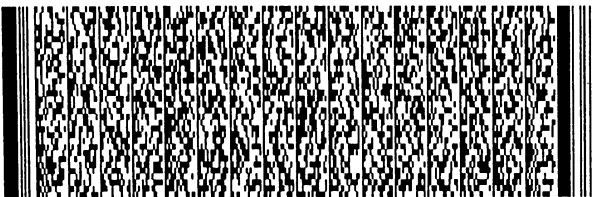


申請日期： 89.12.14	案號： 89126753
類別： G06F 9/48	

(以上各欄由本局填註)

公告本		發明專利說明書	589580
一、 發明名稱	中文	執行高速低費用上下文轉換之裝置及方法	
	英文	DEVICE AND METHOD FOR PERFORMING HIGH-SPEED LOW OVERHEAD CONTEXT SWITCH	
二、 發明人	姓名 (中文)	1. 唯塔利 蘇克尼克 2. 亞歷山大 米力特斯基 3. 亞米特 朵爾	
	姓名 (英文)	1. VITALY SUKONIK 2. ALEXANDER MIRETSKY 3. AMIT DOR	
	國籍	1. 以色列 2. 以色列 3. 以色列	
	住、居所	1. 以色列卡特塞市卡特塞曼那許路705號 2. 以色列海發市米許曼哈唯爾路2之7號 3. 以色列雷曼葛特市哈布魯亞路37號	
三、 申請人	姓名 (名稱) (中文)	1. 美商摩托羅拉公司	
	姓名 (名稱) (英文)	1. MOTOROLA INC.	
	國籍	1. 美國	
	住、居所 (事務所)	1. 美國伊利諾州史堪伯市東阿崗崑路1303號摩托羅拉中心	
	代表人 姓名 (中文)	1. F. 強 莫辛格	
	代表人 姓名 (英文)	1. F. JOHN MOTSINGER	
			

本案已向

國(地區)申請專利

美國 US

申請日期

2000/01/07 09/479,200

案號

主張優先權

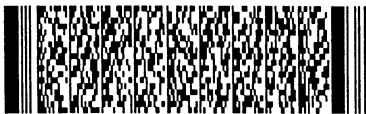
有

有關微生物已寄存於

寄存日期

寄存號碼

無



五、發明說明 (1)

發明範疇

本發明揭示一種用以執行高速低費用上下文轉換的裝置及方法，尤其是在允許多層巢狀中斷及例外狀況的處理器中執行高速低費用上下文轉換的裝置及方法。

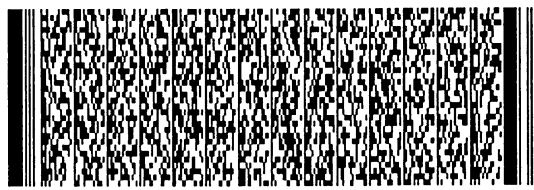
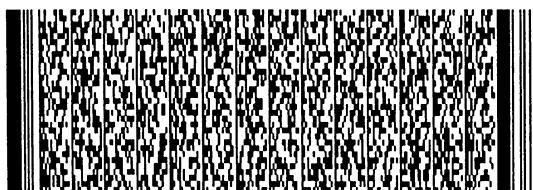
發明背景

大部份的處理器都具有耦合到暫存器檔案的中央處理器(CPU)。中央處理器通常也稱為算術邏輯單元(ALU)。處理器處理工作，而工作是獨立的控制執行緒。與任何工作關聯的是工作上下文。工作上下文是處理器定義關聯工作狀態並能夠繼續執行工作所需的資訊。工作上下文通常包括工作使用權、工作的程式計數器及程式狀態資訊等一般用途暫存器的上下文。工作上下文係由CPU儲存到暫存器檔案中。

處理器處理工作，直到工作結束或直到要求處理器處理較高優先順序的工作。暫停工作，並且處理器執行上下文轉換，讓處理器能夠處理較高優先順序的工作。通常會授予插斷及例外狀況相當高的優先順序。

在某些先前技藝方案中，於工作轉換期間，會將工作上下文轉移到內部或外部記憶體模組。在處理器處理較高優先順序的工作之後，就會從內部或外部記憶體模組擷取工作上下文。

相對高費用與用以執行上下文轉換之部份的先前技藝方法關聯。Siemens的TriCore架構中會實施此類的先前技藝方案。TriCore架構中使用的暫存器檔案被分割成兩半，

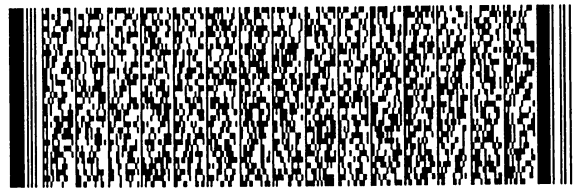


五、發明說明 (2)

稱之為上半部上下文及下半部上下文。TriCore的記憶體模組內具有複數個上下文儲存區域(CSA)。每個CSA都可儲存上半部上下文或下半部上下文。各種CSA互相連結在一起。除非至少將下半部上下文轉移到CSA，否則處理器無法開始處理較高優先順序的工作。此方案耗時且會導致相當高的費用。

Motorola M*Core晶片具有即時事件處理的非常低費用上下文轉換功能。M*Core晶片具有兩個暫存器檔案。一是一般暫存器檔案，另一個是替代暫存器檔案。替代暫存器檔案會降低與關鍵工作之上下文轉換暨儲存/還原時間關聯的費用。當選擇時，針對通常使用一般暫存器的所有指令，替代暫存器檔案會取代一般暫存器檔案。替代暫存器檔案中保留重要的參數及指標值，因此輸入高優先順序工作時能夠很容易存取。當最多有兩個優先順序等級時，M*Core的效率非常高。兩個優先順序等級指示，在每個給定的瞬間最多有兩個相關的上下文-較低優先順序工作上下文及較高優先順序工作上下文。一般暫存器檔案中可儲存第一項，而是替代暫存器檔案中可儲存第二項。因此，上下文轉換不需要將任何上下文寫入到記憶體模組。因此，M*Core的兩個暫存器檔案具有即時事件的非常低費用上下文轉換功能。

近年來，CPU的複雜度，尤其是CPU處理的多樣化工作已成長許多。處理器必須能夠支援多層巢狀工作。先前技藝方法及裝置不具有支援多層(>2)巢狀工作的高速低費用上



五、發明說明 (3)

下文轉換功能。

需要提供一種能夠在支援多層巢狀工作的處理器中，以非常低費用來執行快速上下文轉換的裝置及方法。

圖式簡單描述

圖1顯示根據本發明較佳具體實施例之用以執行高速低費用上下文轉換之裝置的原理圖；

圖2顯示本發明各種操作狀況的原理時序圖；以及

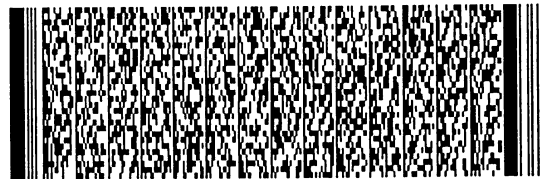
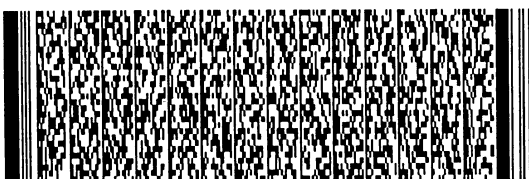
圖3顯示用以執行高速低費用上下文轉換之方法的原理流程圖。

圖式詳細說明

請注意，詳細說明及附圖中所採用的特定術語及措辭及所發表的特定結構及特定操作細節都是基於解說用途所提供，而不是用來在任何程度上限制如隨附之申請專利範圍中說明的本發明範疇。

本發明揭示一種在支援多層巢狀工作的處理器中，執行高速低費用上下文轉換的裝置及方法。只要處理器能夠伺候多達 $N-1$ 個巢狀工作，複數個 $(N-1)$ 暫存器檔案就可讓處理器藉由在暫存器檔案間轉換的方式來執行上下文轉換，而不需要等待將上下文轉移到暫存器檔案，或從暫存器檔案轉移上下文。

本發明提供一種用以執行高速低費用上下文轉換的裝置及方法。處理器會在記憶體模組中的上下文儲存區域與暫存器檔案間轉移第一工作上下文，而使用另一個暫存器檔案來處理第二工作。



五、發明說明 (4)

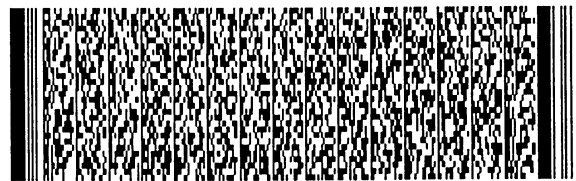
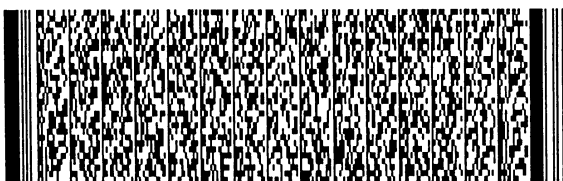
為了便於解說，服務工作優先順序高於處理器目前所處理之工作的要求稱為正向要求(FR)，返回處理先前暫停之較低優先順序工作的要求稱為反向要求(BR)。

本發明提供複數個暫存器檔案及直接記憶體存取機制，可讓處理器響應正向要求，其方式是使用第一暫存器檔案來開始處理較高優先順序的工作，而將暫停的工作上下文從第二暫存器檔案轉移到記憶體模組內的上下文儲存區域。另外，處理器響應反向要求的方式是，使用第一暫存器檔案中儲存的上下文，而將較低優先順序工作上下文轉移到第二暫存器檔案。

通常會授予插斷及例外狀況較高的優先順序。在諸如PowerPC601之類的各種處理器中，會將較高的優先順序授予給諸如非同步化、不精確例外狀況之類的某些例外狀況，而將較低的優先順序授予給諸如同步化精確例外狀況之類的其他例外狀況。

當管線處理器中發生精確例外狀況時，會在執行指令資料流中所有先前的指令後發出正向要求81。

圖1顯示用以執行高速低費用上下文轉換之裝置10的原理圖。裝置10很方便構成處理器的一部份，接著具有CPU 50。裝置10包括第一暫存器檔案20、第二暫存器檔案30、控制單元40及記憶體模組60。CPU 50經由資料匯流排21及控制暨位址匯流排31耦合到暫存器檔案20及暫存器檔案30。控制單元40經由第一控制匯流排41及第二控制匯流排42耦合到暫存器檔案20及30，因此，進一步經由第三控制



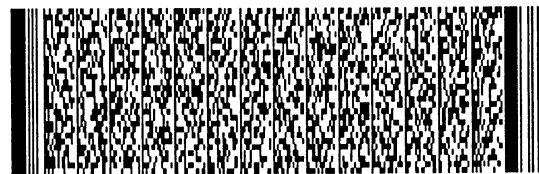
五、發明說明 (5)

匯流排43及第二控制匯流排44耦合到記憶體模組60。暫存器檔案20及30經由DMA資料匯流排33耦合到記憶體模組60。

控制單元40接收正向要求81及反向要求82，並決定CPU 50可存取哪一個暫存器檔案20/30，以及暫存器檔案20/30與記憶體模組60內的上下文儲存區域之間要轉移哪一個上下文。正向要求81通常稱為「插斷要求」，而反向要求82稱為「從插斷要求返回」。控制單元40經由控制匯流排41及42分別傳送控制信號CPU₁83及CPU₂84，以決定CPU 50可存取暫存器檔案20及30中的哪一個，從某方面來說，在每個給定的瞬間，CPU 50可存取單一暫存器檔案20/30。控制單元40經由控制匯流排41及42分別傳送控制信號DMA₁85及DMA₂86，以決定哪一個暫存器檔案20及30涉及上下文轉換，以及是否將上下文從暫存器檔案20/30轉移到上下文儲存區域，或是相反。

記憶體模組60具有可儲存上下文的複數個上下文儲存區域。就便利性而言，有N-1個上下文儲存區域，而N是巢狀深度。

就便利性而言，控制單元40具有直接記憶體存取(DMA)控制器，可允許在暫存器檔案20暨30與記憶體模組60DMA內的上下文儲存區域間進行上下文轉換。使用位址匯流排44暨控制匯流排43，以及控制匯流排41及42，DMA控制決定參與上下文轉換之上下文儲存區域的位址，並決定是否將上下文從暫存器檔案轉移到上下文儲存區域，或是相



五、發明說明 (6)

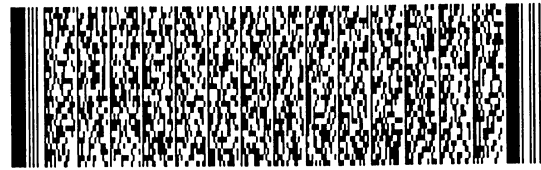
反。

就便利性而言，為了快速進行寫入及儲存處理程序，DMA 資料匯流排33相當寬。

請參考圖2，在 T_1 的瞬間，CPU 50處理 P_4 優先順序工作。記憶體模組60所有的上下文儲存區域都是空的。第一、第二及第三上下文儲存區域71、72及73的位址分別是0000、0100及1000。暫存器檔案20保存 P_4 優先順序工作上下文。

在 T_2 的瞬間，要求CPU 50處理 P_3 優先順序工作。因此，正向要求81變成高位準狀態，而導致控制單元40初始化上下文轉換。短暫時間後，正向要求81變成低位準狀態。控制信號 CPU_1 83變成低位準狀態，而控制信號 CPU_2 84變成高位準狀態，指示CPU 50存取暫存器檔案30，而不是存取暫存器檔案20。控制信號 DMA_1 85及 CMR_1 91變成高位準狀態，並初始化從暫存器檔案20轉換 P_4 優先順序工作上下文的DMA轉換。經由匯流排44傳送到記憶體模組60之位址信號 ADD_S 75的值等於0000。在 T_3 結束從暫存器檔案20到第一上下文儲存區域71的 P_4 優先順序工作上下文轉換，此時 ADD_S 75、 DMA_1 85及 CMR_1 91都處於低位準狀態。

在 T_4 的瞬間，要求CPU 50處理 P_2 優先順序工作。因此，正向要求81變成高位準狀態，而導致控制單元40初始化上下文轉換。短暫時間後，正向要求81變成低位準狀態。控制信號 CPU_1 83變成高位準狀態，而控制信號 CPU_2 84變成低位準狀態，指示CPU 50存取暫存器檔案20，而不是存取暫存器檔案30。控制信號 DMA_2 86及 CMR_2 92變成高位準狀態，

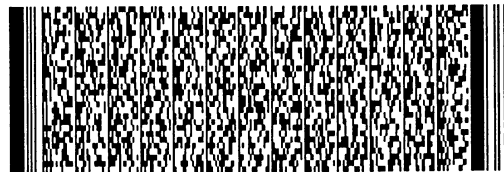
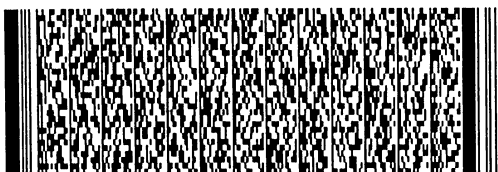


五、發明說明 (7)

並初始化從暫存器檔案30轉換 P_3 優先順序工作上下文的DMA轉換。經由匯流排44傳送到記憶體模組60之位址信號ADDS 75的值等於0100。在 T_5 結束從暫存器檔案30到第二上下文儲存區域72的 P_3 優先順序工作上下文轉換，此時ADDS 75、DMA₂86及CMR₂92都處於低位準狀態。

在 T_6 的瞬間，要求CPU 50處理 P_1 優先順序工作。因此，正向要求81變成高位準狀態，而導致控制單元40初始化上下文轉換。短暫時間後，正向要求81變成低位準狀態。控制信號CPU₁83變成低位準狀態，而控制信號CPU₂84變成高位準狀態，指示CPU 50存取暫存器檔案30，而不是存取暫存器檔案20。控制信號DMA₁85及CMR₁91變成高位準狀態，並初始化從暫存器檔案20轉換 P_2 優先順序工作上下文的DMA轉換。經由匯流排44傳送到記憶體模組60之位址信號ADDS 75的值等於1000。在 T_7 結束從暫存器檔案20到第三上下文儲存區域73的 P_2 優先順序工作上下文轉換，此時ADDS 75、DMA₁85及CMR₁91都處於低位準狀態。

在 T_8 的瞬間， P_1 優先順序工作結束，因此反向要求82變成高位準狀態。短暫時間後，反向要求82變成低位準狀態。CPU 50重新開始處理 P_2 優先順序工作，而 P_2 優先順序工作上下文係儲存在暫存器檔案20中。控制信號CPU₁83變成高位準狀態，而控制信號CPU₂84變成低位準狀態，指示CPU 50存取暫存器檔案20，而不是存取暫存器檔案30。控制信號DMA₂86及CMW₂94變成高位準狀態，並初始化將 P_3 優先順序工作上下文轉換到暫存器檔案30的DMA轉換。經由



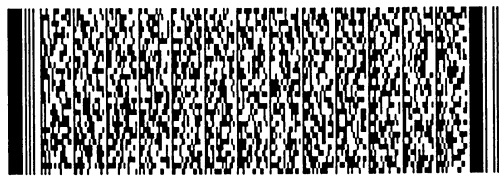
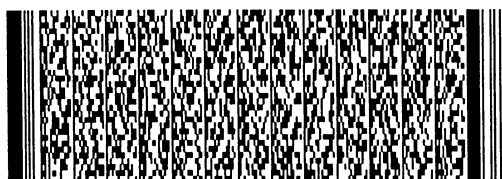
五、發明說明 (8)

匯流排44傳送到記憶體模組60之位址信號ADDS 75的值等於0100。在 T_9 結束從第二上下文儲存區域72到暫存器檔案30的 P_3 優先順序工作上下文轉換，此時ADDS 75、DMA₂86及CMW₂94都處於低位準狀態。

在 T_{10} 的瞬間， P_2 優先順序工作結束，因此反向要求82變成高位準狀態。短暫時間後，反向要求82變成低位準狀態。CPU 50重新開始處理 P_3 優先順序工作，而 P_3 優先順序工作上下文係儲存在暫存器檔案30中。控制信號CPU₁83變成低位準狀態，而控制信號CPU₂84變成高位準狀態，指示CPU 50存取暫存器檔案30，而不是存取暫存器檔案20。控制信號DMA₁85及CMW₁93變成高位準狀態，並初始化將 P_4 優先順序工作上下文轉換到暫存器檔案20的DMA轉換。經由匯流排44傳送到記憶體模組60之位址信號ADDS 75的值等於0000。在 T_{11} 結束從第一上下文儲存區域71到暫存器檔案20的 P_4 優先順序工作上下文轉換，此時ADDS 75、DMA₁85及CMW₁93都處於低位準狀態。

在 T_{12} 的瞬間， P_3 優先順序工作結束，因此反向要求82變成高位準狀態。短暫時間後，反向要求82變成低位準狀態。CPU 50重新開始處理 P_4 優先順序工作，而 P_4 優先順序工作上下文係儲存在暫存器檔案20中。控制信號CPU₁83變成高位準狀態，而控制信號CPU₂84變成低位準狀態，指示CPU 50存取暫存器檔案20，而不是存取暫存器檔案30。

就便利性而言，為了確保不會損壞上下文，當在記憶體模組60與暫存器檔案20或暫存器檔案30任一個之間轉換上



五、發明說明 (9)

下文時，會遮罩正向要求信號81及反向要求信號82。

還進一步加速上下文轉換，其方式是運用可變大小的上下文，從某方面來說，暫存器檔案到上下文儲存區域之間只轉換暫存器檔案的一部份。為了實施此類的機制，控制單元40必須接收用來指示所轉換之上下文大小的SIZE信號，並且必須儲存先前的SIZE信號，以利於將上下文擷取到暫存器檔案。再者，ADDS 75信號值將取決於先前的ADDS 75值及先前的SIZE信號值。上下文儲存區域將具有對應於SIZE信號的可變大小。DMA控制器應接收SIZE信號，並照著只轉換SIZE長度的上下文。

還進一步加速上下文轉換，其方式是運用兩個以上的暫存器檔案，使得CPU 50可在兩個以上的工作間進行轉換，而不需要執行從記憶體模組60到暫存器檔案的上下文轉換。為了實施此類的機制，控制單元40應將額外的控制信號提供給額外的暫存器檔案，並提供給記憶體模組60。額外的暫存器檔案最好類似於暫存器檔案20及30，並以相同的方式耦合到CPU 50、控制單元40及記憶體模組60。例如，如果有一個額外的暫存器檔案(未顯示)，則控制單元40將照著提供諸如CPU₃及DMA₃之類的控制信號，其類似於CPU₁/CPU₂及DMA₁/DMA₂。額外的暫存器檔案應經由第三DMA匯流排耦合到記憶體模組60，並且控制單元40會將控制信號CMR₃及CMW₃(類似於CMR₁/CMR₂及CMW₁/CMW₂)提供給記憶體模組60。

圖3顯示可讓CPU 50執行高速低費用上下文轉換之方法



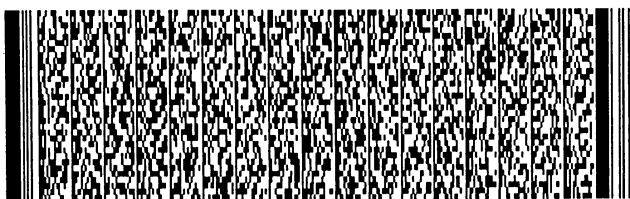
五、發明說明 (10)

的流程圖。

目前正在處理的工作稱為「現行工作」。與現行工作關聯的上下文稱為「現行優先順序工作上下文」。處理現行工作時所存取的暫存器檔案稱為「現行暫存器檔案」。優先順序高於現行工作的工作稱為「較高優先順序工作」。與較高優先順序工作關聯的上下文稱為「較高優先順序工作上下文」。儲存較高優先順序工作的暫存器檔案稱為「較高優先順序暫存器檔案」。優先順序低於現行工作的工作稱為「較低優先順序工作」。與較低優先順序工作關聯的上下文稱為「較低優先順序工作上下文」。儲存較低優先順序工作的暫存器檔案稱為「較低優先順序暫存器檔案」。優先順序更低於較低優先順序工作的工作稱為「更低優先順序工作」。與更低優先順序工作關聯的上下文稱為「更低優先順序工作上下文」。儲存更低優先順序工作的暫存器檔案稱為「更低優先順序暫存器檔案」。

矩形方塊110、120及130代表方法步驟。本方法包括下列步驟：

由CPU 50處理(步驟110)現行工作，而於處理程序期間，允許CPU 50存取現行暫存器檔案。較低優先順序暫存器檔案及記憶體模組60內的上下文儲存區域中可儲存較低優先順序工作上下文。在此類的方法中，如果已事先暫停CPU 50處理較低優先順序工作，以便處於優先順序高於較低優先順序工作的工作時，則會儲存較低優先順序工作。如路徑112所示，如果需要處理較高優先順序工作，



五、發明說明 (11)

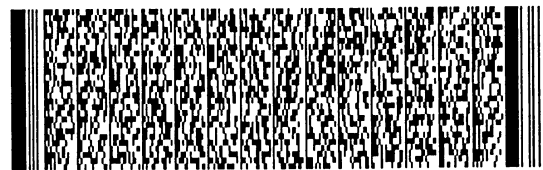
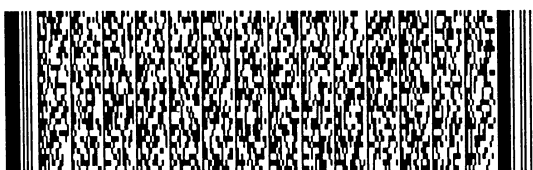
則會在步驟110之後接著步驟120。如路徑118所示，如果現行工作結束並且需要重新開始處理較低優先順序工作，則會在步驟110之後接著步驟130。

如果接收到處理較高優先順序工作的要求，則會執行正向上下文轉換(步驟120)。於步驟120期間，現行暫存器檔案會與較高優先順序暫存器檔案轉換，並將現行工作的上下文轉移到記憶體模組60。就便利性而言，除非將現行工作上下文轉移到記憶體模組60，否則會防止CPU 50存取現行暫存器檔案；跳至步驟110，而現行工作變成較低優先順序工作，並且較高優先順序工作變成現行工作，如路徑122所示。

如果接收到重新開始處理較低優先順序工作的要求，則會執行反向上下文轉換(步驟130)。於步驟130期間，現行暫存器檔案會與較低優先順序暫存器檔案轉換。如果記憶體模組60儲存有效的更低優先順序工作上下文，則會將更低優先順序工作上下文轉移到更低優先順序暫存器檔案；跳至步驟110，而較低優先順序工作變成現行工作，並且，如果具有有效的更低優先順序工作，則更低優先順序工作變成較低優先順序工作，如路徑132所示。

例如，假設有三等級的優先順序 P_1 、 P_2 及 P_3 ，其中 P_1 是最高優先順序而 P_3 是最低優先順序；有兩個暫存器檔案20暨30，並且記憶體模組60內有兩個上下文儲存區域71暨72。

於步驟110期間，CPU 50處理 P_2 優先順序工作，且允許CPU 50存取暫存器檔案20。 P_2 優先順序工作是現行工作，



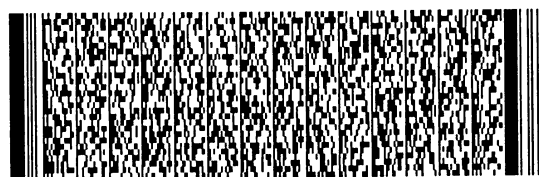
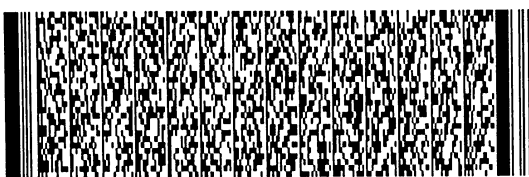
五、發明說明 (12)

而暫存器檔案20是現行暫存器檔案。之前，當接收到處理 P_2 優先順序工作的要求時，CPU 50停止處理 P_3 優先順序工作。 P_3 優先順序工作是較低優先順序工作。 P_3 優先順序工作的上下文是較低優先順序工作上下文，其儲存在暫存器檔案30(稱為「較低優先順序暫存器檔案」)及第一上下文儲存區域71(稱為「較低優先順序上下文儲存區域」)中。

如路徑118所示，當CPU 50完成處理 P_2 優先順序工作時，會在步驟110之後接著步驟130，在步驟130中處理 P_3 優先順序工作，並且CPU 50存取暫存器檔案20。因此，暫存器檔案20變成現行暫存器檔案，而 P_3 優先順序工作變成現行工作。

於步驟110期間，如果要求CPU 50處理 P_1 優先順序工作，則會在步驟110之後接著步驟120，在步驟120中執行第一類型上下文轉換。 P_1 優先順序工作是現行工作。 P_2 優先順序工作變成較低優先順序工作，而 P_3 優先順序工作變成更低優先順序工作。於步驟120期間，會將 P_2 優先順序工作上下文(也就是，較低優先順序工作上下文)轉移到記憶體模組60內的上下文儲存區域。如路徑122所示，會在步驟120之後接著步驟110，於步驟110處理 P_1 優先順序工作。

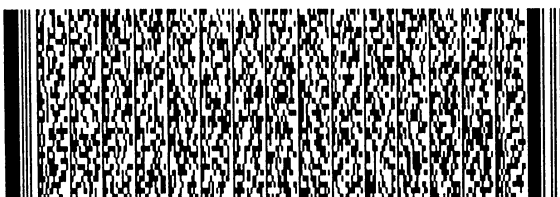
請注意，詳細說明及附圖中所採用的特定術語及措辭及所發表的特定結構及特定操作細節都是基於解說用途所提供，而不是用來在任何程度上限制如隨附之申請專利範圍中說明的本發明範疇。



五、發明說明 (13)

因此，本文中已說明的具體實施例包括用以執行高速低費用上下文轉換之裝置暨方法之改良方法及裝置的至少一項較佳具體實施例。熟知技藝人士應知道，可以各種方式來修改所發表的主題，並可採用除了上文具體開始及說明之較佳形式以外的許多具體實施例。

因此，上文發表的主題應視為解說而不應視為限制，並在法律允許的最大範圍內，隨附的申請專利範圍涵蓋所有此類的修改及其他具體實施例，並屬於本發明的法定精神與範疇內。本發明的範疇係藉由下列申請專利範圍可允許的最廣解說及其同等品所決定，而不是由前文的詳細說明所決定。

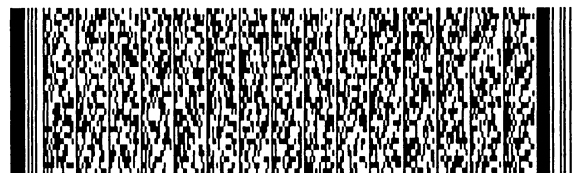
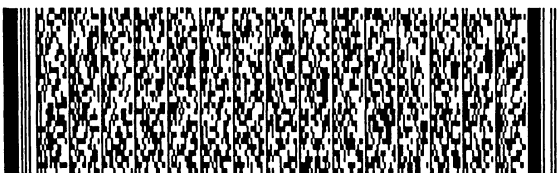


四、中文發明摘要 (發明之名稱：執行高速低費用上下文轉換之裝置及方法)

本發明揭示一種用以執行高速低費用上下文轉換之裝置(10)及方法，尤其是在處理多層巢狀工作之處理器中進行。裝置(10)處理正向要求(81)及反向要求(82)。裝置(10)耦合到CPU(50)並具有複數個暫存器檔案(20, 30)及一直接記憶體存取機制，可讓處理器響應正向要求(81)，其方式是使用第一暫存器檔案(20)來開始處理較高優先順序的工作，同時將暫停的工作上下文轉移到記憶體模組(60)內的上下文儲存區域。處理器響應反向要求(82)的方式是，使用在第一暫存器檔案(20)中儲存的上下文，而將較低優先順序工作上下文轉移到第二暫存器檔案(30)。

英文發明摘要 (發明之名稱：DEVICE AND METHOD FOR PERFORMING HIGH-SPEED LOW OVERHEAD CONTEXT SWITCH)

A device (10) and a method for performing high speed low overhead context switch, and especially in processors that handle multilevel nested tasks. The device (10) handles forward requests (81) and backward requests (82). The device (10) is coupled to a CPU (50) and has a plurality of register files (20, 30) and a direct memory access mechanism that allows the processor to respond to the forward request (81) by starting to handle a higher priority task using a first register file



四、中文發明摘要 (發明之名稱：執行高速低費用上下文轉換之裝置及方法)

英文發明摘要 (發明之名稱：DEVICE AND METHOD FOR PERFORMING HIGH-SPEED LOW OVERHEAD CONTEXT SWITCH)

(20) while transferring the halted task context to a context save area within a memory module (60). The processor responds to the backward request (82) by using the context that is stored in a first register file (20), while transferring to a second register file (30) a lower priority task context.



六、申請專利範圍

1. 一種用以在中央處理單元CPU(50)中執行上下文轉換的裝置(10)，該CPU(50)適用於處理各種優先順序的工作，該裝置(10)包括：

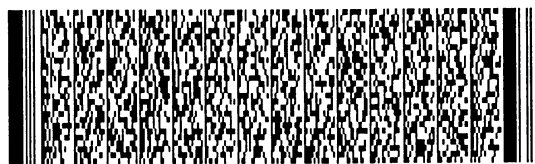
耦合到該CPU(50)的暫存器檔案(20,30)；

耦合到該暫存器檔案(20,30)的記憶體模組(60)，適用於儲存工作上下文；

耦合到該CPU(50)、該暫存器檔案(20,30)及該記憶體模組(60)的控制單元(40)；及

其中該控制單元(40)係用來接收一正向要求(81)，以用於處理優先順序高於該CPU(50)所處理之現行工作的工作，用來在儲存現行工作上下文之現行暫存器檔案(20)與其他暫存器檔案(30)間進行轉換，以一種使得該CPU(50)能夠存取其他暫存器檔案(30)之方式，以及，用來將現行工作上下文從現行暫存器檔案(20)轉移到該記憶體模組(60)。

2. 如申請專利範圍第1項之裝置(10)，其中該控制單元(40)係用來接收一反向要求(82)，以利於重新開始處理先前的工作、處理由於處理較高優先順序工作的要求導致先前所暫停的先前工作；用來在儲存較高優先順序工作上下文之暫存器檔案(20)與儲存先前工作上下文之先前暫存器檔案(30)間進行轉換，使得該CPU(50)能夠存取先前暫存器檔案(30)；以及，如果該記憶體模組(60)儲存優先順序低於先前工作之工作的上下文，則將較低優先順序工作上下文轉移到另一個現行暫存器檔案。



六、申請專利範圍

3. 如申請專利範圍第2項之裝置(10)，其中在任一暫存器檔案與該記憶體模組(60)間轉換上下文期間，會遮罩該反向要求(82)及該正向要求(81)。

4. 如申請專利範圍第2項之裝置(10)，其中該記憶體模組(60)具有複數個上下文儲存區域(71、72、73)，每個上下文儲存區域都是用來儲存單一上下文。

5. 如申請專利範圍第2項之裝置(10)，其中該控制單元(40)具有一直接記憶體存取控制器，用來控制該記憶體模組(60)與該等暫存器檔案(20、30)間的上下文轉換。

6. 如申請專利範圍第2項之裝置(10)，其中在該等暫存器檔案(20, 30)係經由一寬資料匯流排(33)耦合到該記憶體模組(60)。

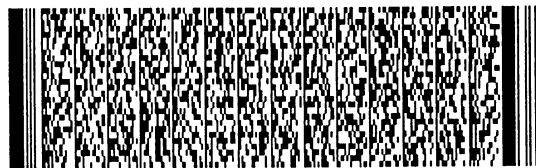
7. 如申請專利範圍第2項之裝置(10)，其中該裝置(10)適用於處理可變大小上下文，其中該裝置(10)適用於接收正向要求(81)及定義現行工作上下文大小的SIZE資訊；適用於將現行工作上下文從該現行暫存器檔案(20)轉移到該記憶體模組(60)，以利於儲存該SIZE資訊；以及

其中該裝置(10)適用於接收反向要求(82)並使用所儲存的SIZE資訊，以利於將先前工作上下文從該記憶體模組(60)轉移到該先前暫存器檔案(30)。

8. 一種具有高速低費用上下文轉換功能的處理器，該處理器適用於處理各種優先順序的工作，該處理器包括：

一中央處理單元(CPU 50)；

耦合到該CPU(50)的暫存器檔案(20, 30)；



六、申請專利範圍

耦合到該暫存器檔案(20, 30)的記憶體模組(60)，適用於儲存工作上下文；

耦合到該CPU(50)、該暫存器檔案(20, 30)及該記憶體模組(60)的控制單元(40)；及

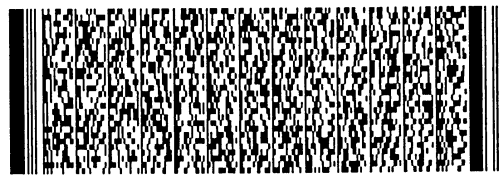
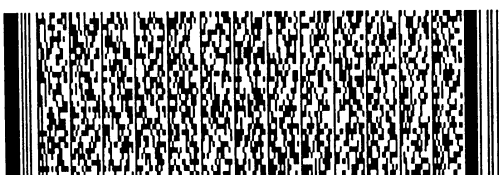
其中該控制單元(40)係用來接收一正向要求(81)，以用於處理優先順序高於該CPU(50)所處理之現行工作的工作，用來在儲存現行工作上下文之現行暫存器檔案(20)與其他暫存器檔案(30)間進行轉換，以一種使得該處理器能夠存取其他暫存器檔案(30)的方式，以及，用來將現行工作上下文從現行暫存器檔案(20)轉移到該記憶體模組(60)。

9. 一種用以在一處理器中執行高速低費用上下文轉換的方法，該方法包括下列步驟：

處理一現行工作，其中允許該處理器的CPU(50)存取現行暫存器檔案(20)；

接收一處理較高優先順序工作的要求，並照著在現行暫存器檔案(20)與較高優先順序暫存器檔案(30)間進行轉換，並將現行工作上下文從現行暫存器檔案(20)轉移到記憶體模組(60)，跳至處理一現行工作的步驟，其中較高優先順序工作變成現行工作，並且較高優先順序暫存器檔案(30)變成現行暫存器檔案；以及

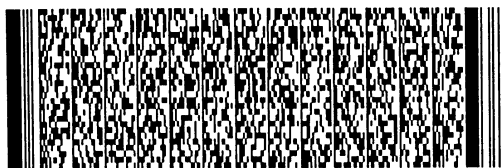
接收一重新開始處理較低優先順序工作的要求，並照著在現行暫存器檔案與較低優先順序暫存器檔案間進行轉換，如果該記憶體模組(60)儲存一更低優先順序工作上下



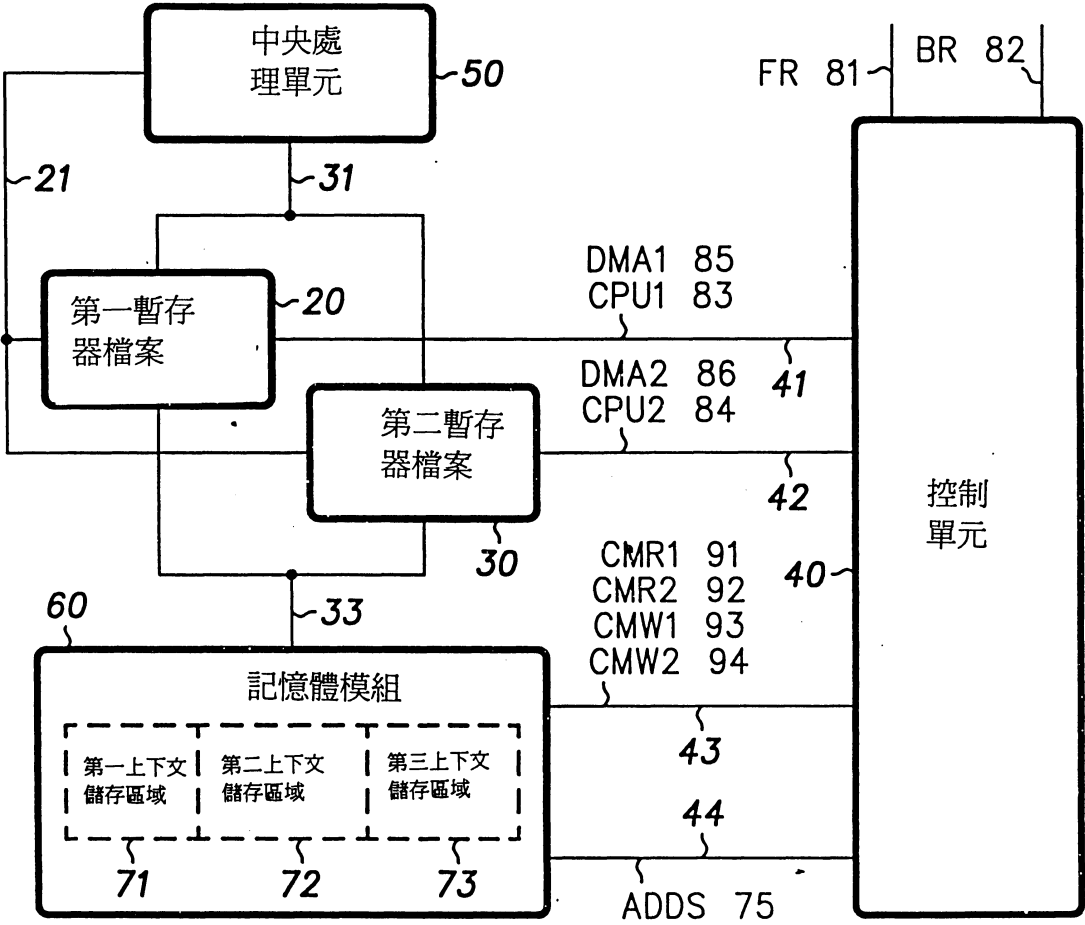
六、申請專利範圍

文，則會將該更低優先順序工作上下文轉移到現行暫存器檔案，跳至處理一現行工作的步驟，其中較低優先順序工作變成現行工作，並且，如果具有更低優先順序工作，則更低優先順序工作變成較低優先順序工作。

10. 如申請專利範圍第9項之方法，其中該等上下文係儲存在該記憶體模組(60)的上下文儲存區域(71、72、73)中，每個上下文儲存區域都是用來儲存單一上下文。



圖式



圖式

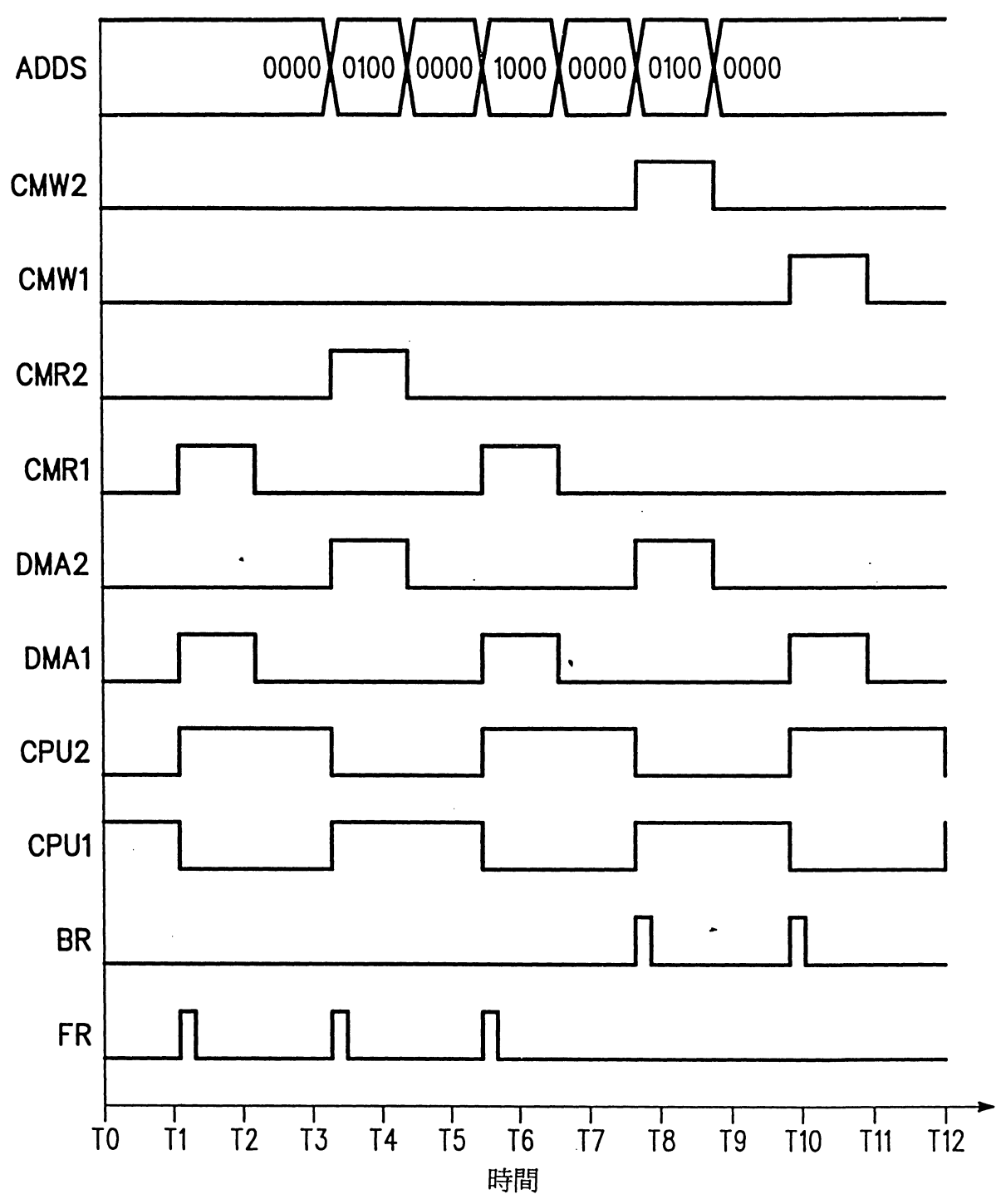


圖 2

圖式

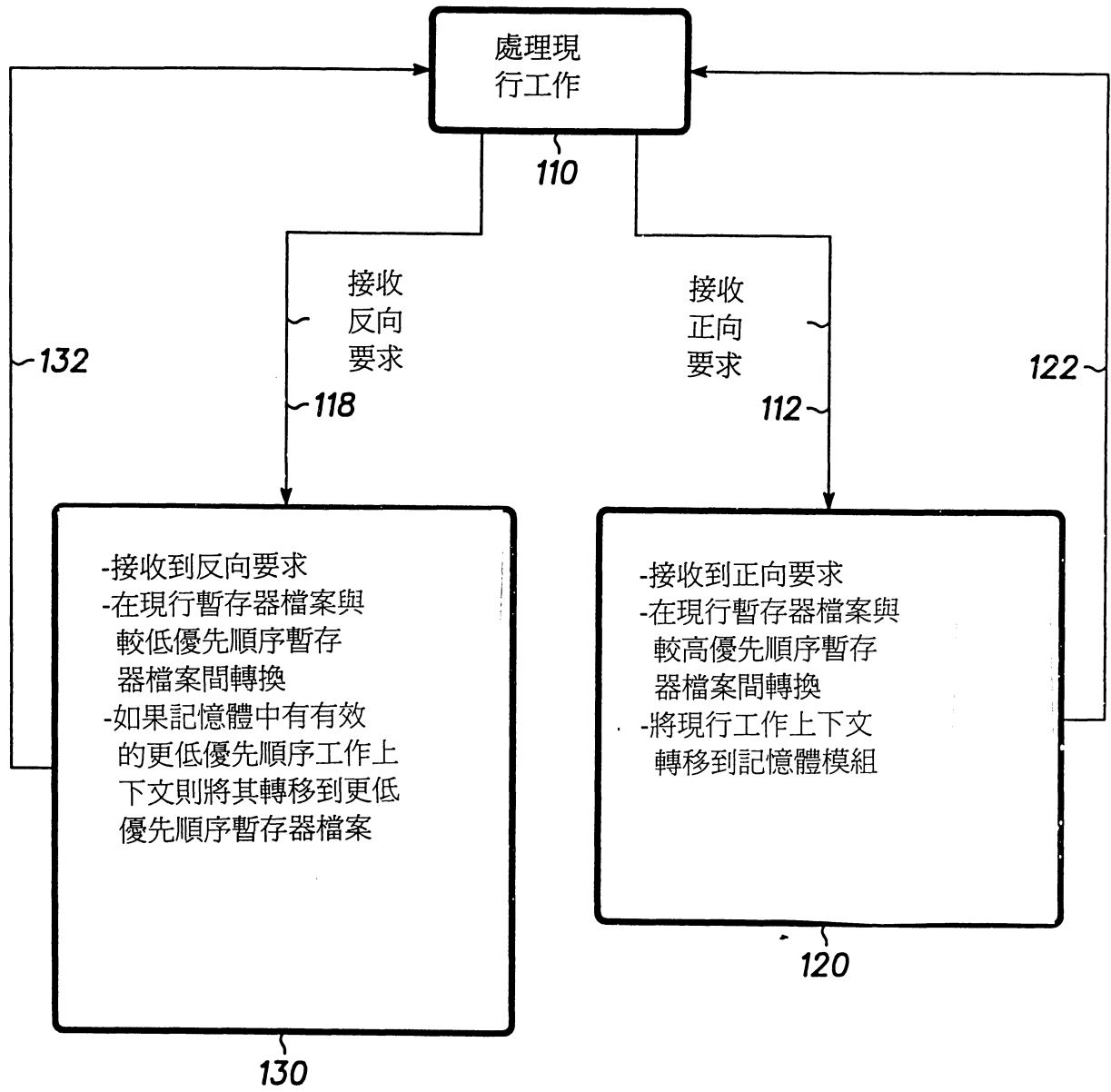


圖 3

圖式簡單說明

圖式元件符號說明

- 10 裝置
- 20, 30 暫存器檔案
- 21 資料匯流排
- 31 控制及位址匯流排
- 33 DMA 資料匯流排
- 40 控制單元
- 41 第一控制匯流排
- 42 第二控制匯流排
- 43 第三控制匯流排
- 44 記憶體位址匯流排
- 50 中央處理器
- 60 記憶體模組
- 71, 72, 73 第一、第二、第三上下文儲存區域
- 81 正向要求
- 82 反向要求

增修
多頁
示
本案
修正
後
是否
變更
實質
內容

