在 Tw 欄 452 中。結果是，藉由在 Tr 及 Tw 欄中指示那些登錄列在交易的未決定期間曾被存取，以將交易的狀態儲存在總體溢位表 400 中。

現回到圖 5，圖中說明多頁溢位表的實施例。在此，儲存在記憶體 500 中的溢位表 505 包括有多頁，諸如頁 510、515、及 520。在一實施例中，處理器中的暫存器儲存第一頁 510 的基礎位址。在寫入到表 505 時，偏移、基礎位址、實體位址、虛擬位址、及這些位址的組合，都參考表 505 內的位置。

在溢位表 505 中，頁 510、515、及 520 可連續，但並非必須連續。事實上，在一實施例中，頁 510、515、及 520 係頁的鏈結表列。在此，次一頁 515 的基礎位址儲存在前一頁 (諸如頁 510) 的登錄列 (諸如登錄列 511) 中。

一開始，溢位表 505 中可能不存在有多頁。例如，當無溢位發生時，可能沒有空間配置給溢位表 505。在另一記憶體溢位時，圖中未顯示，則頁 510 被配置給溢位表

505。頁 510 中的登錄列被寫成在溢位狀態中繼續執行交易。

在一實施例中，當頁 510 被填滿時，頁 510 中沒有更多的空間，嘗試寫入溢位表 505 導致頁錯誤。在此，另一或次一頁 515 被配置。先前對登錄列之寫入的嘗試，經由將該登錄列寫入頁 515 以完成。此外，頁 515 的基礎位址被儲存在頁 510 中的欄 511 中，以使溢位表 505 形成多頁的鏈結表列。同樣地，當頁 520 被配置時，將頁 520 的基礎位址儲存到頁 515 的欄 516 中。

接下來請參考圖 6，圖中說明有能力虛擬化交易式記憶體之系統的實施例。微處理器 600 包括交易式記憶體 610，其為快取記憶體。交易式記憶體 610 的一實施例係在核心 630 中的第一階快取記憶體，類似圖 1 中說明的快取記憶體 103。類似地，交易式記憶體 610 可以是核心 635 中的低階快取記憶體。在另一選擇中，快取記憶體 610 係較高階的快取記憶體，或是處理器 600 中之其它可用的記憶體段。快取記憶體 610 包括線 615、620、及 625。與快取記憶體線 615、620、及 625 相關的附加欄為交易讀取(Tr)欄 616、621、及 626，以及交易寫入(Tw)欄 617、622、及 627。例如，Tr 欄 616 及 Tw 欄 617 對應於快取記憶體線 615，且被用來追蹤對於快取記憶體線 615 的存取。

在一實施例中，Tr 欄 616 及 Tw 欄 617 每一個係快取記憶體線 615 中的單個位元，藉由預設，Tr 欄 616 及 Tw

欄 617 被設定為原設值，諸如邏輯 1。在未決定之交易的執行期間，在從線 615 讀取或載入時，Tr 欄 616 被設定成第二值，諸如邏輯 0，用以表示在未決定之交易的執行期間發生讀取/載入。相應地，如果在未決定的交易期間發生寫入或儲存到線 615，則 Tw 欄 617 被設定成第二值，用以表示在未決定之交易的執行期間發生寫入或儲存。在放棄或確定一交易時，與要被確定或放棄之該交易相關的所有 Tr 欄及 Tw 欄都被重置成原設狀態，以便能夠追蹤對於對應之快取記憶體線的存取。

微處理器 600 也包括用以執行交易的核心 630 及核心 635。核心 630 包括具有溢位旗標 632 及基礎位址 633 的暫存器 631。此外，在 TM 610 係在核心 630 中的實施例中，TM 610 為第一階的快取記憶體或核心 630 中其它可用的儲存區域。同樣地，如前所述，核心 635 包括溢位旗標 637、基礎位址 638、及可能的 TM 610。雖然在圖 6 中說明的暫存器 631 及 635 係分離的暫存器，但也可使用其它的結構來儲存溢位旗標及基礎位址。例如，以微處理器 600 上的單一暫存器來儲存溢位旗標及基礎位址，且核心 630 及 635 總體地可見該暫存器。或者，微處理器 400 或核心 630 及 635 上獨立的暫存器，包括獨立的一或多個溢位暫存器及獨立的一或多個基礎位址暫存器。

初始的交易執行係利用交易式記憶體 610 來執行交易。存取的追蹤、衝突檢查、確認、及其它的交易執行技術，係利用 Tr 及 Tw 欄來實施。不過，在交易式記憶體 610

溢位時，交易式記憶體 610 被延伸進入記憶體 650。如圖示說明，記憶體 650 係系統記憶體，可供處理器 600 專用，或在系統中共用。不過，記憶體 650 也可以是處理器 600 上的記憶體，諸如前所述之第二階的快取記憶體。在此，儲存在記憶體 650 中的溢位表 655 係用來延伸交易式記憶體 610。延伸進入較高階的記憶體也可能意指將交易式記憶體虛擬化或延伸進入虛擬記憶體。基礎位址欄 633 及 638 係用以儲存總體溢位表 655 的基礎位址於系統記憶體 650 中。在一實施例中，溢位表 655 係多頁的溢位表，前一頁(諸如頁 660)將溢位表 655 之次一頁(即頁 665)的次一個基礎位址儲存於欄(即欄 661)中。藉由儲存次一頁的位址於前一頁中，即可建立起記憶體 650 中之頁的鏈結表列，以形成多頁的溢位表 655。

討論以下的例子用以說明系統將交易式記憶體虛擬化之實施例的操作。第一交易從線 615 載入，從線 625 載入，實施計算的操作，並將結果寫回線 620，並接著實施在嘗試確認/確定之前的其它各種操作。在從線 615 載入時，Tr 欄 616 的邏輯值從原設的邏輯狀態 1 被設定為 0，以代表在第一交易的執行期間發生從線 615 的載入，該交易仍為未決定。同樣地，Tr 欄 626 的邏輯值被設定為 0，以代表從線 625 載入。當發生對於線 620 的寫入時，Tw 欄 622 被設定成邏輯 0，以代表在該第一交易的未決定期間發生對於線 620 的寫入。

現在假設第二交易，包括一未得到快取記憶體線 615

的操作，並經由替換演算法，諸如最近使用的演算法，快取記憶體線 615 被選擇用於逐出，而該第一交易仍在未決定中。一快取記憶體控制器或其它邏輯(圖中未說明)偵測導致溢位事件之線 615 的逐出，如 Tr 欄 616 被設定成邏輯值 0，以代表在仍未決定之第一交易的執行期間線 615 被讀取。在另一實施例中，當快取記憶體線 615 因 Tr 欄 616 被設定成邏輯值 0 而被選擇用於逐出時，一中斷被產生。接著，藉由處置器根據該中斷的處置，溢位旗標 632 被設定。核心 630 與 636 間的通信協定被用來設定溢位旗標 637，因此，兩個核心都被通知有溢位事件發生，且交易式記憶體 610 將被虛擬化。

在逐出快取記憶體線 615 之前，交易式記憶體 610 被延伸進入記憶體 650。在此，交易狀態資訊被儲存於溢位表 655 中。一開始，如果未配置溢位表 655，則會產生頁錯誤、中斷、或對核心級程式的其它通信，以請求配置溢位表 655。接著，在記憶體 650 中配置溢位表 655 的頁 660。溢位表 655 的基礎位址，即頁 660，被寫入基礎位址欄 633 與 638。須注意，如上所述，基礎位址可寫入一個核心，諸如核心 635，並透過發訊協定，溢位表 655 的基礎位址可被寫入其它的基礎位址欄 633。

如果溢位表 655 的頁 660 已被配置，一登錄列被寫入頁 660。在一實施例中，該登錄列包括與儲存在線 615 中之該元件相關之實體位址的表示。也可說，該實體位址也與快取記憶體線 615 相關，且該操作使交易式記憶體 610

溢位。該登錄列也包括交易狀態資訊。在此，該登錄列包括 Tr 欄 616 及 Tw 欄 617 的目前狀態，其分別為邏輯 0 及 1。

在該登錄列中另一可能的欄包括用以將運算元、指令、或其它資訊儲存於快取記憶體線 615 中的元件欄，以及用於儲存 OS 控制資訊的操作系統控制欄，諸如上下文識別器。根據快取記憶體線 615 的快取同調狀態，可選擇性地使用元件欄及 / 或元件大小欄。例如，如果快取記憶體線在 MESI 協定中是處於修改狀態，則元件被儲存在該登錄列中。或者，如果該元件是在排除、共用、或無效的狀態中，則元件不儲存在該登錄列中。

假設由於頁 660 已被登錄列填滿，致使登錄列寫入頁 660 中造成頁錯誤，則向諸如作業系統的核心級程式作出請求以產生另一頁。另一頁 665 被配置給溢位表 655。在前一頁 660 的欄 661 中儲存頁 665 的基礎位址，以構成頁的鏈結表列。接著，該登錄列被寫入新加的頁 667。

在另一實施例中，與第一交易相關的其它登錄列（諸如與從線 625 載入及寫入線 620 無關的登錄列），根據溢位而寫入溢位表 655，以虛擬化整個第一交易。不過，並不需要將所有被交易存取的線都複製到溢位表中。事實上，存取追蹤、確認、衝突檢查、及其它的交易執行技術，都可在交易式記憶體 610 及記憶體 650 中實施。

例如，如果第二交易寫入與目前儲存在線 625 中之元件所在的同一實體記憶體位置，由於 Tr 626 表示第一交

易從線 625 載入，因此可偵測到第一與第二交易間的衝突。結果是中斷被產生，且使用者處置器/放棄處置器啓始第一或第二交易的放棄。此外，如果第三交易被寫入該實體位址，其為與線 615 相關之頁 660 中登錄列的一部分。該溢位表被用來偵測該等存取間的衝突，並啓始類似的中斷/放棄處置器常式。

如果在第一交易的執行期間未偵測到無效的存取/衝突，或確認成功，則第一交易被確定。溢位表 655 中與第一交易相關的所有登錄列都被釋放。在此，釋放一登錄列包括從溢位表 655 刪除登錄列。或者，釋放一登錄列包括重置該登錄列中的 Tr 欄及 Tw 欄。當溢位表 655 中的最後一個登錄列被釋放時，溢位旗標 632 與 637 被重置到原設狀態，指示交易式記憶體 610 目前未被溢位。溢位表 655 可選擇性地去配置，以便有效率地使用記憶體 650。

現回到圖 7，圖中說明用以虛擬化交易式記憶體之方法的流程圖的實施例。在流程 705 中，與執行做為交易中之一部分之操作相關的溢位事件被偵測到。該操作參考交易式記憶體中的記憶體線。在一實施例中，該記憶體係為實體處理器上之多核心中之一核心中的低階資料快取記憶體。在此，第一核心包括該交易式記憶體，而其它核心則藉由監聽/請求儲存在該低階快取記憶體中的元件以共同存取該記憶體。或者，該交易式記憶體係為第二階或較高階的快取記憶體，在複數個核心間直接共用。

一位址參考一記憶體線包括經由轉換、調處、或其它

計算以參考與該記憶體線相關的位址而參考到一位址。例如，當被轉換時，該操作參考一參考系統記憶體中之實體位置的虛擬記憶體位址。通常快取記憶體被一位址的一部分或標籤值編索引。因此，索引快取記憶體之共用線之位址的標籤值被虛擬記憶體位址參考，亦即被轉換及/或調處成爲標籤值。

在一實施例中，如果記憶體中的線被未決定的交易前一存取，則溢位事件包括在被該操作參考的記憶體中，逐出或選擇用於逐出的線。或者，對於溢位或造成溢位之事件的任何預測，也都可考慮成溢位事件。

在流程 710 中，當該記憶體被溢位時，則根據該溢位事件設定溢位位元/旗標。在暫存器中的單一個溢位位元可被所有核心或處理器總體地看見，以確保每一個核心都知道該記憶體已溢位，且已被虛擬化。或者，每一核心或處理器包括有溢位位元，其是經由發訊協定設定，以通知溢位及虛擬化的每一個處理器。

如果該溢位位元被設定，則該記憶體被虛擬化。在一實施例中，虛擬化一記憶體包括儲存與該記憶體線相關的交易狀態資訊於總體溢位表中。基本上，涉及記憶體溢位之記憶體之線的表示被虛擬化、延伸、及/或部分地複製到較高階的記憶體中。在一實施例中，存取追蹤欄的狀態及與被操作參考之記憶體之線相關的實體位址，被儲存在較高階記憶體中的總體溢位表中。較高階記憶體中的登錄列被以相同的方法利用，如記憶體被追蹤存取、偵測衝突

、實施交易確認等。

現請參考圖 8，圖中顯示用以系統虛擬化交易式記憶體之流程圖的說明實施例。在流程 805 中，交易被執行。交易包括分類複數個操作或指令。如前所述，交易在軟體中被硬體或該兩者的組合區劃。該等操作通常是參考一虛擬記憶體位址，當其被轉換時，參考系統記憶體中的直線及/或實體位址。在交易的執行期間，在處理器或核心間被共用的交易式記憶體(諸如快取記憶體)被用來追蹤存取、偵測衝突、實施確認等。在一實施例中，每一個快取記憶體線對應於一存取欄，其被用來實施上述的操作。

在流程 810 中，在快取記憶體中選擇要被逐出的快取記憶體線。在此，另一交易或操作嘗試存取一記憶體位置，導致選擇要被逐出的快取記憶體線。任何習知或其它可用的快取記憶體替換演算法都可被快取記憶體控制器或其它邏輯用來選擇用於逐出的線。

如果決定流程 815，則接著決定該被選擇的快取記憶體線在交易的未決定期間是否被前一存取。在此，該存取追蹤欄被檢查，以決定是否發生對於該被選擇之快取記憶體線的存取。如果無存取被追蹤，則該快取記憶體在流程 820 被逐出。如果該逐出是交易內之操作的結果，則該逐出/存取可能被追蹤。不過，如果在未決定之交易的執行期間一存取被追蹤，則在流程 825 決定總體溢位位元目前是否被設定。

在流程 830 中，如果總體溢位位元目前未被設定，則

設定該總體溢位位元，因為逐出在未決定之交易的執行期間被存取的快取記憶體線而發生該快取記憶體的溢位。須注意，在另一實施中，流程 825 可在流程 815、820、及 830 之前實施，且如果指示快取記憶體已被溢位的總體溢位位元目前已被設定，則可跳過流程 815、820、及 830。基本上，在該另一實施中，當該溢位位元已表示該快取記憶體被溢位，則不需要偵測溢位事件。

現回應到說明的流程圖，不過，如果該總體溢位位元被設定，則在流程 835 決定總體溢位表的第一頁是否被配置。在一實施例中，決定總體溢位表之第一頁是否被配置包括與核心級程式通信，以決定該頁是否被配置。如果總體溢位表未被配置，則在流程 840 中配置第一頁。在此，請求作業系統配置記憶體頁導致總體溢位表的配置。在另一實施例中，流程 855-870 被用來決定第一頁是否被配置並配置該第一頁，以下將更詳細討論。本實施例包括嘗試使用基礎位址對總體溢位表的寫入，如果該總體溢位表未被配置，則該寫入會造成頁錯誤，並接著根據該頁錯誤配置該頁。另一方法是在配置該溢位表的初始頁時，該溢位表的基礎位址被寫入執行該交易之處理器/核心的暫存器中。結果是，後續的寫操作可參考一偏移，或其它參考用於登錄列之正確實體記憶體位置的位址，該位址與基礎位址結合寫入該暫存器。

在流程 850 中，與登錄列相關的快取記憶體線被寫入該總體溢位表中。如前所述，該總體溢位表可能包括以下

欄位的組合：位址；元件；快取記憶體線的大小；交易狀態資訊；及操作系統控制欄。

在流程 855 中，其決定在寫操作時是否發生頁錯誤。如前所述，頁錯誤可能是無溢位表之初始配置或溢位表目前已滿的結果。如果該寫操作成功，則回到流程 805 繼續正規的執行、確認、存取追蹤、確定、放棄等。不過，如果產生頁錯誤指示該溢位表中需要更多空間，則在流程 860 中為該總體溢位表配置另一頁。在流程 870 中，該另一頁的基礎位址被寫入前一頁。此形成鏈結表列式的多頁表。接著，經由將該登錄列寫入新配置的另一頁以完成該意欲的寫操作。

如以上說明，較小較不複雜的交易可獲得到使用局部交易式記憶體在硬體中執行交易的優點。此外，隨著要被執行之交易之數量及這些交易的複雜度增加，該交易式記憶體被虛擬化，以在局布共用的交易式記憶體溢位時支援持續的執行。使用總體溢位表完成交易的執行、衝突檢查、確認、及確定，直至該交易式記憶體不再被溢位為止，以取代放棄交易及浪費執行時間。總體溢位表有可能儲存實體位址，以確保可偵測到具有不同虛擬記憶體之視野之上下文間的衝突。

上述的方法、軟體、韌體或碼可經由儲存在可由處理元件執行之機器可存取或機器可讀取媒體上的指令或碼來實施。機器可存取/可讀取媒體包括任何機制，其提供(即儲存及/或傳送)可被機器讀取之型式的資訊，諸如可被電

腦或電子系統讀取。例如，機器可存取媒體包括隨機存取記憶體 (RAM)、諸如靜態 RAM (SRAM) 或動態樣 RAM (DRAM)；ROM；磁性或光學儲存媒體；快閃記憶體裝置；電、光、聲或其它型式的傳播信號(例如載波、紅外線信號、數位信號)等。

在以上的說明書中，已參考特定的例示性實施例詳細描述。不過，很明顯，其可做各種的修改及改變，不會偏離所附申請專利範圍中所提出之發明之較廣義的精神與範圍。因此，本說明書及圖示可視為意在說明而非意在限制。此外，實施例的前述使用及其它例示性的語言並不必然為相同的實施例或相同的例子，而可視為不同且有區別實施例，以及潛在上相同的實施例。

【圖式簡單說明】

圖 1 說明的多核心處理器實施例具有延伸交易式記憶體的能力。

圖 2a 說明的多核心處理器實施例包括有用於每一核心的暫存器，用以儲存溢位旗標。

圖 2b 說明的多核心處理器實施例包括有總體暫存器，用以儲存溢位旗標。

圖 3 說明的多核心處理器實施例包括有用於每一核心的基礎位址暫存器，用以儲存溢位表的基礎位址。

圖 4a 說明溢位表的實施例。

圖 4b 說明溢位表的另一實施例。

圖 5 說明包括有複數個頁之溢位表的另一實施例。

圖 6 說明用來虛擬化交易式記憶體之系統的實施例。

圖 7 說明虛擬化交易式記憶體之流程圖的實施例。

圖 8 說明虛擬化交易式記憶體之流程圖的另一實施例。

【主要元件符號說明】

- 100：多核心處理器
- 101,102：核心
- 110,115：執行單元
- 120,121：排程器
- 160,165,170,175：執行緒
- 140,141：提取解碼方塊
- 150：匯流排介面單元
- 145：較高階的快取記憶體
- 135：微碼 ROM
- 130,131：配置器更名器方塊
- 125,126：重排序/止用單元
- 103,108：較低階的快取記憶體
- 104,105,106：快取記憶體線
- 104a,105a,106a：記憶體線
- 104b,105b,106b：存取追蹤欄
- 107,109：溢位模組
- 136：微碼唯讀記憶體

- 200 : 多核心處理器
- 205-208 : 核心
- 210 : 記憶體
- 230,235,240,245 : 暫存器
- 231,236,241,246 : 旗標
- 250 : 溢位暫存器
- 251 : 溢位旗標
- 310 : 記憶體
- 315,320,325 : 記憶體線
- 316,321,326 : 存取追蹤欄
- 305-308 : 核心
- 330,335,340,345 : 基礎位址暫存器
- 331,336,341,346 : 基礎位址
- 355 : 溢位表
- 350 : 較高階記憶體
- 360,365,370 : 登錄列
- 361,366,371 : 位址欄
- 362,367,372 : 交易狀態資訊欄
- 400 : 總體溢位表
- 405,410,415 : 登錄列
- 406 : 實體位址欄
- 407 : 資料欄
- 408 : 交易狀態欄
- 409 : 作業系統控制欄

411,416 : 實體位址欄
412,417 : 資料欄
413,418 : 交易狀態欄
414,419 : 作業系統欄
451,456,461 : 交易讀取(Tr)欄
452,457,462 : 交易寫入(Tw)欄
500 : 記憶體
505 : 溢位表
510,515,520 : 頁
600 : 微處理器
610 : 交易式記憶體
630 : 核心
635 : 核心
615,620,625 : 記憶體線
615,620,625 : 快取記憶體線
616,621,626 : 交易讀取欄
617,622,627 : 交易寫入欄
632 : 溢位旗標
633 : 基礎位址
631 : 暫存器
637 : 溢位旗標
638 : 基礎位址
650 : 記憶體
655 : 溢位表

661 : 欄

665 : 頁

660 : 頁

101年8月23日 修正
補充

附件 3A：第 096123333 號申請專利範圍修正本

民國101年 8 月 23 日修正

十、申請專利範圍

1. 一種用於虛擬化交易式記憶體系統的總體溢位之裝置，包含：一處理器，包括

一執行模組，適用以執行包含交易式記憶體存取操作之交易；

一快取記憶體，耦接至該執行模組，該快取記憶體包括複數個記憶體線，其中該複數個記憶體線之一記憶體線與一予以適用以保持現行交易狀態資訊的該快取記憶體內的對應追蹤欄相關，該交易狀態資訊用以表示是否該記憶體線回應於在該交易的未決定期間執行之該交易式記憶體存取操作已經為該交易所存取；以及

溢位邏輯，回應於與該記憶體線相關的溢位事件，在該交易未決定期間適用以支援該快取記憶體延伸入到予以被保持在第二記憶體的總體溢位表，其中該延伸入到該總體溢位表包含啓始該總體溢位表的更新，該更新包括：實體位址、來自對應追蹤欄的現行交易狀態資訊、及來自記憶體線的資料。

2. 如申請專利範圍第 1 項的裝置，其中該處理器更包含邏輯，以保持複數架構狀態，該複數架構狀態的第一架構狀態具有該第二記憶體的第一虛擬視圖與該交易相關；以及，該複數架構狀態的第二架構狀態具有該第二記憶體的第二虛擬視圖不與該交易相關，及其中該處理器也包含

衝突偵測邏輯，用以根據保持在該總體溢位表中之該實體位址及該現行交易狀態資訊，偵測相關於該第二架構狀態操作與該交易的衝突。

3.如申請專利範圍第 1 項的裝置，其中該第二記憶體包括共享系統記憶體，及其中該溢位邏輯包含：

一溢位儲存元件，用以回應於該溢位事件而保持一溢位值；

一基礎位址儲存元件，用以保持該總體溢位表之基礎位址的表示值予以保持於該共享系統記憶體中，其中該總體溢位表係包括用來保持該交易狀態資訊與該實體位址的總體溢位登錄列，該總體溢位登錄列的登錄列實體位址係與為該轉譯邏輯所自該虛擬記憶體位址轉譯的該實體位址不同。

4.如申請專利範圍第 3 項的裝置，其中在該交易之未決定期間用以追蹤存取該記憶體線之該對應的追蹤欄包含：

第一位元，用以在該交易之未決定期間追蹤來自該記憶體線的載入；

第二位元，用以在該交易之未決定期間追蹤對該記憶體線的儲存。

5.如申請專利範圍第 4 項的裝置，其中該總體溢位登錄列包含：

元件欄，用以保持與該記憶體線相關的元件；

位址欄，用以保持該實體位址；

一 交易讀取狀態欄，用以保持該對應追蹤欄之該第一位元的狀態；以及

一 交易寫入狀態欄，用以保持該對應追蹤欄之該第二位元的狀態。

6.如申請專利範圍第 5 項的裝置，其中該共享系統記憶體係為該處理器的複數個核心間所共享，各個核心具有其實體記憶體的本身虛擬視圖，且其中該複數個核心中的每一個核心回應於保持該溢位值的該溢位儲存元件，利用該實體位址，檢查於驗證期間該總體溢位表之衝突。

7.如申請專利範圍第 4 項的裝置，其中當在該交易的未決定期間該第一位元追蹤來自該記憶體線的前一載入，或在該交易的未決定期間該第二位元追蹤對該記憶體線的前一儲存時，溢位事件包括選擇用於逐出的該記憶體線，該溢位邏輯更相關於與該現行交易狀態資訊相關的實體位址，將現行資訊由該快取記憶體線寫回到該總體溢位表；及快取控制邏輯，以新資訊替換該記憶體線及在該溢位邏輯啓始對該總體溢位表的更新後，重設對應追蹤欄，以保持相關於現行交易狀態資訊的該實體位址。

8.如申請專利範圍第 1 項的裝置，其中該記憶體線係為保持在該快取記憶體中之虛擬記憶體位址所參考，該虛擬記憶體位址當為在該處理器中之轉譯邏輯轉譯時，參考該實體位址，及其中溢位事件包括為被巢套(nest)在該交易中的第二交易執行開始交易的指令。

9.一種用於虛擬化交易式記憶體系統的總體溢位之裝

置，包含：

一執行單元，適用以執行被分類為一交易的複數個操作；

架構邏輯，適用以保持用於複數軟體緒的複數架構狀態，其中該複數軟體緒的一軟體緒係包含該交易；

一交易記憶體，耦接至該執行模組，該記憶體包括複數個線；以及

一暫存器，耦接至該執行單元，以包含一溢位欄；

溢位硬體，適用以當為該執行單元所執行，回應於該被分類為該交易的複數操作之一操作，而更新該溢位欄至一溢位值，以使得在該交易被執行時所先前存取之複數線的一線被選擇逐出並在以用於多數操作的該操作的新資訊更新該線之前，將該線寫回至交易總體溢位表；及

衝突偵測邏輯，適用以利用至少該交易總體溢位表，回應於保持該溢位值的該暫存器，執行予以包括在第二軟體緒中之第二交易的驗證。

10.如申請專利範圍第 9 項的裝置，其中該架構邏輯包含複數核心，各個核心保持至少一軟體緒的架構狀態，及其中該交易溢位欄係為微處理器之複數個處理核心可見。

11.如申請專利範圍第 9 項的裝置，其中該架構邏輯在單一處理器核心內包含複數硬體緒，各個硬體緒保持用於多數軟體緒之一軟體緒的一架構狀態，及其中該單一處理器核心包含該儲存元件，及該溢位欄係為該複數硬體緒

的各個硬體緒可見。

12.如申請專利範圍第 10 項的裝置，其中各個該複數核心回應於該溢位欄保持一非溢位值，而不是保持溢位值，只利用該交易式記憶體執行驗證。

13.如申請專利範圍第 12 項的裝置，其中回應於在該總體溢位表中之該最後的登錄列被釋放，該溢位欄被清除為非溢位值。

14.如申請專利範圍第 9 項的裝置，其中該儲存元件係特定機器暫存器(machine specific register (MSR))。

15.一種用於虛擬化交易式記憶體系統的總體溢位之裝置，包含：

一處理器，包括

一執行單元，用以執行一交易；

一快取記憶體，耦接至該執行單元；及

一基礎位址暫存器，保持用於總體溢位表的基礎位址的表示值於較該快取記憶體為高階的記憶體中，該總體溢位表保持有關於回應於該交易未決定期間該快取記憶體被溢位，在該交易執行時存取的複數快取記憶體位置的交易狀態資訊，其中該交易狀態資訊包含：有關於該快取記憶體線的第一位元的狀態及第二位元的狀態，在該交易執行時，該第一位元追蹤來自該快取記憶體線的讀取及該第二位元追蹤寫入至該快取記憶體線，

其中該總體溢位表被用來保持於該交易執行時與該快取記憶體被溢位之一快取記憶體線相關的登錄列，其中該

登錄列被用來包括與該快取記憶體線相關的實體位址及交易狀態資訊，

其中如果該快取記憶體線在被修改的狀態，則該登錄列被進一步用來包括：與該快取記憶體線相關之資料元件的複製。

16.如申請專利範圍第 15 項的裝置，其中該登錄列被進一步用來包括：作業系統(OS)控制欄。

17.如申請專利範圍第 15 項的裝置，其中該溢位表也用來保持在該總體溢位表中次頁的實體位址。

18.一種用於虛擬化交易式記憶體系統的總體溢位之裝置，包含：

一執行模組，用以執行一交易；

一記憶體，耦接至該執行模組，該記憶體包括複數個區塊，其中一存取追蹤欄係用以追蹤在該交易的執行期間對該複數區塊之一區塊的存取；

一第一儲存元件，包括一溢位欄，回應於該區塊被選擇逐出及該存取追蹤欄在該交易的執行期間表示對該區塊之前一存取，該溢位欄在對該區塊的目前存取時被設定至一溢位值；以及

第二儲存元件，回應於該溢位旗標被設定，保持總體溢位表的基礎位址；以及

溢位邏輯，利用保持在該第二儲存元件中之該基礎位址，將保持在該存取追蹤欄中之前一存取追蹤資訊及相關於該區塊的位址寫至在該總體溢位表中之登錄列中，

在該交易的執行期間，回應於從該區塊的載入設定該存取追蹤欄的第一位元的邏輯；

在該交易的執行期間，回應於對該區塊的儲存設定該存取追蹤欄的第二位元的邏輯；以及

在該交易的執行期間，如果該第一位元被設定，在確定(committing)該交易之時，清除該第一及第二位元的邏輯。

19.如申請專利範圍第 18 項的裝置，其中，回應於該總體溢位位元被設定，該總體溢位表被用來保持與該區塊相關的登錄列，其中該登錄列包含：

與該區塊相關的實體位址；

回應於該區塊被保持在第一同調狀態，與該區塊相關的資料元件；以及

該第一位元的邏輯值；

該第二位元的邏輯值；以及

作業系統(OS)控制欄。

20.如申請專利範圍第 19 項的裝置，其中該記憶體係快取記憶體，且其中該第一同調狀態係被修改的狀態。

21.如申請專利範圍第 18 項的裝置，其中該第一及第二儲存元件為特定機器暫存器(MSR)。

22.如申請專利範圍第 18 項的裝置，其中該第一儲存元件係溢位暫存器，及該第二儲存元件係基礎位址暫存器。

23.如申請專利範圍第 18 項的裝置，其中該溢位欄包

含一溢位位元，該記憶體係快取記憶體，且該總體溢位表的該基礎位址係在記憶體層級之較該快取記憶體為高位階之記憶體中的實體基礎位址。

24. 一種用於虛擬交易式記憶體系統的總體溢位之系統，包含：

微處理器，包括：

執行單元，用以執行包括交易式記憶體存取操作之交易；

第一記憶體，耦接至該執行單元，該第一記憶體包括與一追蹤欄相關的第一記憶體線，該追蹤欄予以以交易狀態資訊更新，用以回應於該交易式記憶體存取操作存取該第一記憶體線，在該交易的未決定期間，表示該第一記憶體線已經被存取；

溢位邏輯，用以當該追蹤欄被更新以保持表示該第一記憶體線已經在該交易的未決定期間被存取的該交易狀態資訊時，回應於選擇該第一記憶體線用以逐出以替換，以檢測該第一記憶體的溢位；以及，將至少用於該第一記憶體線的位址及該交易狀態資訊寫入被保持於第二記憶體中之總體溢位表的登錄列中，而不必指定新位址，來保持來自該第一記憶體線的資料；以及

其中該第二記憶體，在記憶體層級中之階層高於該第一記憶體。

25. 如申請專利範圍第 24 項的系統，其中該第一記憶體的延伸入該溢位表中包含儲存有關於該交易的交易狀態

資訊於該溢位表中。

26.如申請專利範圍第 25 項的系統，其中該溢位邏輯包含：

— 第一暫存器，用以儲存回應該交易的執行期間所發生之溢位事件而被設定的溢位位元；

— 第二暫存器，用以儲存該溢位表的實體基礎位址於該第二記憶體中。

27.如申請專利範圍第 26 項的系統，其中保持在該第二記憶體中的溢位表包括複數頁，其中該複數頁中的每一頁係用以保持於該溢位表之次頁的下一個實體基礎位址。

28.如申請專利範圍第 26 項的系統，其中該第一記憶體係資料快取記憶體，且該第二記憶體係系統記憶體，且其中溢位事件包括選擇在該資料快取記憶體線中已經在該交易執行期間先前被存取的要逐出的一快取記憶體線。

29.如申請專利範圍第 28 項的系統，其中選擇要逐出的快取記憶體線係由一快取記憶體控制器來實施，且其中回應於選擇已經在該交易的執行期間先前被存取的要逐出的一快取記憶體線來設定該溢位位元包含：

回應於選擇要逐出的該快取記憶體線，則產生一中斷，以及

以被引動以處理該中斷的處理程式(handler)來設定該溢位位元。

30.一種用於虛擬化交易式記憶體系統的總體溢位之方法，包含：

偵測與即將執行為第一軟體緒內的交易之一部分之操作相關的溢位事件，該操作參考交易式記憶體中的記憶體線；

如果該溢位位元目前並未設定，則回應該溢位事件設定一溢位位元；

回應於該溢位位元的被設定，延伸該交易式記憶體進入被保持在第二記憶體內的總體溢位表，而不必在該總體溢位表內，指定一新實體位址來保持來自該記憶體線的資料；

回應於該溢位位元被設定，利用該總體溢位表，執行在該第二軟體緒內的第二交易的驗證；以及

回應於該溢位位元的並未設定，只利用該交易記憶體執行該第二交易的驗證。

31.如申請專利範圍第 30 項的方法，其中回應於該溢位位元的被設定，延伸該交易式記憶體進入第二記憶體包含：回應於該溢位位元的被設定，儲存該交易的狀態在一總體溢位表中。

32.如申請專利範圍第 30 項的方法，其中，偵測與即將執行為交易之一部分之操作相關的溢位事件包含：

選擇要逐出的該記憶體線；

從與該記憶體線相關的存取追蹤欄決定是否該記憶體線在該交易的執行期間先被存取，以及

如果決定該記憶體線在該交易的執行期間已先被存取，則偵測出一溢位事件。

33.如申請專利範圍第 30 項的方法，其中該溢位位元被儲存在為複數個核心可見的特定機器暫存器(MSR)中。

34.如申請專利範圍第 31 項的方法，其中儲存該交易之該狀態在該總體溢位表中包含：

將一登錄列寫入該總體溢位表，其中該登錄列包括：

與該記憶體線相關的實體位址；

用於追蹤在該交易的執行期間，從該記憶體線載入之第一追蹤欄的狀態；

用於追蹤在該交易的執行期間，儲存來自該記憶體線之第二追蹤欄的狀態；以及

如果該記憶體線在被修改的狀態，則一資料元件與該實體位址相關。

35.一種用於虛擬化交易式記憶體系統的總體溢位之方法，包含：

執行被分類為交易之複數個操作中的一操作；

根據該操作，在快取記憶體中選擇一要被逐出的快取記憶體線；以及

如果該被選擇的快取記憶體線在該交易的未決定期間先被存取，則：

如果該總體溢位目前未被設定，則設定總體溢位位元；

如果該總體溢位表的第一頁目前未配置，則配置記憶體的第一頁在用於總體溢位表之第二記憶體中，其中該總體溢位表係用來儲存要被逐出之該快取記

憶體線及包含有關於該予以逐出之該快取記憶體線的狀態資訊的與該交易相關的狀態資訊而不必新實體位址用於該快取記憶體線；以及

在配置該第一頁用於該總體溢位表時，將該系統記憶體中該第一頁的基礎位址寫入一基礎位址暫存器。

36.如申請專利範圍第 35 項的方法，另包含：

如果該被選擇的快取記憶體線在該交易的未決定期間已先被存取，則產生一中斷；以及

以處理程式處理該中斷，其中該總體溢位位元係根據該中斷的該處理被設定。

37.如申請專利範圍第 36 項的方法，其中與該交易相關的狀態資訊包括在該交易的未決定期間，用以追蹤對該快取記憶體線之存取之存取追蹤欄的狀態。

38.如申請專利範圍第 37 項的方法，其中該總體溢位表也用以儲存：

與該快取記憶體線相關的實體位址；以及

作業系統(OS)控制欄資訊。

39.如申請專利範圍第 38 項的方法，其中該 OS 係根據該中斷，配置記憶體的該第一頁在該第二記憶體中。

40.如申請專利範圍第 35 項的方法，另包含：

如果發生溢位頁錯誤及至少該第一頁目前配置給該總體溢位表，則配置另一頁在該第二記憶體中，用於該總體溢位表，以及

將該第二記憶體中之該另一頁的另一基礎位址寫入該第二記憶體中的前一頁，該前一頁在邏輯上係在該總體溢位表中之該另一頁之前。

41. 一種有形機器可讀取媒體儲存程式碼，其當為一機器所執行時，使得該機器執行如申請專利範圍第 34 項所述之方法。

42. 一種有形機器可讀取媒體儲存程式碼，其當為一機器所執行時，使得該機器執行如申請專利範圍第 37 項所述之方法。

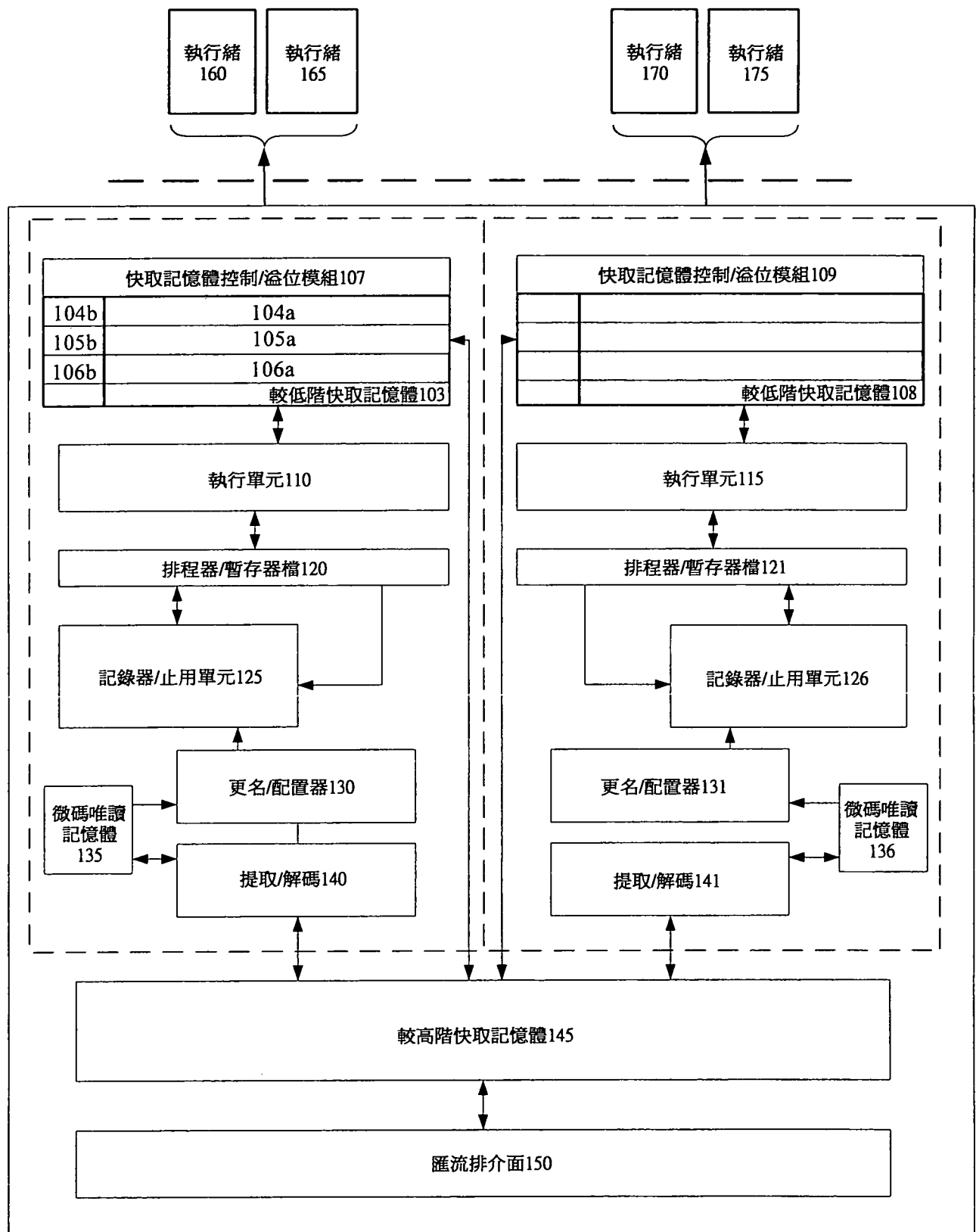


圖 1

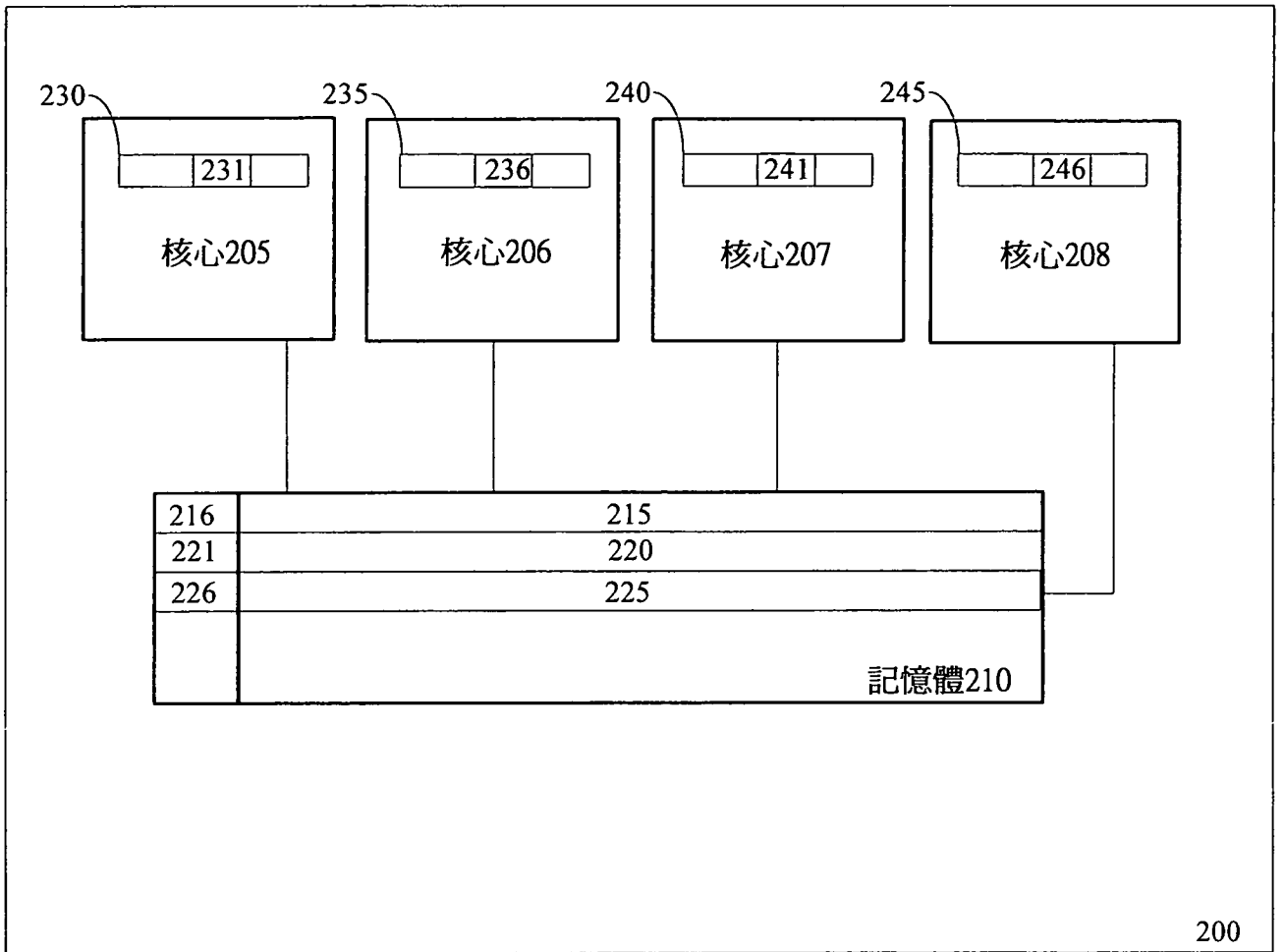


圖2a

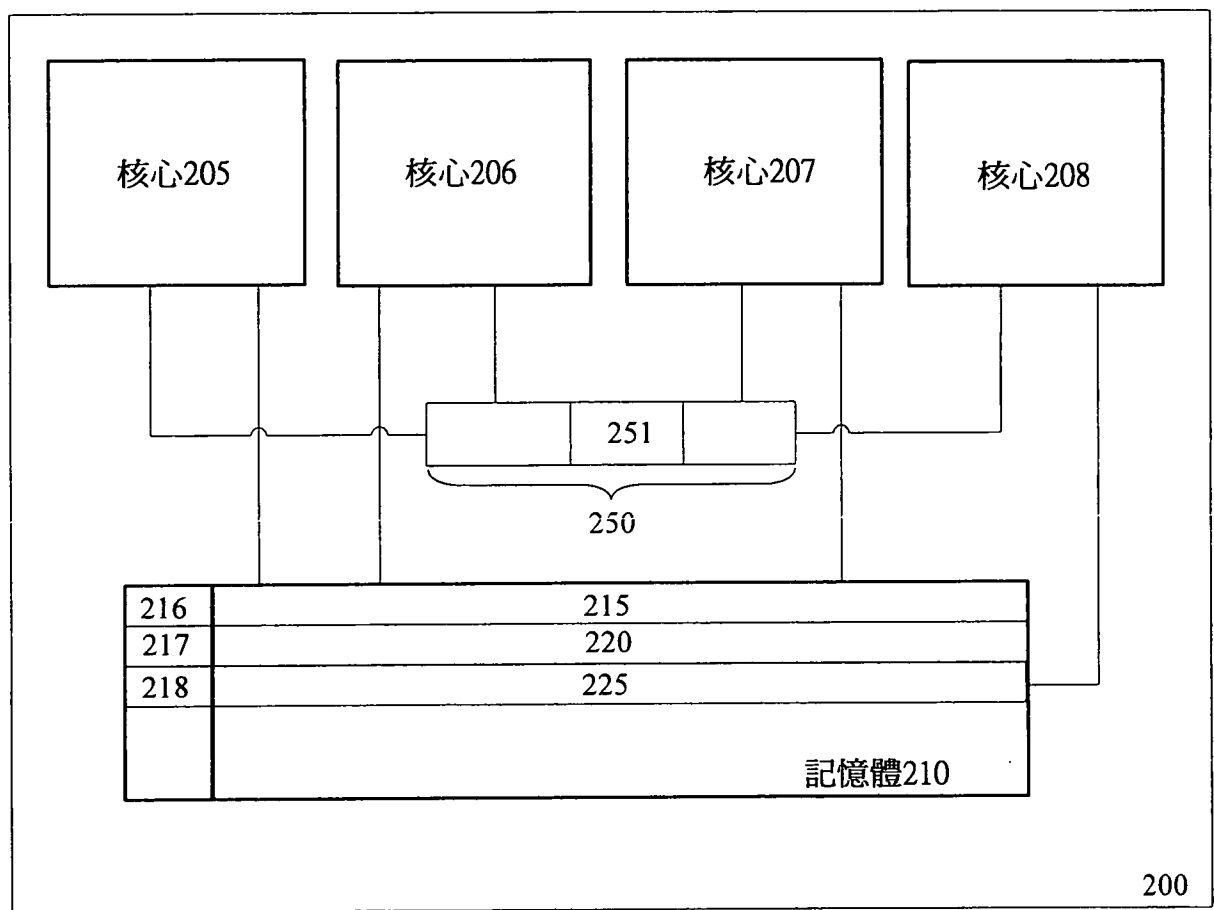


圖 2b

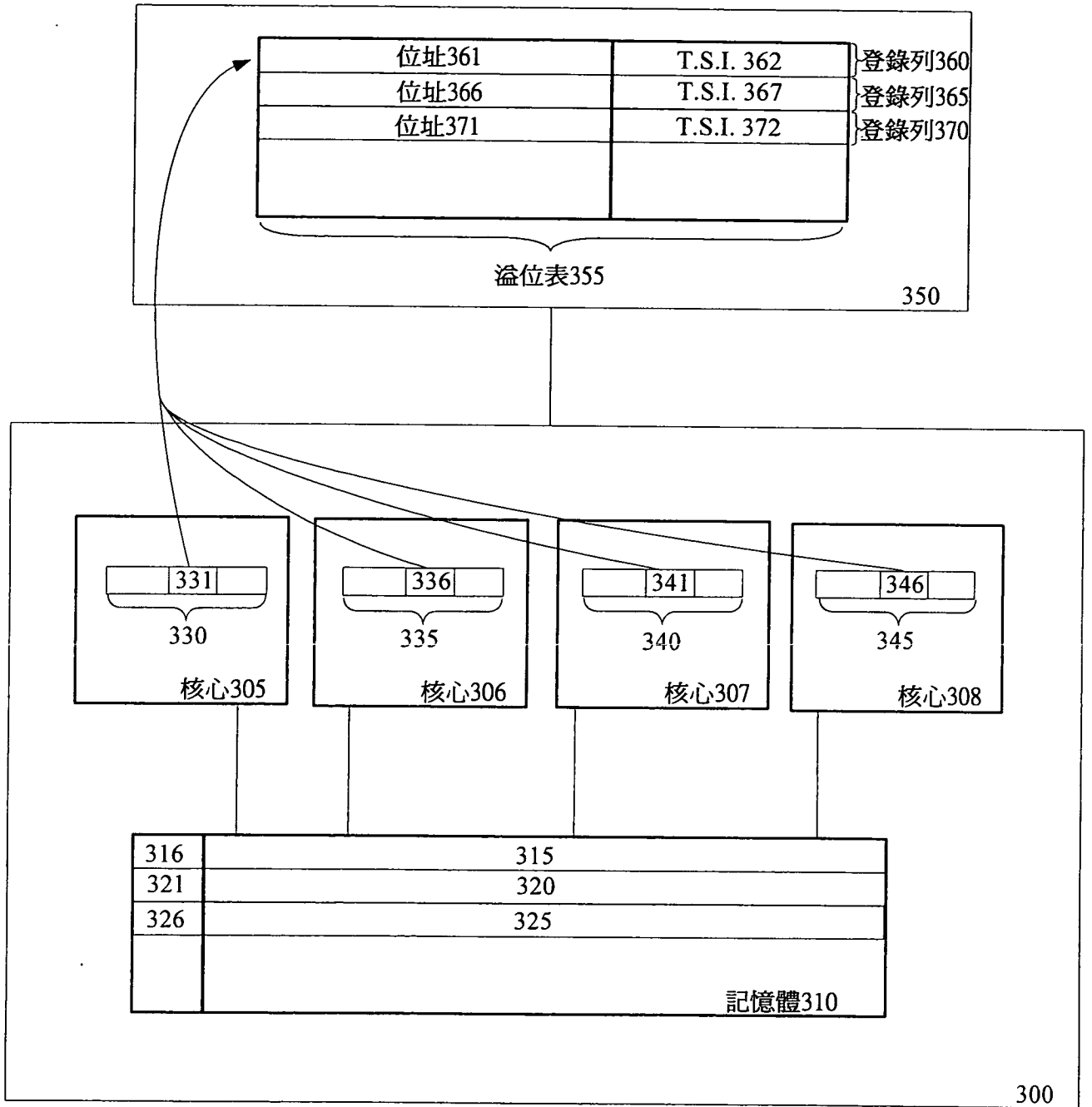


圖3

實體位址406	資料407	T.S. 408	OS欄409	登錄列405
實體位址411	資料412	T.S 413	OS欄414	登錄列410
實體位址416	資料417	T.S 418	OS欄419	登錄列415

溢位表 400

圖 4a

		Tr	Tw		
實體位址406	資料407	451	452	OS欄409	登錄列405
實體位址411	資料412	456	457	OS欄414	登錄列410
實體位址416	資料417	461	462	OS欄419	登錄列415

溢位表 400

圖 4b

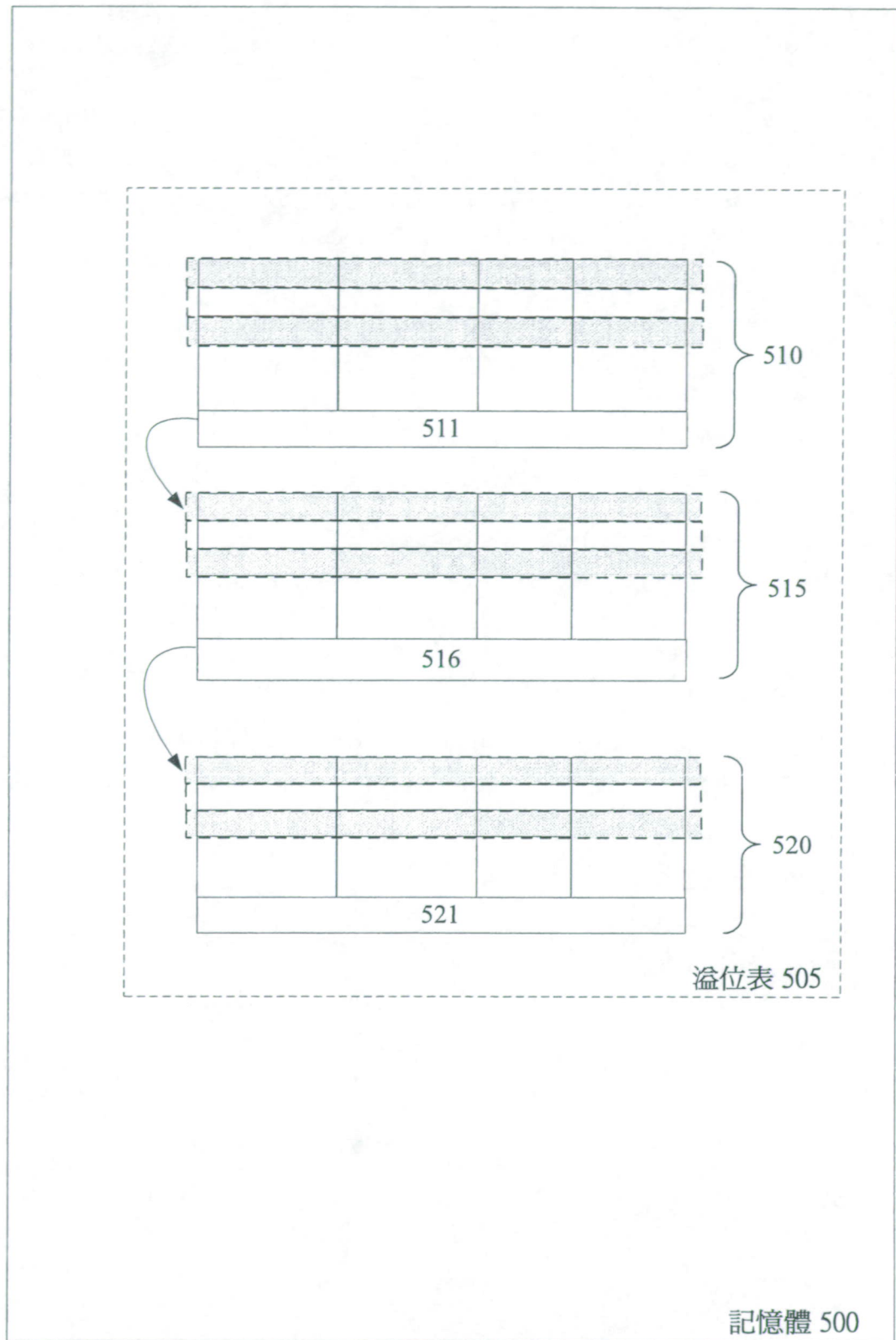


圖5

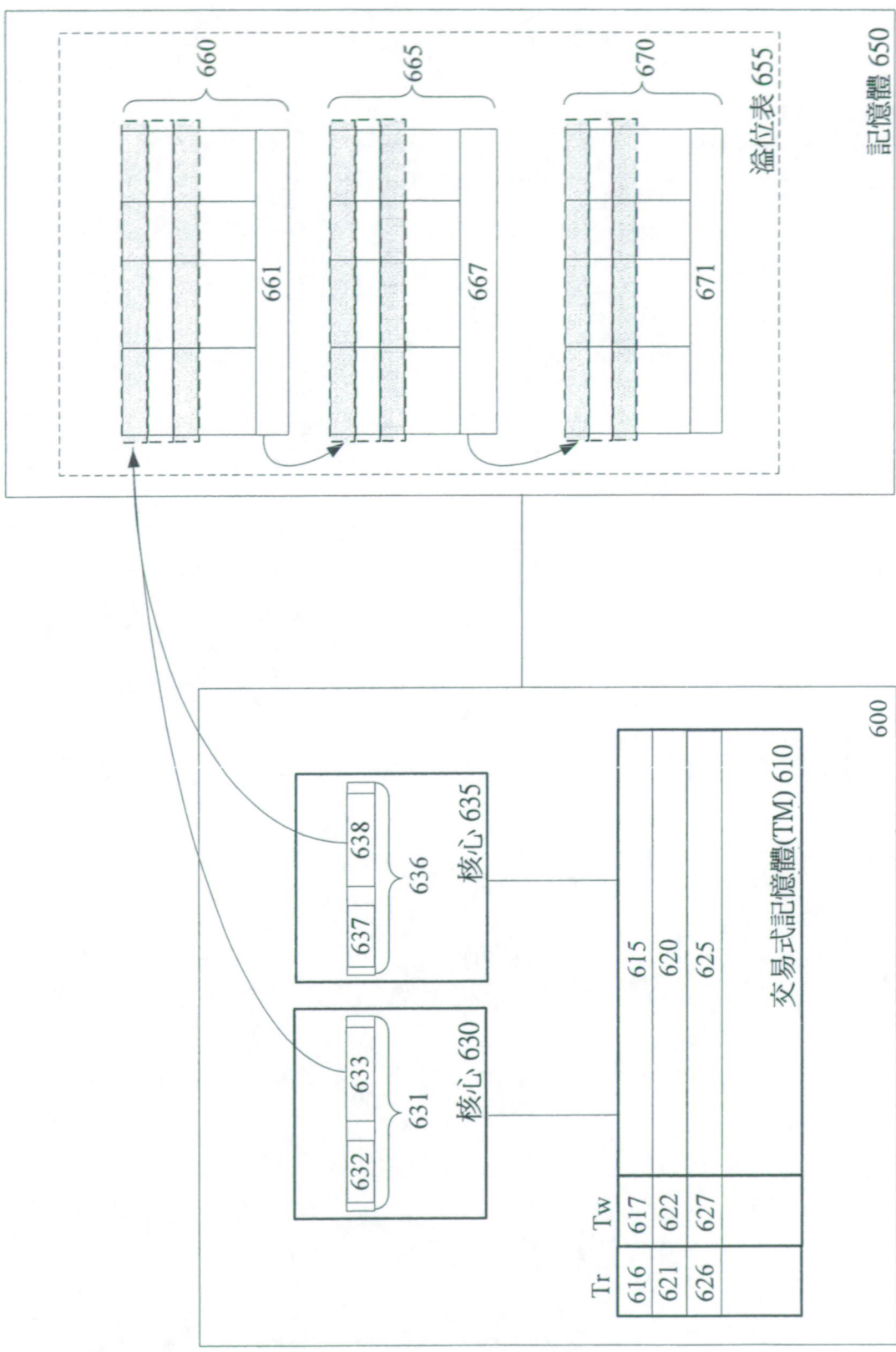


圖6

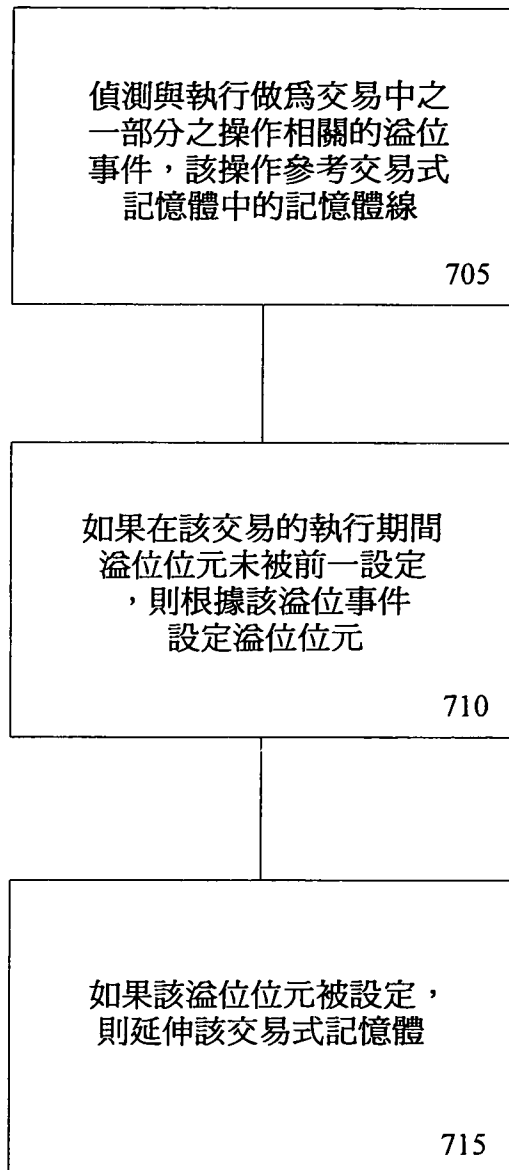


圖 7

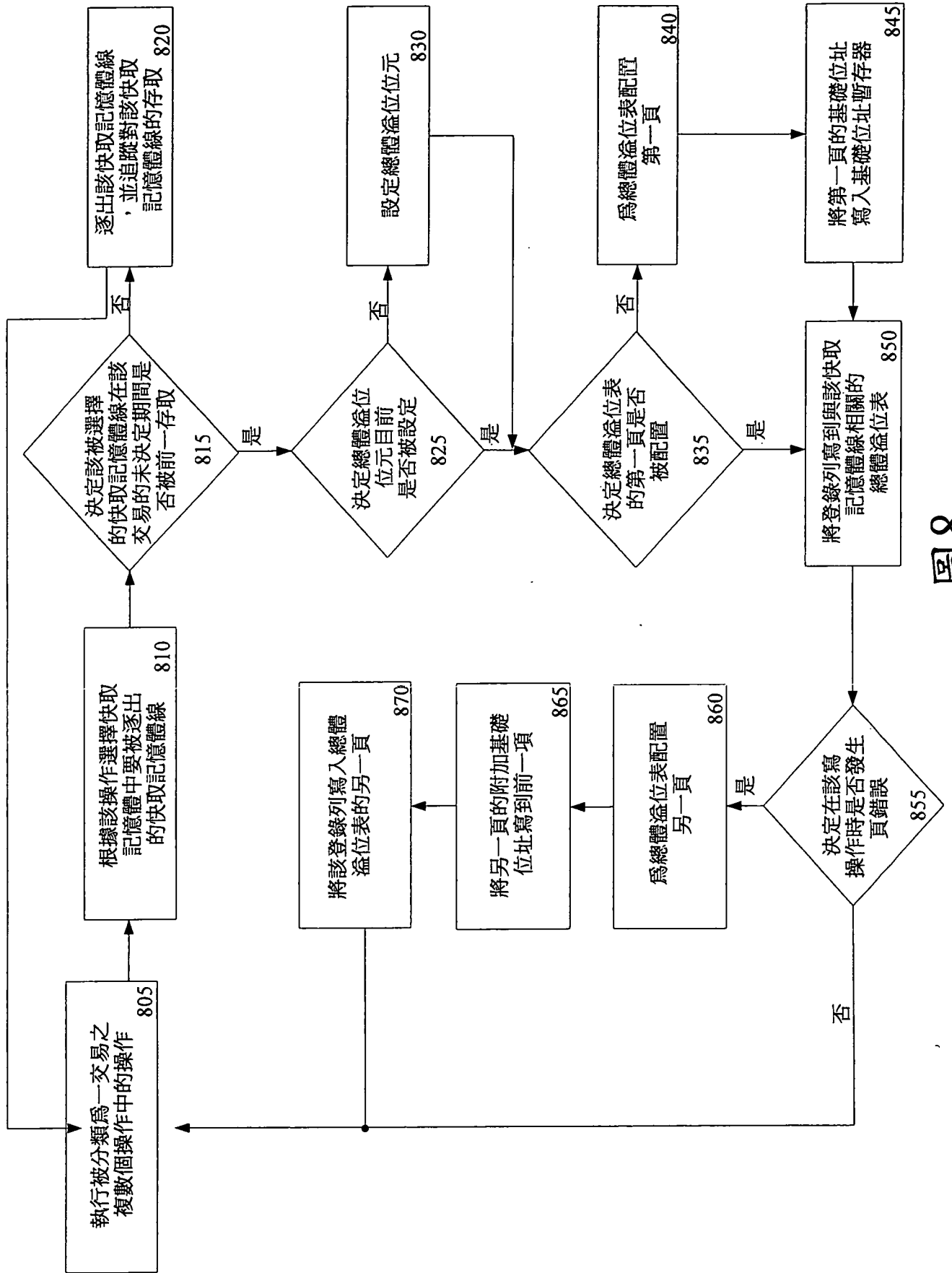


圖8