

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：97131980

※申請日期：97.8.21

※IPC 分類：G06F 9/48 (2006.01)

一、發明名稱：(中文/英文)

多工處理器及其任務切換方法 / MULTITASKING
PROCESSOR AND TASK SWITCH METHOD
THEREOF

二、申請人：(共 2 人)

姓名或名稱：(中文/英文)

1. 財團法人工業技術研究院 / INDUSTRIAL TECHNOLOGY RESEARCH
INSTITUTE

2. 國立交通大學 / NATIONAL CHIAO TUNG UNIVERSITY

代表人：(中文/英文) 1. 史欽泰 / SHIH, CHIN-TAY

2. 吳重雨 / WU, CHUNG-YU

住居所或營業所地址：(中文/英文)

1. 新竹縣竹東鎮中興路四段 195 號 / NO. 195, SEC. 4, CHUNG HSING
RD., CHUTUNG, HSINCHU 31040, TAIWAN, R. O. C.

2. 新竹市大學路 1001 號 / NO. 1001, DASYUE RD., HSINCHU CITY, 300,
TAIWAN (R. O. C.)

國 籍：(中文/英文) 1-2 中華民國/TW

三、發明人：(共 5 人)

姓 名：(中文/英文)

1. 林泰吉 / LIN, TAI-JI

2. 黃保瑞 / HUANG, PAO-JUI

3. 劉志尉 / LIU, CHIH-WEI

4. 陳信凱 / CHEN, SHIN-KAI

5. 王炳勳 / WANG, BING-SHIUN

國 籍：(中文/英文) 1-5 中華民國/TW

四、聲明事項：

主張專利法第二十二條第二項 第一款或 第二款規定之事實，其事實發生日期為： 年 月 日。

申請前已向下列國家（地區）申請專利：

【格式請依：受理國家（地區）、申請日、申請案號 順序註記】

有主張專利法第二十七條第一項國際優先權：

無主張專利法第二十七條第一項國際優先權：

主張專利法第二十九條第一項國內優先權：

【格式請依：申請日、申請案號 順序註記】

主張專利法第三十條生物材料：

須寄存生物材料者：

國內生物材料 【格式請依：寄存機構、日期、號碼 順序註記】

國外生物材料 【格式請依：寄存國家、機構、日期、號碼 順序註記】

不須寄存生物材料者：

所屬技術領域中具有通常知識者易於獲得時，不須寄存。

五、中文發明摘要：

一種多工處理器及其任務切換方法。任務切換方法包含下列步驟：多工處理器執行第一任務，而此第一任務的執行指令中包含了許多的切換點指令。一中斷事件發生，而此中斷事件將使多工處理器暫緩執行目前的第一任務，改為執行第二任務。多工處理器執行中斷事件之處理程序，並設置一任務切換旗標。中斷事件之處理程序結束後，多工處理器不進行任務切換，仍繼續執行第一任務，直到執行至第一任務執行指令中的切換點指令處，多工處理器才進行任務切換，改為執行第二任務。

六、英文發明摘要：

A multitasking processor and the task switching method thereof are provided. The task switching method includes the following steps. An interrupt occurs to change the processor from executing a first task to a second task. The first task includes a plurality of switch instructions, which are interleaved with normal instructions. The interrupt service routine in the disclosed processor asserts a flag to indicate the need for task switching without performing the task switching and the processor continues to execute the first task. Not until a switch instruction in the first task is executed does the processor switch to the second task.

七、指定代表圖：

(一)本案指定代表圖為：圖(2D)。

(二)本代表圖之元件符號簡單說明：

T201~T204：時間

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】

本發明是有關於一種多工處理器及其任務切換方法。

【先前技術】

隨著科技的蓬勃發展，使得通訊及多媒體的標準不斷的更新，眾家業者為了應付通訊及多媒體標準的更新速度，於是以可程式化處理器(programmable processor)取代傳統特定功能整合電路(application specified integrated circuit, ASIC)，整合到嵌入式系統中。而新一代的通訊及多媒體的應用傾向於提高多工處理器運算複雜度，達到以較低位元率提供高畫質內容的功能。為了符合通訊及多媒體上的即時(real-time)處理的需求(同時執行多個應用程式、及時反應使用者需求等)，可程式化處理器(programmable processor)於是使用作業系統(operating system)或微核心(micro kernel)動態即時管理的能力，並以分時(time slicing)多工(multitasking)的方式來實現。

在分時多工的環境下，可程式化處理器必須常常切換所執行的工作(或稱任務，task)，每進行一次任務切換，則必須進行一次內容切換(context switch)，將現在進行的任務狀態(包括暫存器以及其他任務資訊)全部存到堆疊(stack)中。現今可程式化處理器的發展趨勢傾向於增加暫存器的數量，以及加長暫存器的字元長度(word-length)，以便於藉由單一指令多重資料(single instruction multiple

data，簡稱 SIMD)的技術，利用資料層級平行度(data-level parallelism，簡稱 DLP)來提升運算能力。

市面上產品的微核心動態即時管理的方式大多以搶先式(preemptive)排序的為主。圖 1 為習知多工處理器之搶先式微核心的任務切換流程圖。當中斷(interrupt)發生的時候(如步驟 S101)，微核心會控制多工處理器先暫停所有的任務。接著會執行步驟 S102，先將中斷處理(interrupt handler)所需要的部份暫存器的內容保留(備份)至堆疊中。步驟 S103 會進行中斷事件之處理程序，此時微核心會控制多工處理器先儲存堆疊點，接著對所有的任務(包含執行中、等待被執行及中斷載入的任務)進行重新任務排程(reschedule)。在執行到步驟 S104 的時候，會依據重新任務排程的結果，比對是否有比目前執行中的任務具有更高優先權的任務。若有的話，便會做一次完整的內容切換，使多工處理器先載入優先權較高的任務(步驟 S110)。所謂內容切換，包含將原先任務的所有執行內容(暫存器的內容)備份至堆疊中，然後將新任務的所有執行內容存回暫存器中。在完成內容切換後，便可以開始執行新的任務(步驟 S111)。若是沒有更高優先權的任務，則會將步驟 S102 所備份的暫存器內容存回暫存器中(步驟 S120)，以便讓原先執行中的任務繼續執行(步驟 S121)。

圖 1A 為根據習知技術，說明圖 1 之任務時序圖。圖中橫軸表示時間 t 。於圖 1 中，搶先延遲時間為時間 T_{11} 及 T_{12} 。參照圖 1A，在時間 T_{101} 之前，多工處理器執行第一

任務。在時間 T101 發生中斷事件後，多工處理器暫停執行第一任務，並在時間 T101~T102 的期間中進行中斷事件之處理程序(包含中斷處理與重新任務排程)。其中，重新任務排程會對所有等待執行的任務重新排序。

當完成重新任務排程後(也就是在時間 T102 之後)，若重新任務排程顯示原先進行的第一任務為最優先處理的任務，則多工處理器將繼續執行第一任務；當完成重新任務排程後(也就是在時間 T102 之後)，若有一任務(第二任務)取代執行中的任務(第一任務)成為最優先處理的任務時，則多工處理器將任務切換改執行前述之第二任務，並在時間 T102~T103 的期間中進行內容切換。在內容切換期間，多工處理器會將第一任務之執行內容(所有暫存器的內容)備份至堆疊中，然後將第二任務的執行內容載入暫存器中。很明顯地，內容切換所需時間，需視暫存器的數量以及每一個暫存器的字元長度而定。在現今可程式化多工處理器的發展趨勢下，內容切換所需時間(即時間 T102~T103 的期間)會越來越長。在圖 1A 中，其時間 T_1 為搶先延遲(preemption latency)的時間，代表從發生中斷事件到完成內容切換所花的時間。當完成內容切換後(即時間 T103)，此時多工處理器已經準備好並且開始執行第二任務。

在即時處理的系統中，搶先延遲的時間(T_1)是一個考量重點。在習知搶先式微核心中，當執行內容切換時，會將所有的暫存器(包括使用中及未使用的暫存器)作備份，其中亦包括不再使用之暫存器。

【發明內容】

本發明提供一種多工處理器，此多工處理器包含處理單元以及切換旗標，其可以接受二個以上之任務指派。處理單元用以執行含有一切換點指令的指令集，其中該切換點指令為特定的處理器指令。其中，該處理單元執行具有至少一切換點指令的第一任務。當發生中斷事件時，處理單元執行中斷事件之處理程序以判斷是否需要任務切換，並依據判斷結果設定該切換旗標後，該處理單元即繼續執行第一任務，直到處理單元執行該第一任務中之該至少一切換點指令時檢查該切換旗標；若切換旗標表示該處理單元於進行前述中斷事件之處理程序時判斷需進行任務切換，則處理單元即進行任務切換改執行第二任務。若該切換旗標表示處理單元於進行中斷事件之處理程序時判斷不需任務切換，則處理單元則繼續執行第一任務。

本發明亦提出一種多工處理器之任務切換方法。此任務切換方法包括下述步驟。首先由多工處理器執行第一任務。假設某一事件表示該多工處理器應進行任務切換，以便由執行第一任務切換為執行第二任務。當發生該事件時，使該多工處理器暫緩進行該任務切換，並繼續執行第一任務，直至該多工處理器執行至第一任務中之可切換點，逕行該任務切換以切換執行第二任務。

為讓本發明之上述特徵和優點能更明顯易懂，下文特舉較佳實施例，並配合所附圖式，作詳細說明如下。

【實施方式】

如前述所提及，現在的多工處理器設計趨勢傾向於增加暫存器的數量及字元的長度，這使得習知搶先式微核心以分時多工來執行多個任務時，在任務切換過程花費過多的時間儲存所有的暫存器。有鑑於此，本發明將提出一多工處理器，利用增加一切換點指令及一切換旗標，搭配一任務切換方法。

以下諸實施例將說明，當發生某一事件(例如中斷事件)時，本發明之多工處理器如何進行任務切換。前述某一事件表示該多工處理器應進行「任務切換」，以便由執行「第一任務」切換執行「第二任務」。

當發生前述足以觸發任務切換的中斷事件時，本實施例之多工處理器暫緩進行任務切換，並繼續執行目前的任務(第一任務)，直至多工處理器執行至第一任務中之可切換點，才進行任務切換以切換執行新的任務(第二任務)。前述之可切換點可以是第一任務中，進行任務切換所需花費資源較少的地方。另外，該可切換點也可以是第一任務中，為滿足即時性需求(real-time requirements)所設置的任務切換點。所屬領域中具有通常知識者可以任何方式實現所述可切換點。例如，在第一任務中任務切換所需花費資源較少的地方配置某一特定指令(切換點指令)，以標示所述可切換點之位置並觸發多工處理器進行「任務切換」。

值得注意的是，本實施例與下述諸實施例所稱足以觸發「任務切換」的事件(或是中斷事件)，可以是任何形式

的事件。例如，該事件可以是多工處理器外部或內部所發生的軟體中斷事件或硬體中斷事件。又例如，該事件可以是固定週期發生的計時器事件(timer events)。以下再舉另一實施例，以說明本實施例之詳細實現方式。

圖 2A 為根據本發明之一實施例的多工處理器之任務切換方法的中斷事件處理流程圖。圖 2B 為根據本發明之一實施例的多工處理器之任務切換方法的任務切換處理流程圖。圖 2C 為根據本發明一實施例的可切換點(或切換點指令)配置示意圖。圖 2D 為根據本發明一實施例的多工處理器之任務切換方法的時序圖。圖 2D 中橫軸表示時間 t 。

先參照圖 2D，在時間 T201 之前，多工處理器會載入並執行第一任務，其中第一任務會包含至少一個切換點指令。若該多工處理器執行第一任務中，未發生中斷及未執行切換點指令，則多工處理器會繼續執行第一任務。請同時參照圖 2A 與圖 2D，在第一任務執行的過程中，假設於時間 T201 發生中斷事件。此時多工處理器會先暫停執行第一任務，並備份部份暫存器的內容，以保留中斷事件之處理程序所需的部份暫存器(步驟 S210)。接著執行中斷事件之處理程序(步驟 S211)對所有的任務(包括執行中、等待被執行以及觸發中斷的任務)進行重新任務排程。

若重新任務排程的結果顯示第一任務的優先權為最大(步驟 S212)，亦即判斷結果為「否」，則直接執行步驟 S214 以還原中斷事件之處理程序所保留的暫存器而不設置切換旗標。接下來於時間 T202 之後多工處理器會繼續執行第

一任務(步驟 S215)。反之，若重新任務排程的結果顯示第一任務不是優先權最大的任務(步驟 S212)；代表有更高優先權的任務(在此假設最高優先權的任務為第二任務)；在多工處理器在完成步驟 S212 後，多工處理器會設置切換旗標(步驟 S213)，接著於時間 T202 之前還原中斷事件之處理程序所保留的暫存器 (步驟 S214)。在時間 T202 之後，假設在時間 T203 之前，第一任務沒有發生中斷事件及執行切換點指令，多工處理器不進行任務切換而會繼續執行第一任務(步驟 S215)，直到第一任務中的切換點指令被執行(即圖 2D 的時間 T203)。

請參照圖 2B 與圖 2D，當多工處理器執行切換點指令時，步驟 S221 會檢查切換旗標是否已被設置。如多工處理器未曾進行步驟 S213，亦即未設置切換旗標，多工處理器執行第一任務中的切換點指令時將繼續執行第一任務(步驟 S215)。反之，若多工處理器已於步驟 S213 設置切換旗標，則多工處理器在完成步驟 S221 後接著進行「內容切換(context switch)」操作後改執行第二任務；也就是說，當多工處理器執行切換點指令時，若中斷事件之處理程序中重新任務排程的結果表示多工處理器需任務切換，則多工處理器進行內容切換後改執行第二任務。在本實施例中，內容切換包含步驟 S222、S223 與 S224。在步驟 S222 中，多工處理器會記錄堆疊點及將第一任務的執行內容存入堆疊以備份欲保留的任務執行內容。在備份完成後，多工處理器清除切換旗標(步驟 S223)，然後載入第二任務的

執行內容(步驟 S224)。在完成上述動作後，多工處理器就已完成內容切換的操作，並於時間 T204 之後，改執行第二任務(步驟 S225)，同時第一任務就會被閒置至下一次中斷事件發生或第二任務結束。於圖 2D 中，搶先延遲時間為時間 T_{21} 及 T_{22} 。

上述切換點指令於第一任務中的位置，可以是第一任務中使用系統資源較少的位置，亦即多工處理器進行任務切換時需保留之暫存器數量較少的位置。請參照圖 2C，其橫軸表示第一任務之程式執行順序，縱軸表示多工處理器進行任務切換時需保留的暫存器數量。一般而言，可以利用程式語言編譯器進行靜態分析而獲得圖 2C 所示的曲線。在配置切換點指令前，程式設計者可以利用編譯器對第一任務的程式碼 210 作靜態分析，以取得程式碼對暫存器的使用狀態，這些仍將使用的暫存器就是多工處理器進行任務切換時需保留的暫存器。接著編譯器可以依據靜態分析結果將切換點指令配置在程式碼中需保留暫存器數量最少的地方(例如將切換點指令 214 配置在位置 204)，並測試是否符合搶先延遲時間限制。如果不能符合搶先延遲時間限制，編譯器會放寬使用資源的限制並對配置切換點指令不足的程式碼目標區段(target section)做進一步的分析，接著將切換點指令配置在此目標區段中的最佳替代點；此最佳替代點為目標區段中，多工處理器進行任務切換時，需保留暫存器數量最少的地方，如圖 2C 中將切換點指令 211、212 與 213 分別配置在位置 201、202 及 203。

以上述動作反覆執行到第一任務程式碼中兩相鄰切換點指令間的搶先延遲時間不會過長。為了避免在程式碼中配置過多的切換點指令，使得第一任務執行的效能大幅降低，最後要再對程式碼做分析，合併搶先延遲時間過短的兩相鄰切換點指令，以確保編譯後的第一任務中的切換點指令 211~214 數量不會過多及切換點指令 211~214 各自被配置在目標區段中需保留暫存器數量最少的地方 201~204。

上述切換點指令配置方式為第一種配置方式，除此之外，在此提出第二種切換點指令配置的方式。此方式為先將切換點指令配置於第一任務各個子程式(子任務)的結尾端；因子程式結束時，只會保留子程式結束運作完的結果以傳遞至主程式或下個子程式，故其保留暫存器數量(亦即使用系統資源)會最少。接著測試兩切換點指令的配置是否滿足搶先延遲時間，並根據測試結果配置額外切換點指令於最佳替代點。而所使用的測試方式與上述配置方式相同，故不再贅述。

然而，子程式的末端並非一定為保留暫存器數量最少的位置，故可依據上述方式加以變動，以符合搶先延遲時間的限制。切換點指令的第三種配置方式為先對各子程式碼作靜態分析，並將切換點指令配置於保留暫存器數量最少的地方，且此地方不一定為子任務的末端。接著，將整個程式作延遲時間測試以了解切換點指令配置的位置及密度是否符合延遲時間限制，如不符合，則以上述切換點指令配置方式再安插切換點指令。而在其他的實施方式中，

切換點指令的配置方式可為上述第一、第二及第三種切換點指令的配置方式的組合及其他配置方式。例如，切換點指令的配置方式，也可以是在第一任務中，為滿足即時性需求(real-time requirements)所設置的任務切換點。

由前述之實施例可知，本發明提出之多工處理器於執行第一任務時；當發生一中斷事件，多工處理器即暫緩第一任務執行，改執行中斷事件之處理程序。藉此，本發明仍保留對於中斷事件的快速處理能力。

由前述之實施例可知，本發明提出之多工處理器，因切換點指令配置於第一任務中使用系統資源較少的地方，使得多工處理器在使用系統資源較少的時候執行任務切換處理，進行任務切換的時間也相對較少。

圖 3 為根據本發明之一實施例的多工處理器 300 之方塊圖。請參照圖 3，多工處理器 300 包括處理單元以及切換旗標 350。處理單元用以執行含有一切換點指令的指令集，其中該切換點指令為特定的處理器指令。其中，該處理單元執行具有至少一切換點指令的一第一任務。當發生中斷事件時，該處理單元進行該中斷事件之處理程序以判斷是否需要任務切換，並依據判斷結果設定切換旗標後，該處理單元即繼續執行第一任務，直至處理單元執行第一任務中之該至少一切換點指令時檢查切換旗標；若切換旗標表示該處理單元於進行中斷事件之處理程序時判斷需任務切換，則處理單元即進行任務切換而改執行第二任務；

以及若切換旗標表示該處理單元於進行中斷事件之處理程序時判斷不需任務切換，則處理單元則繼續執行第一任務。

所技術領域之技術人員可以任何方式實現處理單元。例如，圖 3 所示的處理單元包括指令提取級 310、指令解碼級 320、執行級 330、資料存取級 340、及閘 AND1 等。圖 3 中並未繪出全部構件與全部信號路徑(例如控制/設定信號路徑等)，以免圖式太過繁雜。指令提取級 310 會循序提取任務程式碼中的指令，並將指令傳送到指令解碼級 320 進行解碼。當解碼完成後，解碼後的指令會傳送到執行級 330，以執行此指令。

指令解碼級 320 包括解碼器 321 及暫存器單元 322。解碼器 321 會將指令進行解碼，以便控制執行級 330 依據指令進行運算。依據解碼器 321 的解碼結果，運算元會從暫存器單元 322 傳送至執行級 330，以便進行該指令之運算。完成運算後，依據解碼器 321 的解碼結果，執行級 330 會將運算結果寫回至暫存器單元 322，或是透過資料存取級 340 寫回至資料記憶體 341。

於本實施例中，指令提取級 310 會先循序提取第一任務的程式碼中的各個指令。其中，第一任務的程式碼中已經配置多個切換點指令，其切換點指令的配置方式可以參照圖 2C 及相關說明，故不再贅述。指令提取級 310 所提取的指令，經解碼器 321 解碼後傳送到執行級 330 執行。

於本實施例中，多工處理器於第一任務中每一個切換點進行任務切換時，都必需備份不同的執行內容。於本實

施例中，將針對每一個切換指令建立專屬的存活暫存器清單(live register list)，以便記錄多工處理器進行任務切換時需要備份的暫存器。每一個切換點指令各自包含一個位址，此位址指向該切換點所對應的存活暫存器清單。以下將利用存活暫存器表(live register table) 342 記錄所有的存活暫存器清單。

在本實施例中，存活暫存器表 342 是配置於資料記憶體 341 中。所屬技術領域之技術人員可以任何形式實施上述存活暫存器表 342。例如，圖 3A 為根據本發明一實施例，說明圖 3 存活暫存器表 342 示意圖。參照圖 3A，因各個切換點要保留的執行內容不盡相同，故要建立多個存活暫存器清單 343(例如圖 3A 之存活暫存器清單 343_1 ~ 343_n)，以便為各個切換點指令記錄其需要保留的暫存器資訊。例如，圖 2C 中切換點指令 211 要存留的暫存器資訊被記錄在圖 3A 的存活暫存器清單 343_1 中，切換點指令 212 要存留的暫存器資訊被記錄在圖 3A 的存活暫存器清單 342_2，以下類推之。以存活暫存器清單 343_3 為例，存活暫存器清單 343_3 中，其記錄切換點指令 213 執行時(即圖 2C 之位置 203)，多工處理器進行任務切換時，所需儲存備份的暫存器(0 表示不儲存，1 表示要儲存)。例如，若存活暫存器清單 343_3 的內容為「011...」，表示暫存器單元 320 中暫存器 R0 的內容不要儲存，暫存器 R1 的內容要儲存，暫存器 R2 的內容要儲存，以此類推。如圖 3A 所示，所有的存活暫存器清單 343_1~343_n 都會記錄在存活

暫存器表 342 中，其中存活暫存器表 342 的寬度 m (代表系統中暫存器個數)及長度 n (代表切換點指令個數)可以依系統環境的需求以及本領域通常知識者的設計而有所變動。

綜上所述，請參照圖 3 及圖 3A，指令提取級 310 會循序提取第一任務程式碼中的指令，並將指令提供給解碼器 321 解碼。當執行級 330 接收到解碼後的指令，則會依照指令的要求而產生不同的動作。暫存器單元 322 包含多個暫存器，用以記錄多工處理器 300 的執行內容。如前實施例所述，在第一任務執行的過程中，當系統於時間 T201 發生中斷事件時(在本實施例是以第二任務觸發中斷)，此時執行級 330 會執行中斷事件之處理程序。在中斷事件之處理程序中，執行級 330 先暫停第一任務的執行，且將暫存器單元 322 中部份暫存器的資料備份至堆疊(或是資料記憶體 341)，以便保留中斷事件之處理程序所需的暫存器。接著對所有的任務進行重新任務排程(步驟 S211)。所屬技術領域之技術人員可以任何技術實施前述中斷事件之處理程序與重新任務排程，例如應用習知技術來實現前述中斷處理與重新任務排程。

當重新任務排程後，多工處理器 300 會檢視第一任務是否為最高優先權的任務(步驟 S212)。當第一任務不為最高優先權的任務時，則多工處理器 300 會設置切換旗標 350 (步驟 S213)；相對的，當第一任務為最高優先權的任務時，多工處理器 300 則重置(reset)或清除切換旗標 350。前述操

作流程可參照圖 2A 及其相關說明。上述步驟 S211、S212、S213 之操作，可以由指令解碼級 320、或執行級 330、或多工處理器 300 內部其他控制電路(未繪示)實行之。

不論切換旗標是否已被設置，在執行級 330 完成重新任務排程後，多工處理器還原中斷處理所保留的暫存器(步驟 S214)。完成暫存器單元 322 的資料回復後，執行級 330 會繼續執行第一任務。也就是說，多工處理器 300 在完成中斷事件之處理程序後不進行任務切換而繼續執行第一任務，直到多工處理器 300 執行至第一任務中的切換點指令。

當指令提取級 310 將第一任務的切換點指令送至解碼器 321 時(即圖 2D 的時間 T203)，解碼器 321 會發出任務切換信號至及閘 AND1。在切換旗標 350 尚未被設置的狀態下(亦即切換旗標 350 為邏輯「0」)，解碼器 321 所發出的任務切換信號會被及閘 AND1 阻擋而無法到達指令提取級 310。在切換旗標 350 被設置的狀態下(亦即切換旗標 350 為邏輯「1」)，解碼器 321 所發出的任務切換信號會經由及閘 AND1 而到達指令提取級 310。指令提取級 310 便可以依據此一任務切換信號來決定是否繼續提取第一任務程式碼的指令，或是改提取任務切換程序指令以便執行第二任務程式。

因此，當指令解碼級 320 執行切換點指令時，若切換旗標 350 未被設置，則多工處理器 300 繼續執行第一任務。反之，當指令解碼級 320 執行切換點指令時，若切換旗標

350 被設置，則多工處理器 300 進行內容切換(如圖 2D 所示之 T203~T204 期間)以執行第二任務。在進行內容切換時，多工處理器 300 依據目前所執行的切換點指令，從存活暫存器表 342 查找出對應此切換點指令的存活暫存器清單(在此假設對應為存活暫存器清單 343_3)。多工處理器 300 會根據此切換點指令對應的存活暫存器清單內容，儲存備份暫存器單元 322 中的存活暫存器，並將這些指定的暫存器的內容存入堆疊中(或資料記憶體 341 中)，如此就能將多工處理器 300 中第一任務在時間點 T203 的執行內容加以備份(步驟 S222)。在備份完成後，多工處理器 300 清除切換旗標(步驟 S223)，然後載入第二任務的執行內容(步驟 S224)，以完成任務切換。上述步驟 S222、S223、S224 及其他操作，可以由指令解碼級 320、或執行級 330、或多工處理器 300 內部其他控制電路(未繪示)進行之，需視不同應用需求而採行不同的設計手段。

在完成任務切換後，指令提取級 310 會開始循序提取第二任務程式碼的指令，並使多工處理器 300 開始執行第二任務。前述操作流程可參照圖 2B 及其相關說明。

在上述本發明之一實施例的多工處理器，在完成中斷事件之處理程序後不進行任務切換而繼續執行第一任務，直到執行至第一任務中的切換點指令。由於切換點指令皆配置在第一任務中使用系統資源較少(亦即保留的暫存器數量較少)的位置，故當多工處理器執行至第一任務中的切換點指令且需進行任務切換時，內容切換所需備份的暫存

器數量亦較少。配置切換點指令雖會影響系統些微的效能，但對於整體程式執行時間上並不會造成明顯影響。藉此，本實施例可降低多工處理器進行任務切換的時間及消耗的功率，並可減少多工處理器在多個任務間進行任務切換所增加的硬體成本(例如堆疊的容量)。

特別要注意的是，上述實施例所提之切換旗標可內建於暫存器單元中，亦可於多工處理器中另外設置。其內建於暫存器單元中的意思為本發明亦可以利用多工處理器未用到的暫存器及記憶體空間來實現切換旗標。另外，雖然上述實施例中將每一個切換點指令所需之存活暫存器清單集中放置在存活暫存器表 342 中，但是其實現方式不應以此受限。例如，在其他實施例中可能將存活暫存器清單編碼至對應的切換點指令中，使得存活暫存器清單可以隨著切換點指令被指令提取級 310 提取，而不需要另外從資料記憶體 341 中提取存活暫存器清單 343。

雖然本發明已以較佳實施例揭露如上，然其並非用以限定本發明，任何所屬技術領域中具有通常知識者，在不脫離本發明之精神和範圍內，當可作些許之更動與潤飾，因此本發明之保護範圍當視後附之申請專利範圍所界定者為準。

【圖式簡單說明】

圖 1 為習知多工處理器之搶先式微核心的任務切換流程圖。

圖 1A 為根據習知，說明圖 1 之任務時序圖。

圖 2A 為根據本發明之一實施例的多工處理器之任務切換方法的中斷事件處理流程圖。

圖 2B 為根據本發明之一實施例的多工處理器之任務切換方法的任務切換處理流程圖。

圖 2C 為根據本發明一實施例的切換點指令配置示意圖。

圖 2D 為根據本發明一實施例的多工處理器之任務切換方法的時序圖。

圖 3 為根據本發明之一實施例的多工處理器之方塊圖。

圖 3A 為根據本發明一實施例，說明圖 3 存活暫存器表示示意圖。

【主要元件符號說明】

S101~S104、S110、S111、S120、S121、S210~S215、
S221~S225：步驟

T120、T101、T103~T105、T110、T201~T204：時間

201~204：暫存器數量最少的位置

210：程式碼

211~214：切換點指令

300：多工處理器

350：切換旗標

322：暫存器單元

- 341：資料記憶體
- 342：存活暫存器表
- 343~343_n：存活暫存器清單
- 310：指令提取級
- 320：指令解碼級
- 321：解碼器
- 330：執行級
- 340：資料存取級
- AND1：及閘

十、申請專利範圍：

1.一種多工處理器，可以接受二個以上之任務指派，該多工處理器包括：

一切換旗標；以及

一處理單元，用以執行由該多工處理器指令集組成的任務，其中該多工處理器指令集包含一切換點指令，該切換點指令為該切換旗標對應的中斷事件處理指令；

其中該處理單元執行具有至少一該切換點指令的一第一任務；當發生一中斷事件，該處理單元進行該中斷事件之處理程序並判斷是否需要任務切換及依據判斷結果設定該切換旗標後，該處理單元即繼續執行該第一任務，直至該處理單元執行該第一任務中之該至少一切換點指令。

2.如申請專利範圍第1項所述之多工處理器，更包括：

若該切換旗標已被設定，即該處理單元於進行該中斷事件之處理程序時，判斷需任務切換，該處理單元即進行任務切換改執行一第二任務；以及

若該切換旗標未被設定，即該處理單元於進行該中斷事件之處理程序時，判斷不需任務切換，該處理單元則繼續執行該第一任務。

3.如申請專利範圍第2項所述之多工處理器，更包括：

一存活暫存器清單，用以記錄該切換點指令所對應的存活暫存器；其中存活暫存器是該處理單元執行至該切換點指令且需任務切換時，所需要儲存備份之暫存器；以及

一存活暫存器表，其內容包括各個切換點指令所對應之該存活暫存器清單。

4.如申請專利範圍第 3 項所述之多工處理器，其中該切換點指令包含一位址指向該切換點所對應的存活暫存器清單；若該處理單元執行至該至少一切換點指令且該切換旗標已被設定，該處理單元則進行任務切換改執行一第二任務；其中，該任務切換包括根據該至少一切換點指令所對應之存活暫存器清單，儲存所有的存活暫存器。

5.如申請專利範圍第 4 項所述之多工處理器，其中該存活暫存器清單及該存活暫存器表的儲存位置為系統之資料記憶體。

6.如申請專利範圍第 4 項所述之多工處理器，其中該處理單元進行之該中斷事件之處理程序包括執行一重新任務排程；若該重新任務排程的結果表示該處理單元需任務切換，該處理單元即設定該切換旗標；以及若該重新任務排程的結果表示該處理單元不需任務切換，該處理單元則重置或清除該切換旗標。

7.如申請專利範圍第 6 項所述之多工處理器，其中該重新任務排程包括比較該第一任務與該第二任務之優先權。

8.如申請專利範圍第 1 項所述之多工處理器，其中該中斷事件包括該處理器之外部或內部發生的軟體中斷事件或硬體中斷事件。

9.如申請專利範圍第 1 項所述之多工處理器，其中該中斷事件包括固定週期發生的計時器事件。

10.如申請專利範圍第 1 項所述之多工處理器，其中該切換點指令於該第一任務之位置，是該第一任務中的子任務的結尾處。

11.如申請專利範圍第 1 項所述之多工處理器，其中該切換點指令於該第一任務之位置，是該第一任務中使用系統資源較少之位置。

12.如申請專利範圍第 1 項所述之多工處理器，其中該切換點指令於該第一任務之位置，是該第一任務中為滿足即時性需求所設置的任務切換點。

13.一種多工處理器之任務切換方法，包括：

一多工處理器執行一第一任務，其中該第一任務具有至少一切換點；

若發生一中斷事件，該多工處理器在完成該中斷事件之處理程序及判斷是否需要任務切換後繼續執行該第一任務，直到該多工處理器執行至該第一任務中的該至少一切換點；

若該中斷事件之處理程序的結果表示該多工處理器需任務切換，當該多工處理器執行至該至少一切換點時，該多工處理器進行任務切換改執行一第二任務；以及

若該中斷事件之處理程序的結果表示該多工處理器不需任務切換，當該多工處理器執行至該至少一切換點時，該多工處理器繼續執行該第一任務。

14.如申請專利範圍第 13 項所述之多工處理器之任務切換方法，其中該中斷事件之處理程序包括：

進行一重新任務排程；以及

依據該重新任務排程的結果，決定是否任務切換。

15.如申請專利範圍第 14 項所述之多工處理器之任務切換方法，其中該重新任務排程包括比較該第一任務與該第二任務之優先權。

16.如申請專利範圍第 13 項所述之多工處理器之任務切換方法，其中該中斷事件包括該處理器之外部或內部發生的軟體中斷事件或硬體中斷事件。

17.如申請專利範圍第 13 項所述之多工處理器之任務切換方法，其中該中斷事件包括固定週期發生的計時器事件。

18.如申請專利範圍第 13 項所述之多工處理器之任務切換方法，其中該切換點於該第一任務之位置，是該第一任務中的子任務的結尾處。

19.如申請專利範圍第 13 項所述之多工處理器之任務切換方法，其中該切換點於該第一任務之位置，是該第一任務中使用系統資源較少之位置。

20.如申請專利範圍第 13 項所述之多工處理器之任務切換方法，其中該切換點於該第一任務之位置，是該第一任務中為滿足即時性需求所設置的任務切換點。

21.一種多工處理器之任務切換方法，包括：

一多工處理器執行第一任務；

發生一事件，其中該事件表示該多工處理器應進行一任務切換以便由執行該第一任務切換為執行第二任務；以及

該處理器暫緩進行該任務切換，並繼續執行該第一任務，直至該多工處理器執行至該第一任務中之可切換點，逕行該任務切換以執行該第二任務。

22.如申請專利範圍第 21 項所述之多工處理器之任務切換方法，其中該事件為外部或內部發生的軟體或硬體中斷事件。

23.如申請專利範圍第 21 項所述之多工處理器之任務切換方法，其中該事件為固定週期發生的計時器事件。

24.如申請專利範圍第 21 項所述之多工處理器之任務切換方法，其中該切換點於該第一任務之位置，是該第一任務中的子任務的結尾處。

25.如申請專利範圍第 21 項所述之多工處理器之任務切換方法，其中該可切換點於該第一任務之位置，是該第一任務中使用系統資源較少之位置。

26.如申請專利範圍第 21 項所述之多工處理器之任務切換方法，其中該切換點於該第一任務之位置，是該第一任務中為滿足即時性需求所設置的任務切換點。

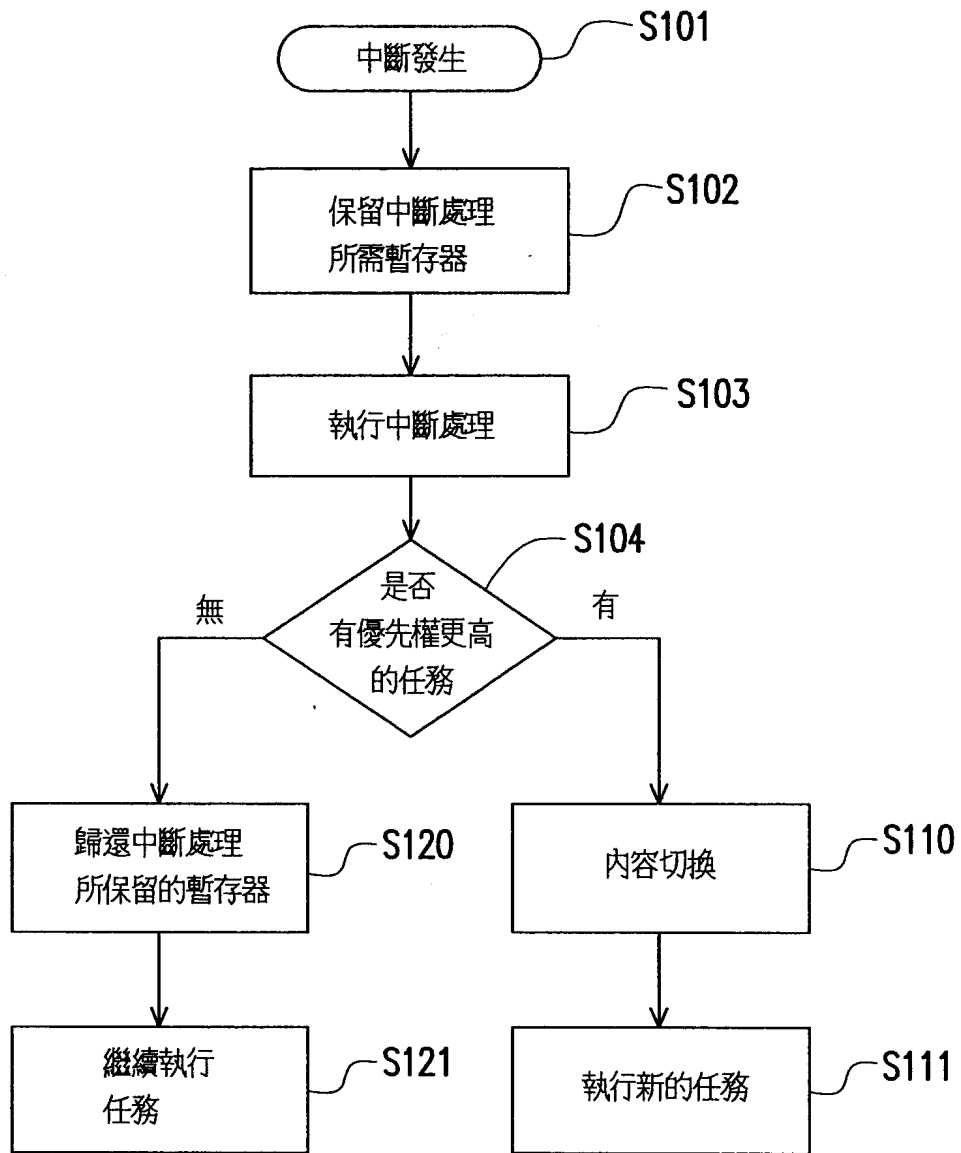


圖 1

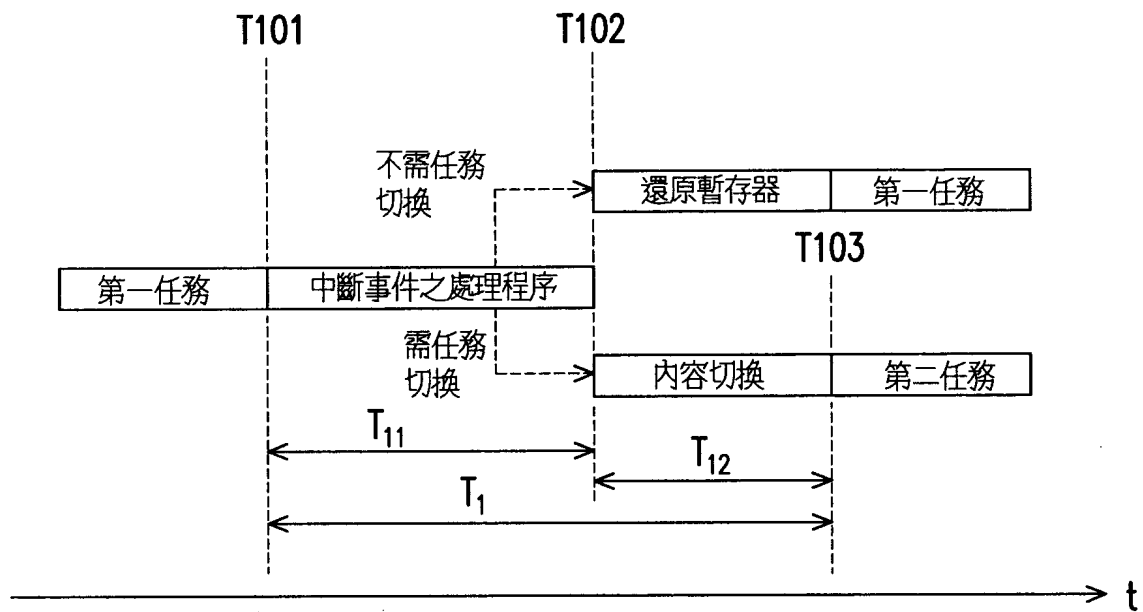


圖 1A

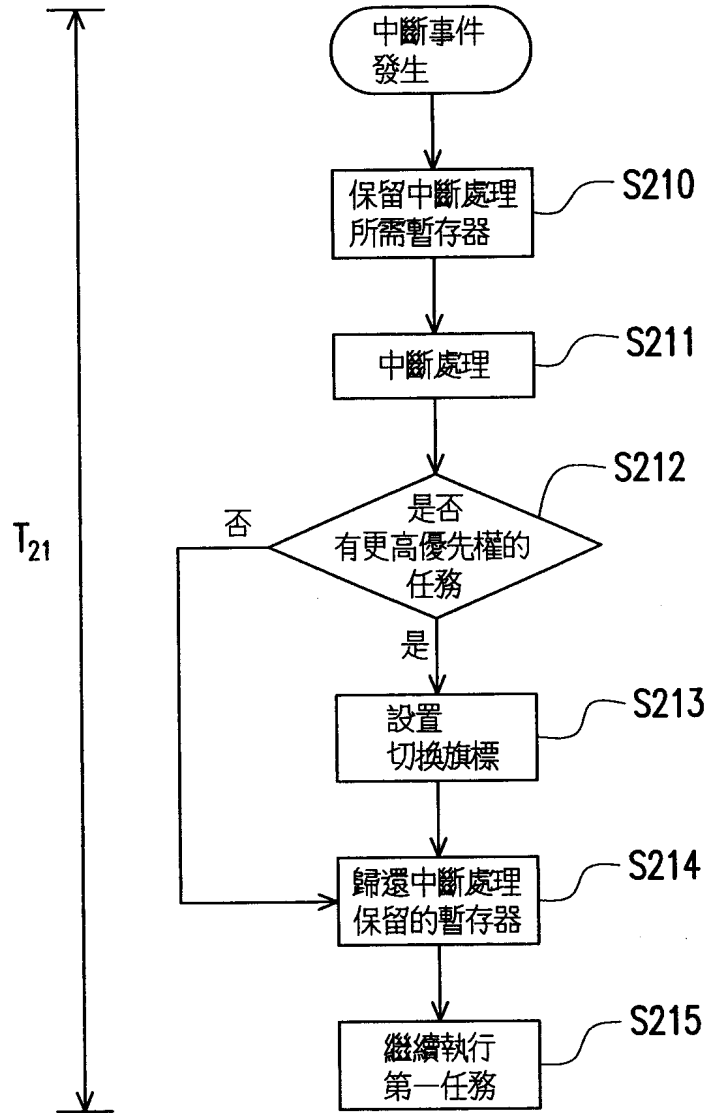


圖 2A

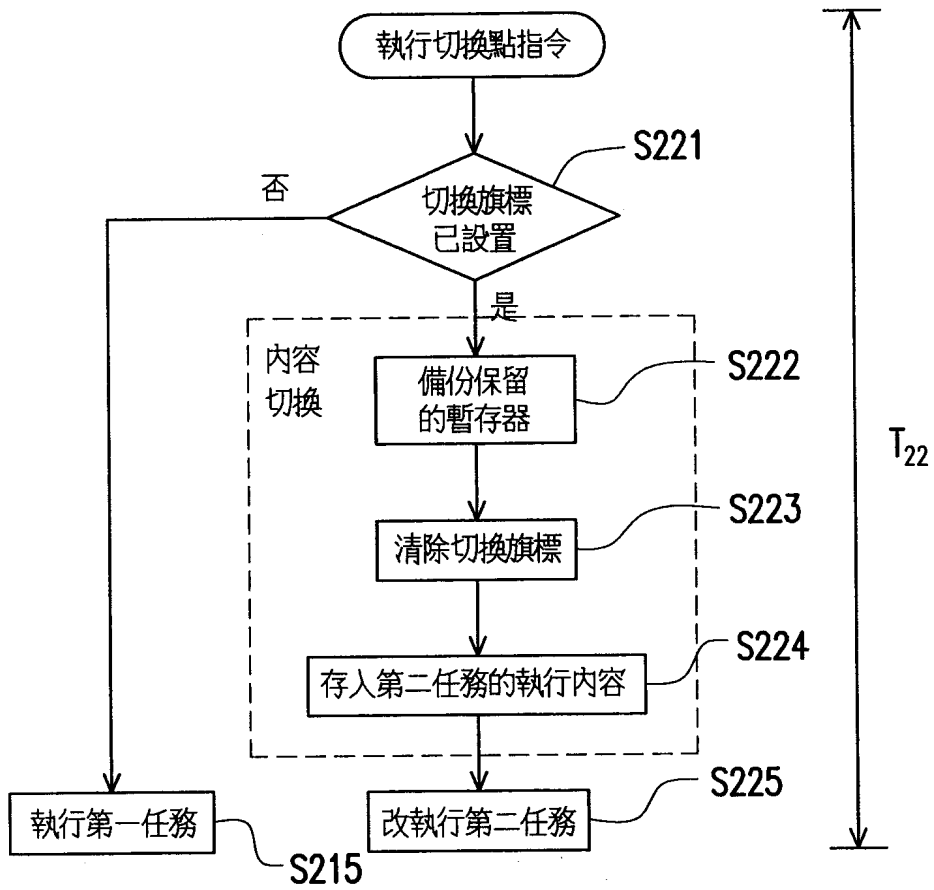


圖 2B

27082TW_J

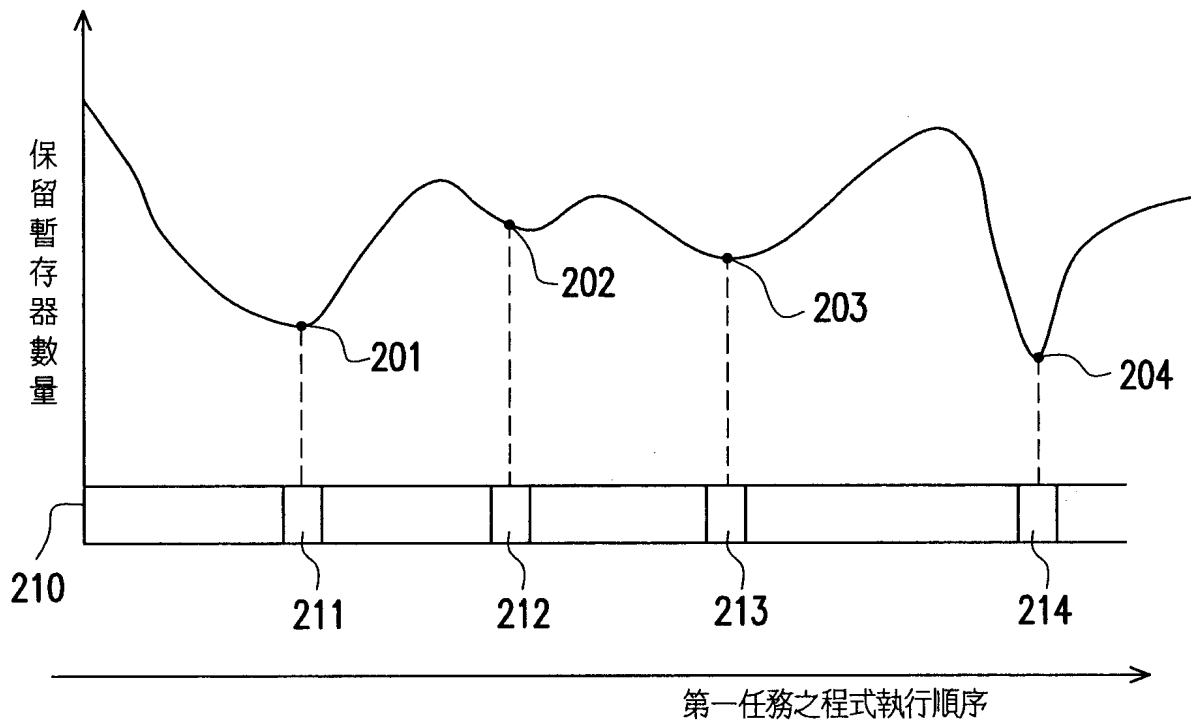


圖 2C

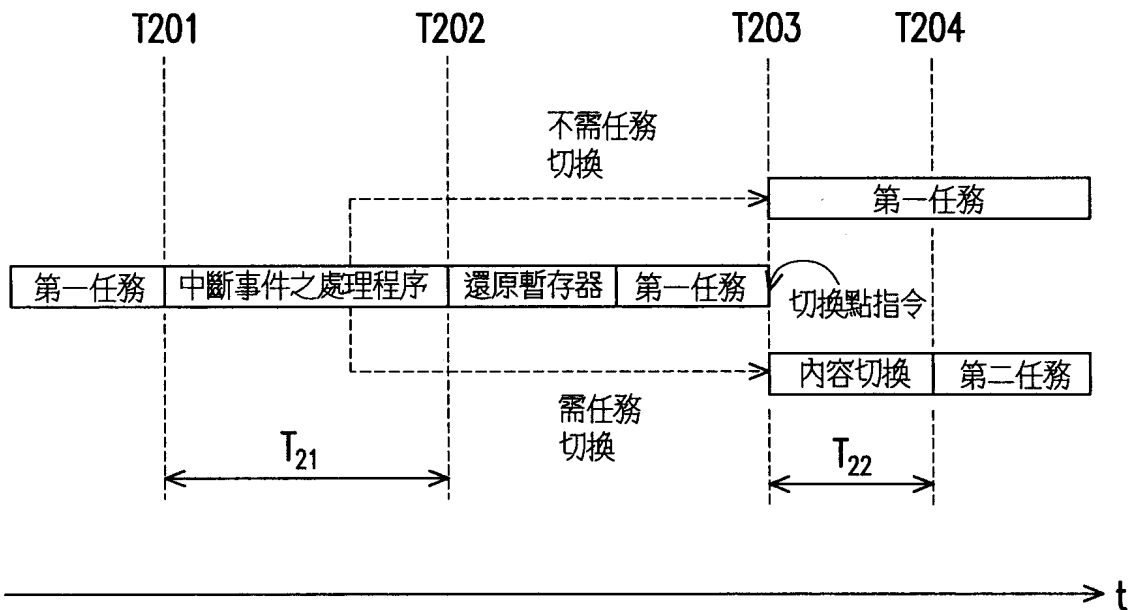
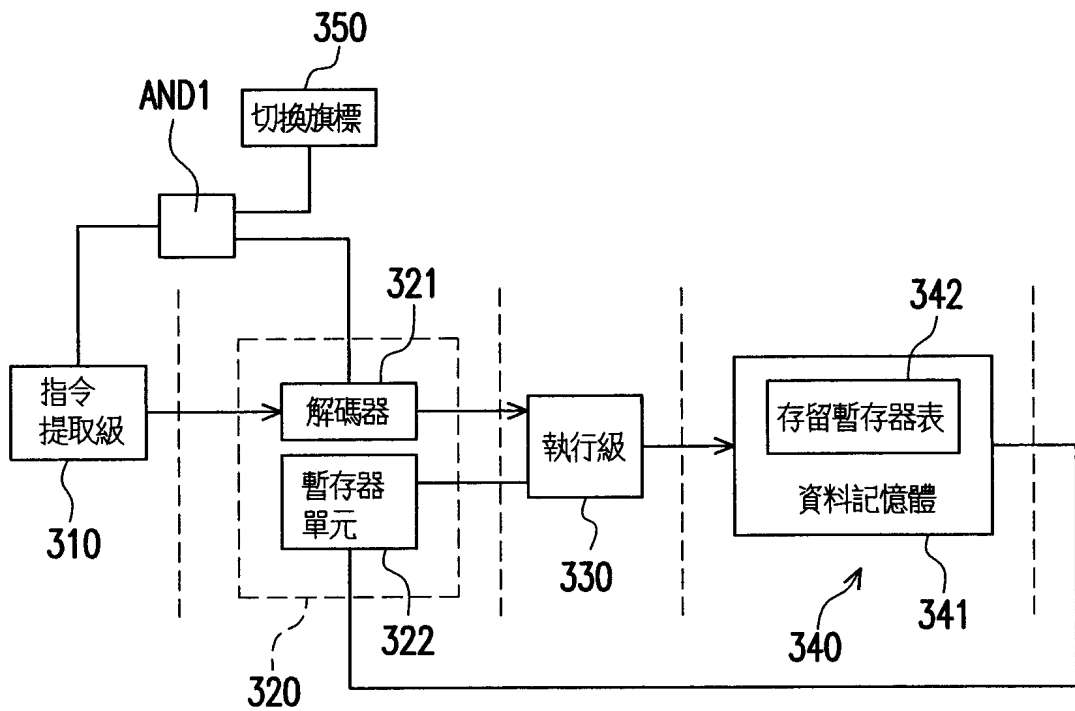


圖 2D



300

圖 3

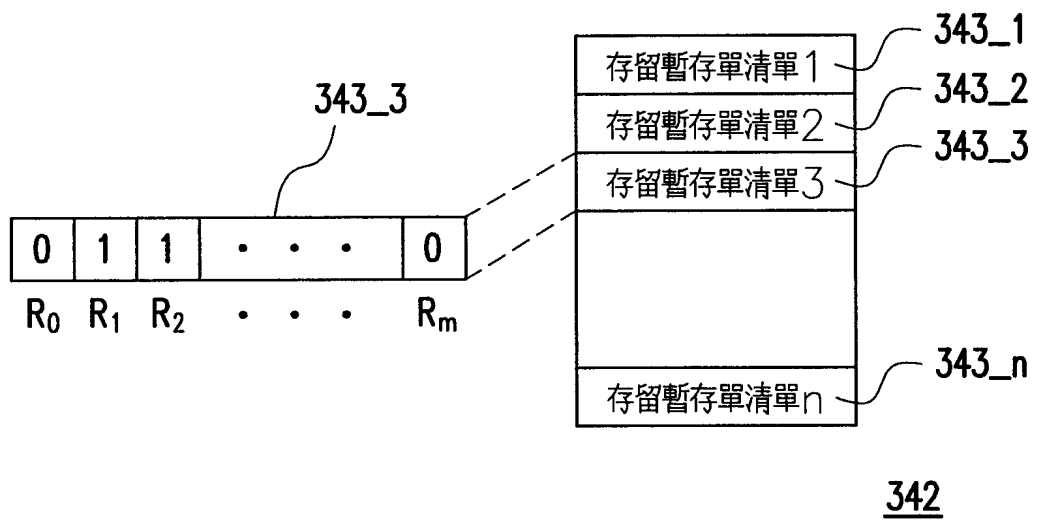


圖 3A